

PRZEWODNIK PO METODYKACH, KTÓRE MUSISZ POZNAĆ! ➤ Jak wybrać metodę działania odpowiednią dla konkretnych projektów i organizacji? ➤ Co pozwala skutecznie zrealizować stworzone plany działania? ➤ Gdzie szukać wiedzy tajemnej z zakresu metodyk zarządzających, wytwórczych i organizacyjnych?

ZARZĄDZANIE PROJEKTAMI IT

PRZEWODNIK PO METODYKACH

Adam **KOSZLAJDA**



Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiejkolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redakcja: Krzysztof Zemanek

Projekt okładki: Urszula Banaszewska

Materiały graficzne na okładce zostały wykorzystane za zgodą iStockPhoto Inc.

Wydawnictwo HELION

ul. Kościuszki 1c, 44-100 GLIWICE

tel. 32 231 22 19, 32 230 98 63

e-mail: helion@helion.pl

WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

http://helion.pl/user/opinie?proinf_ebook

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

ISBN: 978-83-246-4546-6

Copyright © Helion 2010

Printed in Poland.

- [Poleć książkę na Facebook.com](#)
- [Kup w wersji papierowej](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

Spis treści

Wstęp	7
Część I Metodyki zarządcze a praktyka	11
Rozdział 1. PProject IN Controlled Environment — Prince2	13
Szczypta historii	13
Procesy	14
Komponenty	17
Techniki	21
Zarządzanie dokumentacją	24
Dostosowywanie do potrzeb organizacji	25
Certyfikacja	25
Podsumowanie	26
Rozmowa z... ..	27
Rozdział 2. Project Management Body of Knowledge — PMBoK	31
Szczypta historii	31
Obszary wiedzy	33
Procesy i techniki	39
Dostosowanie do potrzeb organizacji	66
Certyfikacja	66
Podsumowanie	67
Część II Metodyki wytwórcze a praktyka	69
Rozdział 3. Rational Unified Process (RUP)	73
Szczypta historii	73
Proces	74
Dyscypliny RUP	76
Abecadło metodyki RUP	79
Adaptacja RUP do potrzeb organizacji	80
Podsumowanie RUP	81
Rozmowa z... ..	82
Przykład Prince2 i RUP — BlogSerwis	85

Rozdział 4. Microsoft Solution Framework (MSF)	105
Szczypta historii	105
Proces	106
Model zespołu	107
Faza Wizji	108
Faza Planowania	109
Faza Konstrukcji	112
Faza Stabilizacji	116
Faza Wdrożenia	120
MSF > MOF	121
Trójkąt negocjacyjny	123
Dyscypliny zarządcze	125
Certyfikacja	126
Podsumowanie — MSF a RUP	126
Rozdział 5. Przykład PMBoK i MSF — wdrożenie systemu BI	129
Część III Metodyki adaptacyjne a praktyka	177
Rozdział 6. eXtreme Programming	179
Szczypta historii	179
Role	180
Proces	180
Wdrożenie	186
Rozdział 7. Scrum	189
Szczypta historii	190
Role	190
Proces	191
Podsumowanie	196
Rozmowa z...	197
Rozdział 8. Joel o oprogramowaniu	205
Rozdział 9. Przykład — Scrum BlogSerwis	209
Część IV Metodyki organizacyjne a praktyka	223
Rozdział 10. Capability Maturity Model Integration (CMMI)	225
Szczypta historii	226
Poziomy CMMI	228
Procesy	229
Podsumowanie	235
Rozdział 11. Six Sigma	239
Szczypta historii	240
Wdrożenie	241
Certyfikacja	245
Podsumowanie	245
Rozdział 12. Information Technology Infrastructure Library (ITIL)	247
Szczypta historii	247
Role	249
Procesy	251
Wdrożenie	254

Certyfikacja	256
Podsumowanie	257
Rozmowa z... ..	258
Rozdział 13. Control Objectives for Information and related Technology (COBIT)	263
Szczypta historii	264
Role	265
Procesy	266
Certyfikacja	273
Podsumowanie	274
Część V Rozwiązania kombinowane	275
Podsumowanie	281
Dodatki	283
Dodatek A Lista wszystkich procesów PMBoK	285
Dodatek B Specyfikacja dyscyplin RUP z Rational Method Composer (RMC) ...	321
Dodatek C Lista wszystkich procesów ITIL	325
Dodatek D Lista wszystkich procesów COBIT	331
Spis rysunków	335
Spis tabel	337
Źródła	339
Skorowidz	343

Wstęp

Prasa informatyczna pełna jest tajemniczych skrótów nazw metodyk, takich jak ITIL, COBIT, CMMI, TQM, PMBOK, PRINCE, Six Sigma, RUP, MSF czy Agile. Nie brakuje egipskich kapłanów, którzy są w stanie podporządkować sobie masy zaklęciami typu: „Według metodyki takiej to a takiej musimy teraz wykonać analizę X oraz zbiór dokumentów Y”. A co naprawdę kryje się pod tymi skrótami?

Rzecz w tym, że każda metodyka¹ powinna zaistnieć w firmie jako potrzeba zrealizowania konkretnego zadania. Oto one:

- ◆ Zarządzanie projektami (PMBOK, Prince2),
- ◆ Zarządzanie wytwarzaniem oprogramowania (RUP, MSF, Agile),
- ◆ Zarządzanie kwestiami organizacyjnymi (ITIL, CMMI, COBIT, Six Sigma).

I tutaj po raz pierwszy pojawia się wujek Dobra Rada... (patrz ramka na następnej stronie).

Jeżeli znasz dogłębnie każdą z wymienionych metodyk

— to nie jest książka dla Ciebie.

Jeżeli w ogóle nie interesuje Cię metodyczne rozwiązywanie problemów

— to nie jest książka dla Ciebie.

Jeżeli szukasz akredytowanych materiałów opisujących szczegółowo konkretną metodykę

— to nie jest książka dla Ciebie.

Jeżeli szukasz przekrojowej lektury o metodykach stosowanych obecnie w Polsce

— to jest książka dla Ciebie.

¹ Często myli się metodykę, czyli ustandaryzowane podejście do rozwiązania problemu (metodę), z metodologią, czyli nauką o metodach (m.in. badaniem ich skuteczności).



Jeżeli nie jesteś osobą oporną i tolerujesz częste „w praktyce” oraz emotikony, takie jak ta ☺

— to jest książka dla Ciebie.

Jeżeli masz związek z projektami IT, a szukasz zasobnika porad i technik³

— to jest książka dla Ciebie.

Głównym celem tej książki jest przegląd metodyk IT wykorzystywanych na polskim rynku. Dzięki niemu czytelnik ma możliwość uświadomienia sobie, jaki rodzaj rozwiązania najlepiej pasuje do jego aktualnych potrzeb. Opis poszczególnych metodyk nie jest wyczerpujący i w przypadku planowanego wdrożenia wybranej metodyki warto zaopatrzyć się w autoryzowane materiały z organizacji będących właścicielem danej metodyki. W związku z tym nie zakłada się, że każdy czytelnik będzie czytał tę książkę do deski do deski. Przykładowo...

- ◆ Programiści i analitycy zainteresują się głównie tematyką związaną z metodykami wytwórczymi,
- ◆ Administratorzy i użytkownicy będą prawdopodobnie zainteresowani metodykami organizacyjnymi,
- ◆ Pracownicy biura projektowego będą interesować się raczej metodykami zarządzczymi,

² Zgodnie z pierwszą radą Omara Shahine’a, proces nie zastąpi myślenia.

³ Z ang. *tips & tricks*.

- ♦ Testerzy zainteresują się kwestiami jakościowymi w metodykach wytwórczych i organizacyjnych,
- ♦ Decydenci i audytorzy będą zainteresowani głównie metodykami zarządczymi oraz organizacyjnymi,
- ♦ Kierownik projektu oraz ciekawscy powinni zainteresować się każdą grupą metodyk 😊.

Oczywiście, ról w organizacji oraz sposobu ich definiowania może być bardzo wiele i dlatego też powyższe sugestie należy traktować dość swobodnie.

Dobry wstęp to krótki wstęp, więc żeby nie przedłużać, postaram się mówić ludzkim językiem o nieludzkich sprawach 😊.

Konwencje zastosowane w książce

W książce wykorzystuje się następujące oznaczenia:

⋮ Porównanie różnych metodyk (całości lub elementów)



W praktyce

Opis praktycznego zastosowania metodyki (całości lub elementów) lub ciekawostki praktyczne



Uwaga

Pewne istotne założenia mające wpływ na zrozumienie całości

Część I

Metodyki zarządcze a praktyka

Metodyki zarządcze stosuje się zazwyczaj w sytuacji:

- ◆ gdy istotna jest precyzyjna regulacja i kontrola współpracy zleceniodawca-zleceniobiorca, a szczególnie wtedy, kiedy zleceniobiorca to firma zewnętrzna,
- ◆ gdy ważny jest nadzór nad przebiegiem projektu od strony formalnej, a szczególnie „materiał dowodowy” dotyczący podejmowanych decyzji projektowych,
- ◆ gdy mamy do czynienia ze średnim lub dużym przedsięwzięciem i wspólny słownik pojęć projektowych jest niezbędny do właściwej komunikacji.

Rozdział 1.

PProject IN Controlled Environment — Prince2

Prince2 to *PProject IN Controlled Environment*, czyli projekt w środowisku kontrolowanym. W praktyce stworzenie takiego środowiska oznacza właściwe udokumentowanie powodów uruchomienia projektu, jego przebiegu oraz zamknięcia. Prince2 koncentruje się właśnie na sposobach podejmowania decyzji w projekcie i kwestiach zarządczych związanych z jego realizacją. Metodyka ta nie wnika w szczegóły jego technicznej realizacji. Powoduje to, że zastosowanie Prince2 nie jest ograniczone do świata IT. Kluczem do sukcesu w przypadku zastosowania tej metodyki jest właściwe jej dostosowanie do potrzeb chwili; niekoniecznie oznacza ona „planowanie na rok do przodu”.

Szczypta historii

W połowie lat 70. ubiegłego wieku prywatna firma Simpect Systems Limited opracowała metodykę Project Resource Organisation Management Plannig Technique (PROMPT). Pomysłem zainteresował się brytyjski rząd, który zlecił tej firmie wzbogacenie standardu o kwestie jakościowe, i w 1983 roku pod nazwą PROMPT II metodyka została oficjalnie wprowadzona w brytyjskiej administracji rządowej.

W 1989 roku brytyjska agenda rządowa Central Computer and Telecommunications Agency (CCTA) opublikowała standard pod nową nazwą — Prince (ang. *PProjects IN Controlled Environments*) — i wskazała jako zbiór najlepszych praktyk zarządzania projektami informatycznymi, ale z powodzeniem stosowano go również w innych obszarach.

W 1996 roku opublikowano metodykę Prince2, która była już niezależna od obszaru zastosowania biznesowego. Standard szybko zdobył bardzo dużą popularność głównie w Wielkiej Brytanii, ale również poza nią, i stał się bardzo ciekawą alternatywą dla metodyki PMBoK.

W 2005 roku *Office for Government Commerce* (OGC) — następca CCTA i obecny właściciel metodyki — opublikował zmiany do Prince2, wprowadzając dodatkowe, drobne usprawnienia. Należy zaznaczyć, że w skali całego świata za wszelkie kwestie związane z przygotowaniem autoryzowanych materiałów szkoleniowych, certyfikację oraz konsultacje i akredytacje odpowiedzialna jest firma APMGroup, która w swojej głównej siedzibie zatrudnia zaledwie 30 osób.



W praktyce

W Polsce metodyka Prince2 jest często wykorzystywana przez sektor publiczny, a przy niektórych przetargach publicznych pojawia się nawet wymóg, aby kierownik projektu posiadał certyfikat Prince2 Practitioner lub równoważny.

Procesy

Prince2 to zestaw ośmiu procesów, ośmiu komponentów (kluczowe „tricki” stosowane w procesach) i trzech technik (czyli istotnych strategii postępowania wykorzystujących komponenty we wszystkich procesach).

Nazwa polska *Uruchamianie Projektu/Przygotowanie Założeń Projektu (PP)*

Nazwa ang. *Starting Up a Project (SU)*

Opis Na podstawie zlecenia zewnętrznego lub wewnętrznego (mandatu) w postaci dokumentu Zlecenie Przygotowania Założeń Projektu (ZPZP) sformowany zostaje Komitet Sterujący, który nominuje Kierownika Projektu. W praktyce oznacza to stworzenie komponentu Organizacja w kontekście konkretnego projektu informatycznego. Kierownik Projektu jest odpowiedzialny za przygotowanie dokumentów wejściowych do pierwszej strategicznej decyzji (proces ZS1) o uruchomieniu (bądź nie) przygotowań do projektu. Jeżeli Komitet Sterujący podejmie pozytywną decyzję, spowoduje to zapoczątkowanie procesu Inicjowanie Projektu (IP).

Wejście Zlecenie Przygotowania Założeń Projektu (ZPZP) — (ang. *Project mandate*)

Wyjście Podstawowe Założenia Projektu (PZP), Formuła Realizacyjna (np. decyzja o tym, czy projekt będzie realizowany wewnętrznie, czy zewnętrznie), plan kolejnego etapu Inicjowania Projektu (IP).

Nazwa polska *Strategiczne Zarządzanie Projektem (ZS)*

Nazwa ang. *Directing a Project (DP)*

Opis Proces ten opisuje wszelkie strategiczne decyzje podejmowane przez Komitet Sterujący na wniosek Kierownika Projektu w różnych etapach projektu. Proces ten jest uruchamiany zawsze wtedy, gdy zakończy się etap, przed rozpoczęciem następnego LUB w sytuacjach kryzysowych, które wymagają eskalacji problemu (decyzja przekracza uprawnienia Kierownika Projektu).

Prince2 opisuje pojęcie przeglądu bramkowego (ang. *gate review*). Polega ono na niezależnej, zewnętrznej weryfikacji całego projektu i opinii, czy dany projekt powinien być kontynuowany, czy nie.

Wejście Dokumenty z poprzedniego etapu i plany następnego przygotowane przez Kierownika Projektu.

Wyjście Decyzja.

Nazwa polska *Inicjowanie Projektu (IP)*

Nazwa ang. *Initiating a Project (IP)*

Opis Strategiczne Zarządzanie Projektem zapewnia strategiczną decyzję o uruchomieniu przygotowania do projektu. Przed startem projektu należy jednak przygotować solidne uzasadnienie biznesowe oraz precyzyjną wizję projektu, jego podział na etapy, harmonogram prac, plan jakości, rejestr ryzyk itp.

Jednym z głównych celów tego procesu jest przygotowanie Dokumentu Inicjującego Projekt (DIP).

Wejście PP i zezwolenie komitetu sterującego (ZS1).

Wyjście Dokument Inicjujący Projekt (DIP) i dokumenty powiązane.

Nazwa polska *Sterowanie Etapem (SE)*

Nazwa ang. *Controlling a Stage (CS)*

Opis Realizacja całego projektu jest podzielona na etapy zarządcze, których nie należy mylić z etapami technicznymi. W szczególnym przypadku koniec etapu zarządczego może wydawać się „sztuczny” z punktu widzenia technicznego.

Na etapie zarządczym pełna odpowiedzialność jest delegowana na Kierownika Projektu (KP), który operuje w ramach założonych ograniczeń czasowych i budżetowych (progów tolerancji). Jeżeli KP zauważy, iż projekt przekracza lub wyraźnie zmierza do przekroczenia określonych progów tolerancji, zobligowany jest do sformułowania planu awaryjnego (z zamknięciem projektu włącznie) i przedstawienia go do akceptacji Komitetowi Sterującemu w celu podjęcia strategicznej decyzji wykraczającej poza uprawnienia Kierownika Projektu.

Proces ten jest również nadrzędny w stosunku do procesu Zarządzania Wytwarzaniem Produktów (WP).

Wejście	DIP i dokumenty wyjściowe z procesu IP lub poprzedniego etapu.
Wyjście	Zrealizowany zbiór pakietów prac i zaktualizowana dokumentacja projektowa.
Nazwa polska	<i>Zarządzanie Wytwarzaniem Produktów (WP)</i>
Nazwa ang.	<i>Managing Product Delivery (MP)</i>
Opis	<p>Kierownik Projektu może przekazywać zadania kierownikom zespołów poprzez mechanizm delegowania pakietów prac (ang. <i>work packages</i>) i ma to miejsce szczególnie w dużych zespołach projektowych lub relacji zleceniodawca-zleceniobiorca pomiędzy dwoma firmami.</p> <p>Jeżeli zleceniobiorca jest firmą zewnętrzną i nie wykorzystuje Prince2, za pomocą tego procesu może stosunkowo łatwo komunikować się ze zleceniodawcą, niezależnie od tego z jakiej metodyki korzysta wewnętrznie (może również nie stosować żadnej). Proces ten jest stosunkowo prosty — działa według mechanizmu <i>Akceptacja</i> (1), <i>Wykonanie</i> (2), <i>Dostarczenie</i> (3).</p>
Wejście	Autoryzacja wykonania pakietu prac (SE1).
Wyjście	Raport wykonanych prac lub odbiór pakietu prac.
Nazwa polska	<i>Zarządzanie Zakresem Etapu (ZE)</i>
Nazwa ang.	<i>Managing Stage Boundaries (SB)</i>
Opis	<p>Gdy etap projektu zmierza ku końcowi, konieczne jest przygotowanie planu do następnego etapu lub procesu Zamykania Projektu. Zadania te realizowane są w procesie Zarządzania Zakresem Etapu, który zawiera w sobie aktualizację planów projektu, uzasadnień biznesowych, rejestru ryzyk itp.</p> <p>Proces, oprócz tego aspektu, zawiera w sobie również elementy zarządzania kryzysowego, kiedy konieczne jest naniesienie zmian w stosunku do pierwotnego planu bazowego etapu (np. wtedy, gdy pojawia się wymóg dodania nowej funkcjonalności).</p>
Wejście	Bieżąca dokumentacja projektowa.
Wyjście	Raport zakończenia etapu i plan następnego etapu lub wsad do procesu Zamykania Projektu.
Nazwa polska	<i>Zamykanie Projektu (ZP)</i>
Nazwa ang.	<i>Closing a Project (CP)</i>
Opis	<p>W wielu projektach, szczególnie dużych, istnieje stały problem z jasnym określeniem momentu zakończenia (ang. <i>neverending story</i>). Proces Zamykania Projektu ma na celu odgórne narzucenie wymogu zamknięcia projektu i — jeśli to konieczne — przejście od klasycznych prac projektowych do typowych prac „utrzymaniowych” lub innych działań następczych (ang. <i>FollowUp Actions</i>).</p>

Proces ten jest również dobrą okazją do analizy porażki lub zdyskontowania sukcesu. W jego ramach następuje też zrewidowanie, czy zakładane korzyści biznesowe rzeczywiście miały miejsce.

W ramach tego procesu następuje odnotowywanie dobrych i złych doświadczeń przy realizacji projektów oraz stopniowe budowanie bazy wiedzy na ten temat w obrębie firmy (ang. *post-mortem analysis*). W metodyce Prince2 opisano mechanizm koleżeńskich przeglądów (ang. *peer review*), który polega na ocenie projektu przez niezależną osobę. Dzięki temu można uzyskać świeży punkt widzenia i rozpropagować „nauczki” projektowe w ramach całej organizacji.

Wejście Dokument Inicjujący Projekt (DIP) oraz bieżąca dokumentacja projektowa.

Wyjście Zarchiwizowana dokumentacja projektowa wraz z analizą rezultatów.

Nazwa polska *Planowanie (PL)*

Nazwa ang. *Planning (PL)*

Opis Planowanie to proces, który jest podrzędny w stosunku do procesów omawianych powyżej. To nie tylko przygotowanie harmonogramu, ale również szeregu innych elementów, takich jak zarządzanie konfiguracją, jakością i ryzykami.

Wejście Bieżąca dokumentacja projektowa.

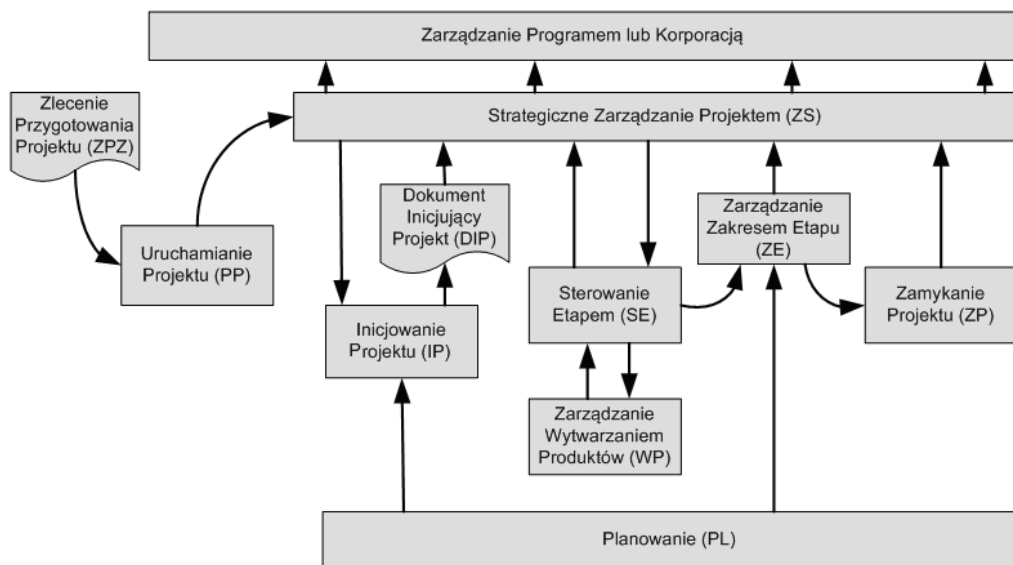
Wyjście Plan projektu lub etapu.

Z jednej strony, kluczem do zrozumienia metodyki Prince2 jest świadomość, że procesy nie następują po sobie sekwencyjnie i relacje między nimi są dość skomplikowane — rysunek 1. Z drugiej strony, każdy z procesów oddzielnie jest bardzo spójny wewnętrznie i łatwo zrozumieć jego cele, moment występowania oraz zaangażowane grupy osób. Prince2 zapewnia kompletny i spójny mechanizm wyzwalania kolejnych procesów, ale konieczna jest dogłębna znajomość wszystkich podprocesów, aby w pełni zrozumieć te zależności. Na bardzo ogólnym poziomie wygląda to mniej więcej tak...

Komponenty

Każdy z procesów wykorzystuje jeden lub kilka z wymienionych poniżej komponentów.

- ♦ **Uzasadnienie biznesowe** (ang. *Business Case*) — zdefiniowanie przyczyn uruchomienia projektu, którymi mogą być planowane zyski, oszczędności bądź inne korzyści, które firma planuje uzyskać.
- ♦ **Organizacja** (ang. *Organization*) — struktura organizacyjna projektu, która definiuje role osób w niego zaangażowanych. Prince2 narzuca pewne rozwiązania w tej kwestii.



Rysunek 1. Architektura procesów Prince2

Prince2 wymaga Komitetu Sterującego, który zajmuje się podejmowaniem strategicznych, okazjonalnych decyzji na temat projektu. Komitet Sterujący posiada przewodniczącego, który jest wspierany co najmniej przez reprezentanta użytkowników i wykonawcy. W najnowszych opracowaniach jest wzmianka o komitecie zarządzającym programem (zarządzie firmy), który jest organem nadrzędnym w stosunku do Komitetu Sterującego. Prince2 nie definiuje jednak tego organu ani w żaden sposób nie formalizuje współpracy między Komitetem Sterującym a komiteciem zarządzającym programem (zarządem firmy).

Kluczową rolę pełni Kierownik Projektu, który podejmuje wszystkie bieżące decyzje oraz zarządza projektem w ramach ustalonych etapów i progów tolerancji. Jeżeli pojawią się zagadnienia projektowe lub sytuacje kryzysowe, które przekraczają jego kompetencje, ma możliwość odwołania się do decyzji Komitetu Sterującego. W takim przypadku do dobrej praktyki należy przedstawienie sugerowanego planu naprawczego lub prośby o zamknięcie projektu.

Kierownicy zespołów (opcjonalni) odpowiadają przed Kierownikiem Projektu za pracę specjalizowanych, technicznych podzespołów.

- ♦ **Plany** (ang. *Plans*) — to produkty procesu Planowania, czyli dokumenty opisujące, co w projekcie lub na danym etapie powinno się osiągnąć, kiedy, jak i przy czyjej pomocy.
- ♦ **Elementy sterowania** (ang. *Controls*) — mechanizmy opisujące sposób kontroli projektu przez Kierownika Projektu i Komitet Sterujący. W skład tego komponentu wchodzi takie elementy jak punkty kontrolne w harmonogramie (ang. *milestones*), raportowanie oraz współczynnik tolerancji dla Kierownika Projektu. Proóg tolerancji jest negocjowany w trakcie uruchamiania projektu i definiuje, jak dużą decyzyjność posiada kierownik projektu. Progi tolerancji dotyczą głównie czasu i kosztów projektu, ale Prince2 opisuje również tolerancje

na zakres, ryzyko, zyski i jakość. Przykładowo Komitet Sterujący, nominując Kierownika Projektu, może dać mu pełną decyzyjność, o ile czas i budżet projektu nie przekroczą plus/minus 20% zaplanowanych wartości. Te 20% to właśnie próg tolerancji.

- ♦ **Zarządzanie ryzykiem** (ang. *Management of Risk*) — oznacza zidentyfikowanie wraz z całym zespołem możliwego ryzyka związanego z realizacją planów, zarejestrowanie ich w rejestrze ryzyk, a następnie właściwe nimi zarządzanie. Prince2 proponuje kilka sposobów postępowania:

- ♦ unikanie (np. inne rozwiązanie, w którym ryzyko nie wystąpi),
- ♦ minimalizację (np. przygotowanie studium wykonalności),
- ♦ przeniesienie (np. przekazanie ryzykownego zadania do podwykonawcy lub ubezpieczenie),
- ♦ akceptację i planowanie budżetu rezerwowego (zaplanowanie budżetu na wypadek wystąpienia ryzykownej sytuacji).

Każde zarejestrowane ryzyko ma przypisaną co najmniej:

- ♦ osobę odpowiedzialną za zarządzanie ryzykiem,
- ♦ właściciela ryzyka.

Przykładową kartę ryzyka zaprezentowano na rysunku 2.

- ♦ **Jakość w środowisku projektu** (ang. *Quality In Project Environment*) — mechanizmy gwarantujące, że projekt dostarczy produkty o odpowiedniej jakości. Prince2 nie zawiera konkretnych technik planowania testów ani ich wykonania. Przykładowo Prince2 nie wymienia typów testów ani nie definiuje sposobu wykonywania testów wydajnościowych. Metodyka ta koncentruje się raczej na wysokopoziomowych kwestiach jakościowych, takich jak konieczność wykonania planu testów, zdefiniowania miar jakościowych i stworzenia rejestru jakości. Pojawia się tutaj odwołanie do norm ISO oraz wewnętrznych, organizacyjnych polityk i systemów jakościowych. Najbardziej techniczną kwestią, jaką oferuje Prince2, są kryteria akceptacyjne i technika przeglądu jakości. Rezultatem tych przeglądów mogą być protokoły rozbieżności, które definiują różnice pomiędzy wytworzonymi produktami a wejściowymi dokumentami.
- ♦ **Zarządzanie konfiguracją** (ang. *Configuration Management*) — w trakcie realizacji projektu pojawia się wiele jego wersji, środowisk oraz możliwości konfiguracyjnych. Zarządzanie konfiguracją zawiera m.in.:
 - ♦ planowanie konfiguracji, czyli jaki poziom zarządzania konfiguracją będzie wymagany przez projekt,
 - ♦ specyfikację i identyfikację wszystkich komponentów końcowego rozwiązania,
 - ♦ kontrolowanie zmian w planie bazowym konfiguracji, zgodnie z mottem: „Nic nie ulegnie zmianie bez autoryzacji”,
 - ♦ zapamiętywanie i raportowanie danych związanych z każdym produktem,
 - ♦ audyty konfiguracji,

Tytuł	Brak pierwszego klienta biznesowego
Właściciel	Koszlajda Adam
Przydzielono do	Koszlajda Adam
Stan	(3) Zamknięte
Kategoria	(2) Kategoria 2
Data ukończenia	
Prawdopodobieństwo	90%
Oddziaływanie	8
Narażenie na ryzyko	7,2
Koszt	0,00 zł
Narażenie na ryzyko wyrażone w kosztach	0,00 zł
Opis	Brak rzeczywistych przypadków biznesowych (formularze, <input type="text"/> <div style="background-color: #cccccc; height: 50px; width: 100%;"></div>
Plan łagodzenia	Szybkie zaangażowanie potencjalnych klientów w proces tworzenia rozwiązania.
Plan awaryjny	Negocjacje z klientem.
Opis wyzwacza	Pierwszy klient na <input type="text"/>
Wyzwalacz	Inne
Łączy	
Utworzony o 2008-05-26 14:55 przez Koszlajda Adam Data ostatniej modyfikacji: 2008-07-09 15:47 , autor ostatniej modyfikacji: Koszlajda Adam <div style="float: right; border: 1px solid #ccc; padding: 2px 5px;">Zamknij</div>	

Rysunek 2. Przykładowa formatka ryzyka w MS Enterprise Project Management

- ♦ wersjonowanie kodu, rejestrów, dokumentacji i innych zmiennych elementów projektu,
- ♦ migrację rozwiązania ze starej wersji do kolejnej,
- ♦ współbieżne zarządzanie różnymi instalacjami systemu (takimi jak np. środowisko konstrukcyjne, testowe, produkcyjne),
- ♦ próby rozwiązywania błędów poprzez zmianę konfiguracji,
- ♦ kwestie administracyjne,
- ♦ kwestie bezpieczeństwa.
- ♦ **Sterowanie zmianami** (ang. *Change Control*) — sposób zarządzania propozycjami zmian w planie bazowym, czyli np. tak powszechny problem: „Co zrobić, gdy w trakcie realizacji ktoś proponuje »drobną« zmianę funkcjonalności?”. Mechanizm ten definiuje sposób rejestrowania zagadnień projektowych, a następnie ich analizy poprzez porównanie korzyści w stosunku do ryzyka, kosztu i czasu realizacji. W przypadku budżetu opisane są przykładowo mechanizmy związane z budżetem zmian. Finalna decyzja o zmianie może być podjęta przez Kierownika Projektu lub przedstawiona do decyzji Komitetowi Sterującemu.

Wujek Dobra Rada — odcinek 2. „Pewna jest tylko zmiana”



ŻADEN PROJEKT NIE POTOCZY SIĘ DOKŁADNIE TAK,
JAK ZOSTAŁO TO ZAPLANOWANE

Techniki

Techniki są pokrewne w stosunku do komponentów, jednak są bardziej ogólne i obejmują swoim zakresem cały czas trwania projektu, a nie konkretną grupę procesów. Ciekawą cechą technik jest to, że mogą być (i są) stosowane w oderwaniu od całej metodyki.

Planowanie oparte na produktach (ang. *Product Based Planning*) jest ekspresją kluczowej cechy całej metodyki — podejścia produktowego. Produkt występuje tu w rozumieniu „dostarczenia czegoś” i może być produktem zarządczym wynikającym z metodyki (np. Dokument Inicjujący Projekt) lub specjalistycznym (np. zaprojektowanie i wykonanie sieci LAN, instalacja, wykonanie oprogramowania).

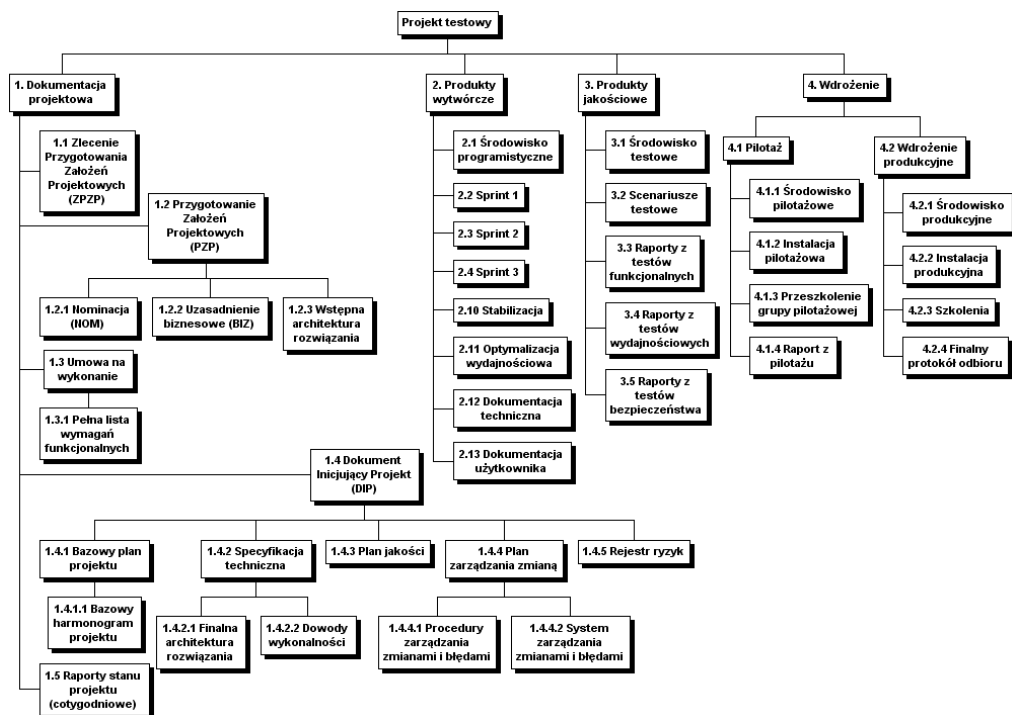
Prince2 definiuje konieczność formalnego opisu każdego ze zdefiniowanych produktów oraz rozplanowanie struktury produktowej (ang. *Product Breakdown Structure*).



W praktyce

W praktyce bardzo często pojęcie struktury produktowej, charakterystyczne dla Prince2, reprezentowane jest przez strukturę podziału prac (ang. *Work Breakdown Structure*).

Na rysunku 3. zaprezentowano przykładowy diagram WBS.



Rysunek 3. Przykładowa struktura prac (ang. *Work Breakdown Structure*)

Od strony praktycznej istotna jest wczesna definicja wszystkich produktów w projekcie, nadanie każdemu z nich unikalnego i niezmiennego identyfikatora oraz przydzielenie jednej osoby lub firmy, które będą za nie odpowiedzialne (np. poprzez odpowiednie pokolorowanie poszczególnych elementów). Zaprezentowany powyżej sposób numeracji umożliwi w trakcie realizacji projektu stosunkowo łatwe dodawanie nowych, dodatkowych produktów.

Po wykonaniu tego zadania pozostaje „jedynie” zadbanie:

- ♦ aby zrozumienie poszczególnych produktów było spójne i powszechne,
- ♦ by zakresy poszczególnych produktów nie nachodziły na siebie (redundacja pracy i potencjalne konflikty) lub nie rozchodziły się (niewykonane prace i syndrom „ziemi niczyjej”),
- ♦ o monitorowanie, czy prace toczą się zgodnie z harmonogramem i budżetem.

Warto zauważyć, że narzędzia, takie jak Microsoft Project, nie posiadają standardowo funkcjonalności tworzenia struktury podziału prac (WBS). Możliwości takie daje bardzo ciekawe narzędzie WBS Chart Pro 4.7, które kosztuje około 200 USD i posiada opcję integracji z MS Project.



W praktyce

Sterowanie zmianami (ang. *Change Control*) opisuje modelowy proces zarządzania zmianą oraz szablon rejestru zgłoszeń (ang. *Issue log*). Dodatkowo technika zawiera również konieczność wykonania analizy wpływu zmiany na projekt oraz kwestie związane z autoryzacją zmiany. Technika ta jest bardzo mocno związana z komponentem o tej samej nazwie i w wielu miejscach opisuje te same kwestie.

W bardzo wielu projektach, które nie stosują metodyki Prince2, zakłada się, że osoba zgłaszająca propozycję zmiany musi dostarczyć formalny dokument żądania zmiany (ang. *Change Request*). Dokument jest następnie analizowany od strony technicznej i wraz z adnotacjami przedstawiony do decyzji Kierownikowi Projektu lub Komitetowi Sterującemu. Przykładowy formularz może zawierać niżej wymienione pola.

1. Osoba inicjalizująca zmianę:

- ♦ Unikalny identyfikator — nadawany automatycznie,
- ♦ Projekt, którego żądanie dotyczy,
- ♦ Właściciel żądania,
- ♦ Krótki tytuł,
- ♦ Generalny opis,
- ♦ Priorytet żądania,
- ♦ Opis wymiernych korzyści z wprowadzenia zmiany,
- ♦ Opis, co się stanie, jeśli zmiana **nie zostanie zaimplementowana**.
- ♦ Osoba weryfikująca możliwość realizacji:
 - ♦ Plan wprowadzenia zmiany — jakie czynności muszą zostać wykonane, aby zmianę można było wprowadzić,
 - ♦ Koszt wprowadzenia zmiany — idealnie, gdy można ten koszt wyrazić finansowo,
 - ♦ Horyzont czasowy realizacji zmiany — kiedy mogłaby być wprowadzona produkcyjnie,
 - ♦ Ryzyka wprowadzenia zmiany — np. wpływ na już istniejącą funkcjonalność,
 - ♦ Inne możliwości realizacyjne — opis innego sposobu lub sposobów realizacji zmiany.

- ♦ Osoba decyzyjna:
 - ♦ Decyzja,
 - ♦ Uzasadnienie.
- ♦ Osoba realizująca:
 - ♦ Opis rozwiązania.



W praktyce

Przegląd jakości (ang. *Quality Reviews*) opisuje sposób zapewnienia jakości w projekcie poprzez dokonywane przeglądy jakości, czyli weryfikację, czy zrealizowany produkt jest zgodny z opisem przygotowanym w fazie planowania. Technika zakłada sesje weryfikacyjne i konieczność sporządzenia protokołu rozbieżności, który będzie zawierał wykryte błędy, braki lub nadmiar funkcjonalności.

Zarządzanie dokumentacją

W trakcie realizacji projektu zgodnie z metodyką Prince2 powstaje zazwyczaj dość duża ilość dokumentacji i warto zadbać o wsparcie ze strony biura zarządzania projektami (ang. *Project Management Office*). Jego zadaniem będzie pomoc w szeregu prac biurowych, standaryzacyjnych, organizacyjnych, archiwizacyjnych itp.

Metodyka bazuje na trzech głównych folderach:

- ♦ projektu,
- ♦ etapu,
- ♦ jakości.

W ramach poszczególnych procesów foldery te są wypełniane konkretnymi dokumentacjami, takimi jak rejestr ryzyka, rejestr zagadnień projektowych (propozycji zmian), plan projektu, plan testów itp.



W praktyce

O ile większość dokumentacji może być powszechnie dostępna, o tyle należy zwrócić szczególną uwagę, aby do danych finansowych miały dostęp jedynie wybrane osoby.

Dostosowywanie do potrzeb organizacji

W ostatnich publikacjach o Prince2 znaleźć można informacje o wielu opcjach dostosowania metodyki do skali projektu. W wersji minimalnej:

- ♦ nadal należy stworzyć dobre uzasadnienie biznesowe i odpowiedzieć na pytanie, czemu projekt jest uruchamiany,
- ♦ wymagane są tylko dwa procesy — analityczny (*Inicjowanie Projektu*) i wykonawczy (*Sterowanie Etapem*),
- ♦ Kierownik Projektu jest osobiście odpowiedzialny za realizację całego projektu (nie ma kierowników podzespołów).



W praktyce

Niestety, w praktyce bardzo często zdarza się, że firmy chwalą się realizacją projektów zgodnie z metodyką Prince2, podczas gdy w rzeczywistości wyżej wymienione procesy, komponenty i techniki nie są stosowane lub wykorzystywane są szczątkowo. Mówi się wtedy o syndromie PINO, czyli „Prince tylko z nazwy” (ang. *Prince In Name Only*). Bardzo dobrym testem na weryfikację stanu rzeczy jest sprawdzenie, jak często Kierownik Projektu spotyka się z Komitetem Sterującym. Czy dla każdego projektu istnieje Komitet Sterujący? Jaki jest jego stopień zaangażowania w projekt? Czy każdy projekt posiada stosowny, aktualizowany plan i czy jego realizacja jest monitorowana?

Prince2 prawie nigdy nie powinien być stosowany do wszystkich projektów w firmie, a jedynie do tych, które wymagają bardzo dużej formalności podejmowanych decyzji (np. ze względu na możliwe audyty). Należy bowiem pamiętać, że spora część czasu Kierownika Projektu będzie przeznaczona na zbieranie, weryfikowanie i przygotowywanie „materiału dowodowego”.

Certyfikacja

Za kwestię szkoleń i certyfikacji odpowiedzialna jest organizacja APM Group, która poprzez sieć akredytowanych partnerów oferuje na całym świecie dwa poziomy certyfikacji z wiedzy na temat Prince2 w dziewięciu językach, także w języku polskim.

Prince2 Foundation jest godzinny test składający się z siedemdziesięciu pięciu pytań wielokrotnego wyboru, bez możliwości korzystania z materiałów, gdzie 50% poprawnych odpowiedzi wystarczy do zaliczenia. Test ten jest zdawany przez 99% uczestników. Uzyskany certyfikat pozwala na uczestnictwo w projektach prowadzonych zgodnie z metodyką.

Prince2 Practitioner¹ jest trzygodzinnym testem składającym się z dziewięciu zestawów pytań bazujących na dziewięciu scenariuszach, z możliwością korzystania z oficjalnych podręczników Prince2, gdzie 50% poprawnych odpowiedzi wystarczy do zaliczenia. Test ten jest zdawany przez 75 – 80% kandydatów. Uzyskany certyfikat pozwala na prowadzenie projektów zgodnie z metodyką Prince2.

Oba certyfikaty są bezterminowe.

W Polsce powszechnie znany jest kilkudniowy kurs w jednej z dwóch certyfikowanych firm szkoleniowych: CRM (pierwsza firma na rynku oferująca te szkolenia od wielu lat) lub Infovide-Matrix. Firmy te nie zezwalają na podejście do egzaminów bez szkoleń. Aby podejść do egzaminu Practitioner, należy posiadać zdany egzamin Foundation. Koszt takiego przedsięwzięcia w lutym 2008 roku wynosił:

- ♦ od 4000 do 5000 zł za szkolenie Prince2 Foundation wraz z oficjalnymi podręcznikami w języku polskim,
- ♦ od 1000 do 1500 zł za egzamin Prince2 Foundation,
- ♦ od 4000 do 5000 zł za szkolenie i egzamin Prince Practitioner.



W praktyce

Należy nadmienić, że APM Group nie wymaga przejścia certyfikowanego szkolenia i można bezpośrednio w tej organizacji zamówić książki, a następnie zarejestrować się na sam egzamin w British Council, który oficjalnie udostępnia tego typu usługi w wielu krajach, a w tym i w Polsce. Istotną kwestią w tym przypadku jest jednak bardzo jasne zaznaczenie języka, w którym zdawany będzie egzamin, ponieważ konieczna jest znajomość dokładnej nazwy kluczowych terminów w konkretnym języku.

Brytyjska strona APM Group oferuje bardzo ciekawy serwis, gdzie bez konieczności logowania można sprawdzić, czy konkretna osoba posiada jeden z wymienionych wyżej certyfikatów (ang. *Successful Candidate Register*).

W kwietniu 2008 roku na całym świecie było około 60 tysięcy osób z certyfikatem Prince2 Practitioner.

Podsumowanie

Prince2 koncentruje się na „logistycznym” aspekcie projektu i nie wnika w kwestie techniczne. Procesy stoją na straży jakości, harmonogramu i budżetu. Kluczową kwestią jest właściwe dostosowanie metodyki do skali projektu. Prince2 koncentruje uwagę zaangażowanych osób na słowie pisanym — w sytuacji gdy mamy do czynienia ze współpracą

¹ Od 1 września 2007 zaprzestano otwartej, esejowej formy tych egzaminów.

wielu firm, kontraktami o stałej cenie oraz wysokim stopniu formalizmu, wykorzystanie takiej metodyki jest dobrym wyborem. Jeżeli jednak cały projekt realizowany jest w obrębie jednej firmy i nie jest wymagane dokumentowanie każdej decyzji, warto zastanowić się nad inną metodyką lub minimalną wersją Prince2. Jeżeli mamy do czynienia z krótkim jednomiesięcznym przedsięwzięciem, najprawdopodobniej nie będzie konieczny ZPZP i PZP — wystarczy wykonać sam dokument DIP, a po zakończeniu prac napisać jeden raport.

W podrozdziale „Przykład Prince2 i RUP — BlogSerwis” opisany został przykład projektu realizowanego zgodnie z metodykami Prince2 i RUP.

Rozmowa z...

Tomasz Koniecko to kierownik projektu z dwunastoletnim doświadczeniem w dostarczaniu projektów dla międzynarodowych i polskich firm, np. QAD, PWPW i S&T. Przewodził dużym zespołom ludzkim w trakcie kompleksowych projektów IT bazujących na Prince2, takich jak ePoltax, i uczestniczył w pracach wdrażających wewnętrzne standardy korporacyjne oraz procedury jakościowe.

Adam Koszlańda (AKO): Tomku, jesteś kierownikiem projektów, który już od kilku lat stosuje metodykę Prince2. W jakich projektach ją stosowałeś?

Tomasz Koniecko (TKO): Prince2 z moim udziałem stosowany był w projekcie e-Poltax, realizowanym dla Ministerstwa Finansów, i obecnie w projekcie dla Ministerstwa Środowiska oraz kilku innych w formie mniej okazałej.

AKO: Dlaczego akurat Prince2?

TKO: No cóż, w dwóch wymienionych przypadkach był to poniekąd wymóg zamawiającego.

AKO: Czy oznacza to, że stosujesz Prince tylko wtedy, kiedy musisz?

TKO: Nie do końca. Sposoby stosowania Prince wytrenowałem w boju. Mogę go bardzo szybko dostosować do potrzeb każdego projektu. Ogólnie metodyka to pewna formalizacja działań, która ma zagwarantować poprawną realizację.

AKO: A nie jest tak, że metodyki te bardzo dobrze nadają się do formalizowania relacji zlecającego-zlecaniobiorcy?

TKO: Tak. Dlatego metodyką powinien być zainteresowany klient, wykonawca ma się w nią wpisać. Lata całe ewangelizowałem klientów, że projekt jest ich własnością. To oni mają lub powinni mieć uzasadnienie biznesowe do powołania projektu i wynikające z tego konsekwencje. Na marginesie, dostawcy mają manię, by ogłaszać, że mają własną metodykę i ją stosują, a na potwierdzenie tego potrzebne są referencje. Dociekliwy badacz dotarłby zapewne do projektów, gdzie jedynie w procesie poszlakowym można by wskazać, jaką metodyką były prowadzone ;).

AKO: W Prince2 nazywa się to syndromem PINO (ang. *Prince In Name Only*).

TKO: Ciekawe, jak syndrom ten się nazywa w metodyce PMBoK?

AKO: Też jestem ciekaw ;).

TKO: Sądę, że PINO.

AKO: *PMBoK IN Name Only* :D? Czy nie jest jednak tak, że więcej firm stosuje PMBoK niż Prince, nawet jeśli ministerstwa chętniej wybierają tę drugą metodykę?

TKO: Nie znam statystyk, które potwierdziłyby to jednoznacznie. Utarło się przekonanie, że jeszcze nikomu nie udowodniono winy, jeśli zastosował Prince2 umiejętnie, ale jeśli dokumentacja ma przez kolejne kilka lat opierać się każdej kontroli, trzeba wiedzieć, jak ją zrobić. W firmach komercyjnych środek ciężkości jest ulokowany nieco inaczej. Przez lata wdrażałem w QAD różne rozwiązania według metodyki Q-Advanatage wykonanej na wzór PMBoK, później Prince2 i — szczerze mówiąc — nie bardzo widzę różnice w zastosowaniu. PMBoK stawia w punkcie centralnym kierownika projektu i jako zbiór wiedzy daje mu narzędzia, zaś Prince2 ukierunkowany jest na procesy, które obejmują całość organizacji.

AKO: Słyszałem opinie, że Prince2 jest nieco bardziej sztywny...

TKO: Mój szanowny kolega Roman Bratek — wykładowca Prince — twierdzi, że sztywność tej metodologii jest stereotypem wśród ludzi, którzy nie potrafią jej zastosować.

AKO: A jak ma się to do rosnącej liczby certyfikacji PMBoK? Czy nie jest tak, że właśnie ta metodyka jest stosowana w sektorze prywatnym?

TKO: No cóż... APM Group notuje w ostatnim roku znaczący wzrost certyfikacji Prince2 w Chinach i USA. W moim przypadku S&T aktualnie stosuje w projektach SAP-owych metodykę PROMET, która jest zbudowana na wzór Prince2 i wyposażona w szereg dodatkowych narzędzi.

AKO: Wygląda więc na to, że udana implementacja metodyki w danej firmie musi uzyskać swoją nazwę ;).

TKO: Trochę tak jest, jeśli wartość dodana jest wystarczająco wyraźna.

AKO: A gdybyś pokusił się o porównanie PMBoK-owego Q-Adavantage z firmy QAD z PROMETEM wzorowanym na Prince z S&T?

TKO: Porównanie metodyk Q-Advanatage i PROMET to temat na doktorat :D i być może świetnym przyczynkiem do dyskusji nad metametodyką łączącą podejście PMBoK z Prince2.

AKO: To z innej beczki. Zadam takie prowokacyjne pytanie... Jak klient sobie nie życzy żadnej metodyki? To hulaj dusza, piekła nie ma?

TKO: Nie spotkałem się z przypadkiem, aby sobie nie życzył zastosowania jakiegś metodyki, częściej klient nie identyfikuje metodyki jako przydatnego mu narzędzia. Jeżeli już pojawia się jakiś problem, to raczej w trakcie realizacji projektu.

Czasami okazuje się np. że klient wstępnie deklarował pełne zaangażowanie, a później nie ma chęci włożyć się w procedury metodyki lub chce realizować projekt zgodnie z metodyką, ale jej po prostu nie zna.

AKO: I w takim przypadku trzeba wysłać taką osobę na szkolenie z Prince...

TKO: Nie spotkałem się z takim przypadkiem, ale słyszałem, że czasami może to przynieść trudne do przewidzenia skutki.

AKO: To znaczy?

TKO: Niektóre osoby, które wracają z takiego szkolenia zrealizowanego w trakcie projektu, traktują pewne sprawy bardzo kategorycznie i okres „wygrzewania” świeżego kierownika spada na środek projektu, generując niepotrzebne ryzyko. Jest to jednak chyba bardziej sprawa personalna, niezwiązana z jakąś metodyką. Podejrzewam, że w PMBoK byłoby tak samo.

AKO: A gdybyś miał dać jakąś przestrozę osobie, która właśnie wróciła ze szkolenia i przygotowuje się do realizacji projektu w zgodzie z Prince2?

TKO: Szczerze mówiąc, nie uważam, że powinienem ostrzegać przed Prince2 czy inną metodyką, wszystko zależy od rozsądku, z jakim się do tego podchodzi, ale... najważniejsze są relacje z ludźmi wewnątrz organizacji i z dostawcą. Nie wierzę w skuteczną realizację projektu, jeśli tylko jedna ze stron będzie chciała współpracować. To jest jednak sprawa niezależna od metodyki, która może tyle samo popsuć, co naprawić.

AKO: To po co je w ogóle stosować?

TKO: Jest to język porozumiewania się w okolicznościach projektowych. Język, który ma być zrozumiały dla obu stron. Umiejętność przewidzenia trudnych zagadnień i sposób ich obsługi jest domeną kierownika projektu i w tym ma mu pomóc metodyka, bo wiadomo, że „ekskreментy trafiają czasem w wentylator”.

AKO: Czyli że metodyka to pewne ramy, pole do działania dla kierownika projektu?

TKO: Metodyka to narzędzie dla kierownika projektu, który — sięgając po różne metody — ma osiągnąć zamierzony cel. Trawestując powiedzenie estradowe — „Kierownik projektu jest jak biustonosz, niby niepotrzebny, ale podtrzymuje całość” ;).

Rozdział 2.

Project Management Body of Knowledge — PMBoK

Podejście PMBoK jest często przedstawiane jako „metodyka PMI”, czyli metodyka organizacji Project Management Institute zrzeszającej specjalistów z dziedziny zarządzania projektami. PMBoK koncentruje się na zebraniu i przedstawieniu dobrych praktyk związanych z zarządzaniem projektami w ramach zdefiniowanych obszarów wiedzy.

W pewnym sensie PMBoK jest podejściem konkurencyjnym w stosunku do Prince2 i ze względu na nieco większą swobodę implementacji jest częściej stosowany przez duże korporacje z sektora prywatnego. Dodatkowo PMBoK wkracza w obszary, których Prince2 nie obejmuje, takie jak zarządzanie zasobami ludzkimi oraz zaopatrzeniem.

PMBoK w wersji czwartej definiuje pięć grup procesów, takich jak procesy rozpoczęcia, planowania, realizacji, kontroli i zakończenia. Każdy z tych procesów należy również do jednego z dziewięciu obszarów wiedzy, takich jak zarządzanie integralnością projektu, zakresem, czasem, kosztami, jakością, zasobami ludzkimi, komunikacją, ryzykiem i zaopatrzeniem.

Szczypta historii

Organizacja PMI powstała w USA w 1969 roku jako organizacja non profit zrzeszająca profesjonalistów różnych specjalności w celu wyspecyfikowania standardowych praktyk zarządczych. W 1987 roku opublikowana została pierwsza edycja PMBoK, która była rezultatem prac wykonanych we wczesnych latach 80. W roku 1998 American National Standards Institute (ANSI) zaakceptował to rozwiązanie jako narodowy standard zarządzania projektami, a Institute of Electrical and Electronics Engineers (IEEE) zaadaptował

to podejście jako standard 1490¹. Od tej pory, co około 4 lata pojawiają się kolejne aktualizacje tej metodyki (w latach 1996, 2000 i 2004). Ostatnia, czwarta edycja ujrzała światło dzienne 31 grudnia 2008 roku.

Wersja czwarta nie zawiera rewolucyjnych zmian w stosunku do wersji trzeciej, ale można zauważyć kilka pozytywnych zmian ewolucyjnych. Oto one.

- ◆ Lepsze zarządzanie interesariuszami, które pojawia się już w grupie procesów rozpoczęcia jako nowy proces o nazwie „10.1. Identyfikacja interesariuszy”.
- ◆ Z grupy procesów rozpoczęcia zniknął proces „4.3. Opracowanie wstępnego zakresu projektu” (ang. *Develop preliminary scope statement*). Pokrywał się on z procesem „5.1. Planowanie zarządzania zakresem projektu”, który w PMBoK4 nazywa się już „5.1. Planowanie zarządzania zakresem projektu”. Nie jest to tylko zmiana nazwy, ale przejaw bardziej dojrzałego podejścia do zarządzania wymaganiami projektowymi. Pojawia się tu ciekawa technika „Matrycy śledzenia wymagań” (ang. *Requirements Traceability Matrix*).
- ◆ Unifikacja pewnych elementów przekazywanych pomiędzy procesami. Przykładowo pojawia się jeden, kluczowy plan zarządzania projektem zamiast oddzielnych planów do zarządzania poszczególnymi obszarami wiedzy (np. plan zarządzania zakresem, harmonogramem, kosztem itd.). Analogicznie pojawiło się bardziej ogólne żądanie zmiany, które zawiera w sobie rekomendowane działania naprawcze, prewencyjne i naprawę defektów. Tego typu podejście jest o wiele bardziej praktyczne i umożliwia większą swobodę w implementacji tych mechanizmów.
- ◆ Znacznemu uproszczeniu i generalizacji uległ obszar wiedzy dostaw. PMBoK4 przyjął tutaj nowy model **Planowanie>Wykonanie>Administrowanie>Zamknięcie**. Dodatkowo wyspecyfikowane zostały nowe podtypy kontraktów o ustalonej cenie. Nie wiadomo jednak, czy na polskim rynku będą one miały znaczenie praktyczne.
- ◆ Technika Uzyskanej Wartości (ang. *Earned Value Technique* — EVT), która była częścią techniki „analizy miar wydajnościowych” w procesie „7.3. Kontrola kosztów”, stała się pełnoprawną techniką Zarządzania Uzyskaną Wartością (ang. *Earned Value Management* — EVM). Technika ta uległa również pewnemu rozwinięciu i pojawił się nowy „indeks wydajności niezbędnej do zakończenia projektu” (ang. *To-Complete Performance Index* — TCPI).
- ◆ Uporządkowaniu uległo nazewnictwo wszystkich procesów oraz ich numeracja, która bazuje już wyłącznie na numerach rozdziałów i podrozdziałów.

¹ Standard IEEE Std 1490-1998 został zaktualizowany w 2004 (!) roku do IEEE Std 1490-2003.

Obszary wiedzy

PMBoK składa się z czterdziestu dwóch procesów, z których każdy przynależy do jednej z pięciu grup procesów i jednego z dziewięciu obszarów wiedzy. Każdy z procesów posiada numer główny od 4. do 12., który wskazuje określony obszar wiedzy², i poboczny numer porządkowy (np. proces 5.2 wskazuje drugi proces z obszaru wiedzy o nazwie „Zakres” opisanego w 5. rozdziale). Oto poszczególne obszary wiedzy.

Obszar wiedzy

Integracja (rozdział 4.)

Ten obszar wiedzy jest odpowiedzialny za ogólne, wysokopoziomowe kwestie zarządcze związane z realizacją projektu informatycznego, a szczególnie za:

- ♦ kwestie związane z uruchomieniem projektu (np. zdobycie mandatu na realizację projektu) — 4.1,
- ♦ przygotowanie planu zarządzania projektem — 4.2,
- ♦ zarządzanie bieżącymi pracami projektowymi — 4.3,
- ♦ kontrolę prac projektowych i zintegrowane zarządzanie zmianą — 4.4, 4.5,
- ♦ zamknięcie projektu — 4.6.

Integracja to codzienne wybory miejsca koncentracji zasobów i wysiłku, wyprzedzenie możliwych kłopotów i zarządzanie nimi, zanim staną się krytyczne.



— Integrować się! Integrować — rzecze Najlepszy Kierownik Projektu.

² Numery obszarów wiedzy są związane z numerami rozdziałów w oficjalnych podręcznikach PMBoK.

Obszar wiedzy Zakres (rozdział 5.)

Ten obszar wiedzy skupia się na zarządzaniu zakresem funkcjonalności projektu, a szczególnie na tworzeniu:

- ♦ definicji zakresu projektu wraz z strukturą wytwarzanych produktów (ang. *Work Breakdown Structure*) — 5.1, 5.2, 5.3,
- ♦ sformalizowanych mechanizmów weryfikacji i kontroli zakresu projektu — 5.4, 5.5.

Ten obszar wiedzy PMBoK łączy w sobie tematy, którymi zajmują się procesy Zarządzanie Zakresem Etapu (ZE) i Zarządzanie Wytwarzaniem Produktów (WP) w Prince2.



W praktyce

Inżynierowie z firmy produkującej auta otrzymali plany nowego, eksportowego modelu auta i w trakcie analizy zauważyli wymóg konieczności zamontowania tylnych szyb odpornych na prędkość 50 km/h! 50 km/h? Przecież na biegu wstecznym auto nigdy nie osiągnie takiej prędkości! Zgodnie stwierdzono więc, że w celu redukcji kosztów zamontowane zostaną inne szyby o gorszych parametrach.

Inżynierowie pracowali w pocie czoła przez wiele długich miesięcy i to niejedną raz po godzinach. Jak każdy projekt, ten też miał swoje dobre i złe chwile, ale w końcu udało się skonstruować pierwszy prototyp, który pomyślnie przeszedł pierwszą serię testów terenowych. Podjęto decyzję o uruchomieniu produkcji i wyprodukowano pierwszą setkę pięknych, czerwonych, lśniących nowych aut; chłopcy pana Stefana przez cały dzień polewowali je na parkingu firmowym. Z powodzeniem zamknięto projekt i wypłacono premie, a po tygodniu zadzwonił telefon z zagranicy...

— Co wyście zrobili!!!!

— Nowe auta.

— Wszystkie tylne szyby powybijane! Co my teraz mamy zrobić?

— Jak to powybijane!?

— Właśnie ściągnęliśmy je z lawety kolejowej! Będziemy musieli jakoś załatwić tę sprawę u nas, lokalnie... klienci czekają...

— Zaraz, zaraz... Transport kolejowy!?

— Tak! Przecież pisaliśmy, że tylne szyby mają wytrzymywać szybkość 50km/h.

— No właśnie. Po co?

— Po co ???!!! Przecież te auta są ustawiane na wagonach tylną szybą do przodu!!! Wystarczyło TYLKO wykonać to, co zapisaaliśmy! Hrrrr...

Obszar wiedzy Czas (rozdział 6.)

Ten obszar wiedzy to wszystkie działania związane z wykonaniem projektu w założonym terminie, o ile zmianie nie uległy przyjęte założenia oraz zakres. Szczegółowo przyjmuje on prostą sekwencję zdarzeń:

- ♦ zdefiniowanie zbioru planowanych działań — 6.1,
- ♦ ustanowienie wzajemnych zależności czasowych pomiędzy tymi działaniami (co musi być zrobione najpierw, a co potem, jakie działania mogą być wykonywane równocześnie) — 6.2,
- ♦ oszacowanie potrzeb zasobowych i czasu trwania poszczególnych działań — 6.3, 6.4,
- ♦ stworzenie harmonogramu projektu oraz jego kontrola — 6.5, 6.6.

Wszystkie te procesy z wyjątkiem ostatniego należą do grupy procesów planistycznych.

Obszar wiedzy Koszt (rozdział 7.)

Projekt informatyczny jest inwestycją, czyli kosztami, które w dłuższej perspektywie mają przynieść organizacji zysk. Aby projekt zakończył się powodzeniem, konieczne jest właściwie zarządzanie budżetem, czyli:

- ♦ szacowanie — 7.1,
- ♦ zaplanowanie (stworzenie bazowej wersji budżetu) — 7.2,
- ♦ kontrola — 7.3.

Wszystkie te procesy z wyjątkiem ostatniego należą do grupy procesów planistycznych.

Obszar wiedzy Jakość (rozdział 8.)

PMBoK proponuje następujące procesy w celu zapewnienia właściwego zarządzania jakością:

- ♦ zaplanowanie sposobu zapewniania odpowiedniej jakości w projekcie — 8.1,
- ♦ wdrożenie tego planu poprzez systematyczne wykonanie rutynowych czynności — 8.2,
- ♦ kontrolę mechanizmów zapewnienia jakości — 8.3.

Obszar wiedzy Zasoby ludzkie (rozdział 9.)

Ten obszar wiedzy opisuje szereg dobrych praktyk związanych z zarządzaniem ludźmi postrzeganymi jako pojedyncze osoby i zespoły. Oznacza to:

- ♦ zaplanowanie potrzeb zasobowych — 9.1,
- ♦ procesy tworzenia zespołów ludzkich, ich rozwój i zarządzanie nimi — 9.2, 9.3, 9.4.



W praktyce

Rekrutacja właściwych osób to jeden z najważniejszych i najtrudniejszych procesów w trakcie przygotowania do uruchamiania projektu. Jednym z ciekawszych artykułów na ten temat jest przetłumaczony na język polski *Partyzancki poradnik rekrutacji* Joela Spolsky'ego³. Jego głównym przesłaniem jest rada, by rekrutować *tylko i wyłącznie* osoby bystre i realizujące cele.

Przy okazji „tematyki kadrowej” warto również nadmienić o niezwiązanym z PMBoK „procesie efektywności zespołowej” Kena Blancharda, który opisuje cztery poziomy gotowości pracowników.

- ♦ R1 — pracownik o niskich kompetencjach, który na swoim koncie nie ma większych sukcesów i dlatego nie jest zmotywowany do pracy.
- ♦ R2 — pracownik o niskich kompetencjach, który nie może samodzielnie wykonywać zadań, ale jest bardzo zmotywowany do ich wykonania.
- ♦ R3 — pracownik o dużych kompetencjach, który może samodzielnie wykonać zadania, ale brakuje mu motywacji z powodu braku we własne siły lub znużenia.
- ♦ R4 — pracownik o wysokich kompetencjach i chęciach, który potrafi i chce samodzielnie wykonywać zadania.

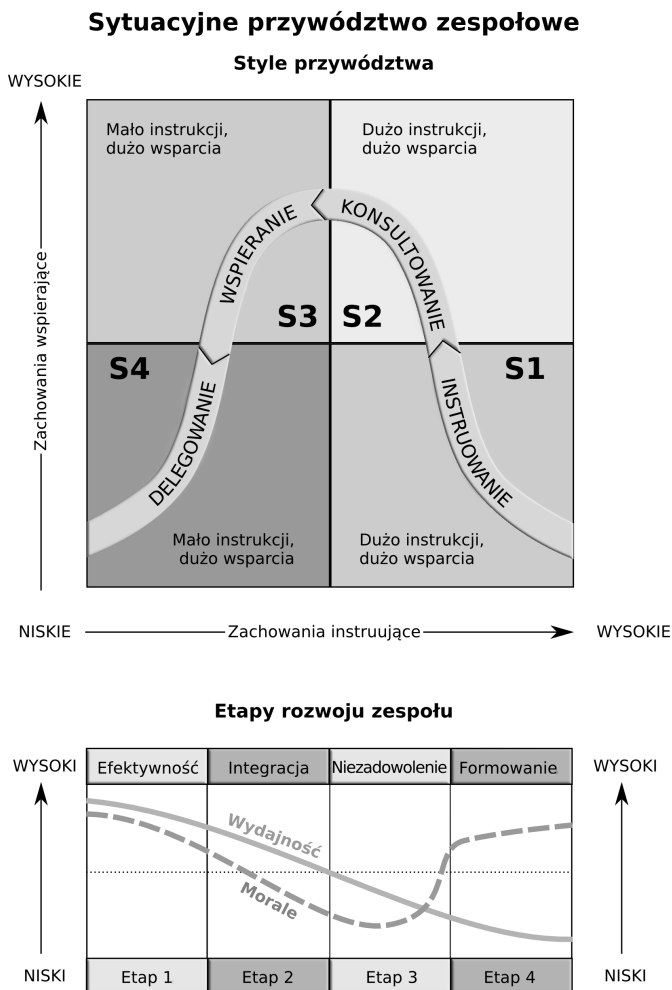
Ken Blanchard opisuje również cztery style przywództwa (rysunek 4).

- ♦ S1 — **instruowanie** to „suche” przekazanie małych, częściowych zadań do wykonania i rozliczenie z ich wykonania.
- ♦ S2 — **konsultowanie** to również przekazanie zadań, ale skoncentrowane na utrzymaniu wysokiej motywacji pracownika.
- ♦ S3 — **wspieranie** koncentruje się na właściwym zmotywowaniu pracownika, który sam wie, jakie zadania należy wykonać.
- ♦ S4 — **delegowanie** to styl, w którym pracownik wie, co należy zrobić, i jest zmotywowany do samodzielnego podjęcia odpowiedzialności.

Technika przywództwa sytuacyjnego koncentruje się na właściwym dopasowaniu poziomu gotowości do stylu przywództwa, ponieważ nie można jednej miary przykładać do wszystkich osób. Jak łatwo się domyślić, delegowanie złożonych zadań pracownikom o niskich

³ <http://polish.joelonsoftware.com/Articles/Interviewing.html>

Rysunek 4.
Model przywództwa
zespołowego
Kena Blancharda



Źródło: Blanchard International Polska - <http://www.blanchard.pl>

kompetencjach doprowadzi do katastrofy, a szczegółowe instruowanie doświadczonych pracowników demotywuje ich do pracy. Model ten koncentruje się również na rozwoju każdego pracownika i zwiększeniu jego poziomu gotowości.

Obszar wiedzy Komunikacja (rozdział 10.)

Ten obszar wiedzy koncentruje się na zapewnieniu właściwej komunikacji z interesariuszami projektu. Szczegółowo oznacza to:

- ♦ identyfikację kluczowych interesariuszy w trakcie inicjacji projektu — 10.2,
- ♦ stworzenie planu komunikacji z interesariuszami projektu — 10.2,

- ♦ właściwą dystrybucję informacji i zarządzanie interesariuszami — 10.3, 10.4,
- ♦ przygotowanie i dystrybucję kontrolnych raportów wydajnościowych — 10.5.

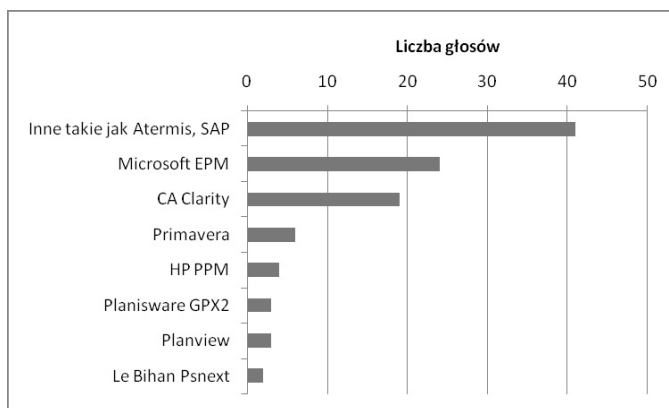


W praktyce

Należy zauważyć, że w dzisiejszych czasach dobra komunikacja nie jest możliwa bez właściwych systemów informatycznych. Bardzo wskazane jest posiadanie kompleksowego rozwiązania intranetowego, które umożliwi współdzielenie wiedzy projektowej wewnątrz firmy (ang. *Enterprise Project Management*). Każda z firm wybiera system według własnych potrzeb, a popularność poszczególnych rozwiązań jest dość rozproszona, co zobrazowano na rysunku 5.

Rysunek 5.

Wynik ankiety
„Jakich systemów
EPM używasz?”
Pierra-Jeana
Cherreta⁴



Inną ciekawą alternatywą dla rozwiązań tego typu jest firmowe wiki rozwijane na bazie rozwiązań darmowych, takich jak MediaWiki, na której bazuje Wikipedia, lub rozwiązań odpłatnych, np. Confluence.

W rozproszonych zespołach warto dodatkowo zainwestować w narzędzia współpracy zdalnej (ang. *collaboration tools*), takie jak WebEx⁵, GoToMeeting, MS Office Live Meeting (poprzednio PlaceWare) lub Adobe Connect (poprzednio Breeze).

Obszar wiedzy Ryzyko (rozdział 11.)

W tym obszarze zarządzanie ryzykiem projektowym odbywa się przy użyciu rejestru ryzyk. Działania te polegają na:

⁴ Na podstawie ankiety „Jakich systemów EPM używasz?” uruchomionej przez Pierra-Jeana Cherreta w serwisie społecznościowym Plaxo.

⁵ O skali tego rynku niech świadczy zakup, jakiego dokonało Cisco 15 marca 2007 roku, które za „drobne” 3,2 miliarda dolarów przejęło firmę WebEx.

- ♦ stworzeniu planu zarządzania ryzykiem — 11.1,
- ♦ identyfikacji, analizie i planowaniu odpowiedzi na ryzyka — 11.2, 11.3, 11.4, 11.5,
- ♦ monitorowaniu i kontrolowaniu ryzyk projektowych — 11.6.

Wszystkie te procesy z wyjątkiem ostatniego należą do grupy procesów planistycznych.

Obszar wiedzy

Dostawa (rozdział 12.)

Dostawa to zakup lub zdobycie produktów i usług spoza zespołu projektowego. Zarządzanie dostawą to:

- ♦ planowanie dostaw — 12.1,
- ♦ realizacja dostaw — 12.2,
- ♦ kontrola sposobu i stopnia realizacji dostaw — 12.3,
- ♦ zamykanie procesu dostawczego — 12.4.



W praktyce

Jednym z najbardziej rozpaczliwych raportów na temat kiepskiego zaopatrzenia jest relacja kpt. Behra z 6. armii, który wspomina swoją wizytację wojsk rumuńskich w Stalingradzie:

„Ci biedacy tkwili w śniegu z bardzo marnym wyposażeniem, niektórzy bez koców, ze starymi karabinami, które przypominały czasy Napoleona. Zaopatrzenie u nich było bardzo złe, ale na tyłach oficerowie rozpierali się przy nakrytych białymi obrusami stołach, pili wino i nie odmawiali sobie niczego. Na miejscu tych rumuńskich żołnierzy też nie miałbym ochoty z entuzjazmem stawać do walki za Hitlera i poświęcać własne życie”⁶.

Procesy i techniki

Każdy z procesów, oprócz przynależności do obszaru wiedzy, należy do jednej z 5 głównych grup procesów. Wzajemna zależność tych grup jest stosunkowo prosta (rysunek 6).

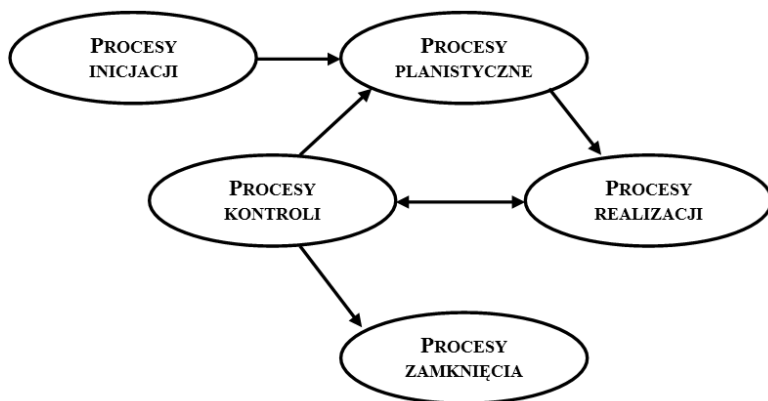
Jeden projekt może składać się z wielu etapów⁷ i każdy z nich będzie zarządzany przez PMBoK oddzielnie. W szczególnym przypadku, gdy dwa etapy zachodzą na siebie,

⁶ G. Knopp, *Stalingrad*, Świat Książki, Warszawa 2007, s.150.

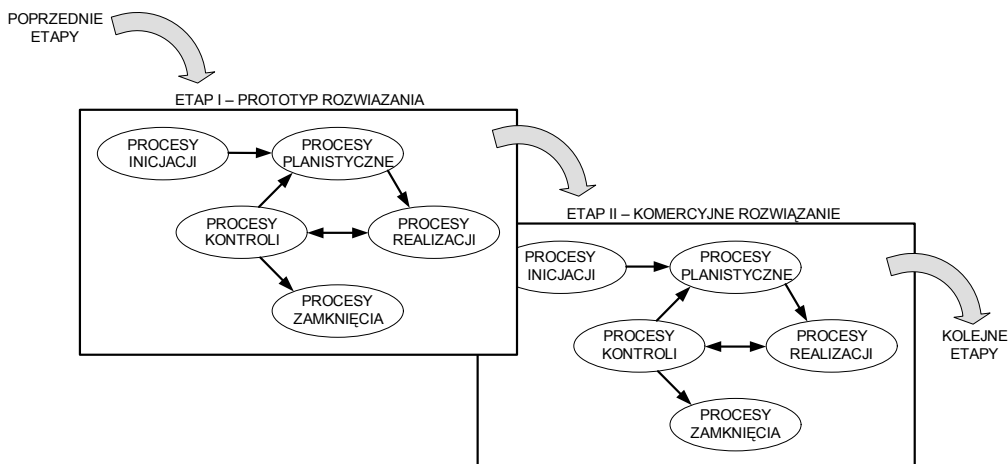
⁷ Formalnie, w nomenklaturze PMBoK „etap” nazywa się „fazą”.

Rysunek 6.

Architektura grup procesów PMBoK



możemy mieć do czynienia z sytuacją, gdy równocześnie uruchomione są procesy z różnych grup. Metodyka przewiduje sytuację, gdy faza pierwsza operuje na procesach zamknięcia, a równocześnie faza druga jest w okresie inicjacji (rysunek 7).

**Rysunek 7.** Współbieżność grup procesów PMBoK

W Prince2 etapy powinny być wykonywane sekwencyjnie. Można zastosować analogiczne podejście, ale taki wariant nie jest opisywany przez oficjalną dokumentację APM Group.

Poniżej zawarty został opis wszystkich grup procesów, ogólny opis każdego procesu oraz najbardziej interesujące techniki. Załącznik A — Lista wszystkich procesów PMBoK — zawiera szczegółowy opis wszystkich procesów.

Procesy inicjacji według PMBoK to wszelkie operacje związane z uruchomieniem projektu.

- ♦ 4.1. Przygotowanie dokumentu otwarcia — proces jest wymagany w każdym projekcie i inicjowany przez przekazanie dokumentu zakresu planowanych prac (ang. *Project Statement of Work*) i (lub) konkretnej umowy. W stosunku do tego procesu sugerowana jest technika polegająca na zasięgnięciu osądu eksperta, który może być pracownikiem danej organizacji, konsultantem, przedstawicielem klienta, inną osobą albo organizacją.
- ♦ 10.1. Identyfikacja interesariuszy — w ramach tego procesu identyfikowane są wszystkie osoby lub organizacje, które mają wpływ na projekt. Tworzony jest rejestr tych osób i organizacji oraz wykorzystywana technika analizy interesariuszy pod kątem najlepszego szablonu komunikacji. Istnieją cztery takie szablony:
 - ♦ utrzymanie satysfakcji (ang. *Keep Satisfied*) — dedykowany osobom o wysokim wpływie, ale niskim zainteresowaniu,
 - ♦ bliska współpraca (ang. *Manage Closely*) — dedykowany osobom o wysokim wpływie i wysokim zainteresowaniu,
 - ♦ stałe informowanie (ang. *Keep Informed*) — dedykowany osobom o niskim wpływie i wysokim zainteresowaniu,
 - ♦ monitorowanie (ang. *Monitor*) — dedykowany osobom o niskim wpływie i niskim zainteresowaniu.

Jest to proces analogiczny do procesu uruchamiania projektu (Przygotowanie Założeń Projektu (PP)) z metodyki Prince2.

Procesy planistyczne według PMBoK to uszczegółowienie zaakceptowanych ram projektowych i szereg taktycznych odpowiedzi na temat sposobu realizacji zadań, którego wynikiem jest kompletny, szczegółowy plan prac. W skład tej grupy procesów wchodzi procesy z różnych obszarów wiedzy.

- ♦ 4.2. Opracowanie planu zarządzania projektem — główny proces planistyczny, w ramach którego kierownik projektu uruchamia i zamyka wszystkie pozostałe procesy z tej grupy.
- ♦ 5.1. Zbieranie wymagań — udokumentowanie wymagań interesariuszy w kontekście realizacji celów projektowych.
- ♦ 5.2. Definiowanie zakresu projektu — ustalenie, co projekt ma zrealizować.
- ♦ 5.3. Utworzenie struktury pakietów roboczych, WBS (ang. *Work Breakdown Structure*) — definicja struktury pakietów roboczych, analogicznie do sposobu zaprezentowanego w Prince2.
- ♦ 6.1. Zdefiniowanie czynności — przejście od pakietów roboczych do listy zadań do wykonania (czynności).
- ♦ 6.2. Szeregowanie czynności — zazwyczaj pewne zadania muszą być wykonane w pewnej konkretnej kolejności. Rezultatem tego procesu jest pierwsza wersja harmonogramu.
- ♦ 6.3. Szacowanie zasobów czynności — zarezerwowanie odpowiednich zasobów (osób i sprzętu) niezbędnych do realizacji projektu.

- ♦ 6.4. Szacowanie czasu trwania czynności — długość trwania poszczególnych zadań.
- ♦ 6.5. Opracowanie harmonogramu — proces, który zamyka działania wykonane w procesach od 6.1 do 6.4 i daje w rezultacie bazowy harmonogram projektu.
- ♦ 7.1. Szacowanie kosztów — dane finansowe służące do oceny kosztu całego projektu, przygotowywane na bazie pakietów roboczych (5.3) i listy czynności (6.1).
- ♦ 7.2. Zatwierdzenie budżetu — bazowy plan kosztów jest wdrażany równoległe z zakończeniem prac nad harmonogramem (6.5).
- ♦ 8.1. Planowanie jakości — przygotowanie planu zarządzania jakością wytwarzanych przez projekt produktów i samego sposobu realizacji projektu.
- ♦ 9.1. Planowanie zasobów ludzkich — zdefiniowanie odpowiedzialności poszczególnych członków zespołu oraz ustalenie, kto, do kogo i kiedy raportuje w obrębie zespołu projektowego.
- ♦ 10.2. Planowanie komunikacji — zdefiniowanie mechanizmów przekazywania informacji pomiędzy kierownikiem projektu a interesariuszami.
- ♦ 11.1. Planowanie zarządzania ryzykiem — zdefiniowanie procedur zarządzania ryzykiem.
- ♦ 11.2. Identyfikacja ryzyka — określenie wejściowej listy ryzyk, które zostały wykryte przez zespół projektowy lub osoby spoza tego zespołu.
- ♦ 11.3. Jakościowa analiza ryzyka — analiza merytoryczna poszczególnych ryzyk.
- ♦ 11.4. Ilościowa analiza ryzyka — przełożenie wiedzy na temat ryzyka na wartości liczbowe w takich obszarach jak prawdopodobieństwo występowania lub wpływ na projekt.
- ♦ 11.5. Planowanie reakcji na ryzyko — podejmowanie decyzji związanych z ryzykami.
- ♦ 12.1. Planowanie zaopatrzenia — co, kiedy i jak powinno zostać zakupione lub uzyskane, włącznie z podjęciem decyzji typu „zrobić, czy kupić”.

Procesy planistyczne z PMBoK zawierają w sobie mechanizmy analogiczne do procesów planowania (PL), inicjowania projektu (IP) i zarządzania zakresem etapu (ZE) z metodyki Prince2, ale w obszarach związanych z zarządzaniem zasobami ludzkimi oraz zaopatrzeniem PMBoK wyraźnie wykracza poza ramy Prince2.

Każdy z wymienionych powyżej procesów zawiera pewną grupę sugerowanych technik. Wszystkie są wyliczone w załączniku A, ale część z nich jest szczególnie interesująca. ...

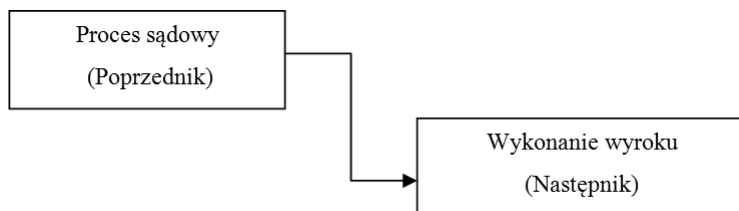
Techniki związane z procesem 6.2. Szeregowanie czynności

- ♦ **Metoda Diagramów Następstw** (ang. *Precedence Diagramming Method*) opisuje najbardziej popularny sposób wiązania ze sobą czynności w takich pakietach jak MS Project. Definiuje czynności jako węzły, które są ze sobą połączone strzałkami. Relacja pomiędzy zadaniami może być następująca.

Koniec-do-Początku: poprzednik musi zakończyć się, zanim zacznie się następnik (najczęściej wykorzystywany mechanizm łączenia zadań). Przykład — proces sądowy musi dobiec końca, zanim zacznie się wykonanie wyroku (rysunek 8).

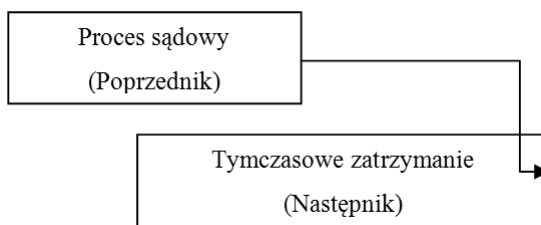
Rysunek 8.

Relacja pomiędzy zadaniami Koniec-do-Początku



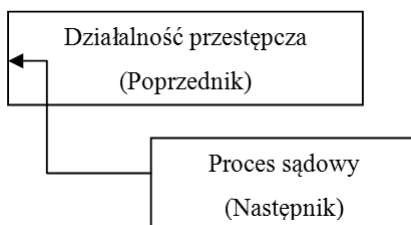
Oto inne, rzadziej stosowane typy relacji.

Koniec-do-Końca: poprzednik musi zakończyć się, zanim skończy się następnik (bardzo rzadko wykorzystywany mechanizm). Przykład — proces sądowy musi się skończyć, zanim skończy się tymczasowe zatrzymanie (rysunek 9).



Rysunek 9. *Relacja pomiędzy zadaniami Koniec-do-Końca*

Początek-do-Początku: poprzednik musi się rozpocząć, zanim zacznie się następnik (bardzo rzadko wykorzystywany mechanizm). Przykład — działalność przestępcza musi się rozpocząć, zanim zacznie się proces sądowy (rysunek 10).



Rysunek 10. *Relacja pomiędzy zadaniami Początek-do-Początku*

Początek-do-Końca: poprzednik musi się zacząć, zanim następnik się zakończy (bardzo rzadko wykorzystywany mechanizm). Przykład — proces sądowy musi się zacząć, zanim nastąpi przedawnienie (rysunek 11).

Rysunek 11.

Relacja pomiędzy
zadaniami
Początek-do-Końca



W praktyce

W praktyce zależności pomiędzy zadaniami dokumentuje się zazwyczaj za pomocą wykresów Gantta z wykorzystaniem aplikacji typu MS Project lub w arkuszu Excel. W obu przypadkach, jeżeli zachodzi konieczność wiązania ze sobą zadań, najczęściej stosuje się relacje typu Koniec-do-Początku, czyli w kolumnie Poprzednik zaznacza się konkretne zadania.

- ♦ **Technika analizy zależności** (ang. *Dependency Determination*) wyjaśnia charakter zależności pomiędzy czynnościami. I tak mamy:
 - ♦ zależności wymagane — związane z naturą pracy do wykonania,
 - ♦ zależności rozważne — związane z tradycją, najlepszymi praktykami, czyli logiczne,
 - ♦ zależności zewnętrzne — związane ze stanami lub produktami, które muszą zostać osiągnięte, dostarczone poza projektem. Zależności te powinny być elementem rejestru ryzyk.
- ♦ **Technika przyspieszeń i opóźnień** (ang. *Applying Leads and Lags*) wiąże dwie czynności na zasadzie „Rozpocząć zadanie B na X jednostek czasu, zanim zakończy się zadanie A” (przyspieszenie) lub „Zadanie B może się rozpocząć na X jednostek czasu po zakończeniu zadania A” (opóźnienie).



W praktyce

MS Project posiada tego typu funkcjonalność; jest to parametr o nazwie „Zwłoka” przy definiowaniu poprzedników zadania. Na rysunku 12. przedstawione jest zadanie B, które ma zacząć się na jeden dzień przed zakończeniem zadania A. Parametr „Zwłoka” w MS Project może przybierać wartość dodatnią (opóźnienie) lub ujemną (przyspieszenie).

- ♦ **Szablony harmonogramu sieciowego** (ang. *Schedule Network Templates*) — są to szablony harmonogramów wykorzystywane wtedy, kiedy w ramach projektu mają zostać dostarczone identyczne lub bardzo podobne produkty, takie jak piętra wieżowca.



Rysunek 12. Przyspieszenia i opóźnienia zadań w MS Project

Techniki związane z procesem 3.5. Opracowanie harmonogramu

Oprócz standardowych rozwiązań, takich jak wykres Gantta, PMBoK opisuje następujące techniki.

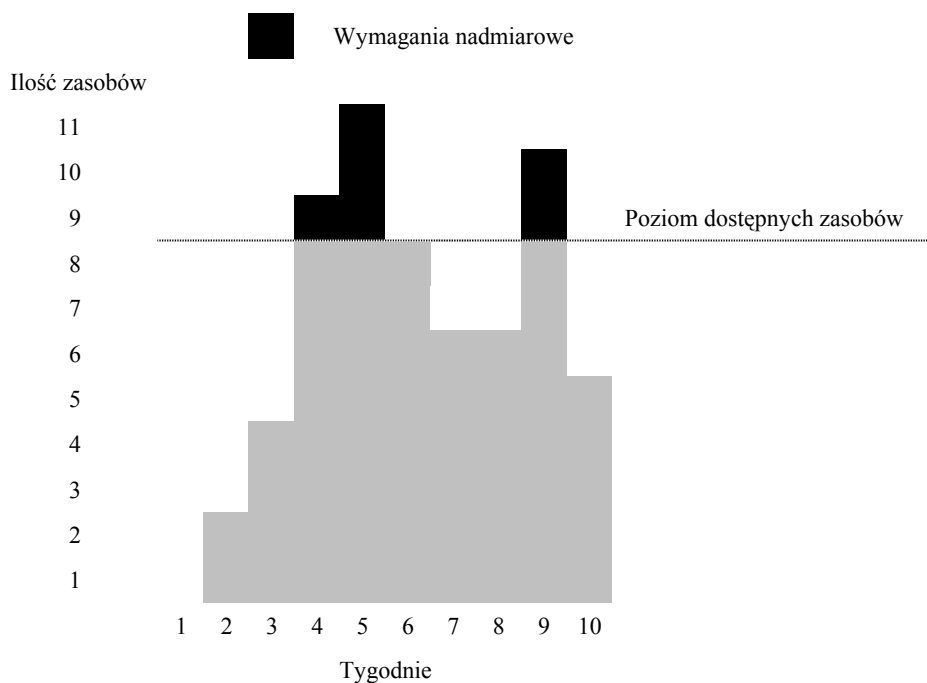
- ♦ **Analiza sieciowa harmonogramu** zawiera wszystkie techniki związane z tworzeniem harmonogramu projektu, takie jak metoda ścieżki krytycznej, metoda łańcucha krytycznego, analiza „co jeśli” i równoważenie zasobów. Głównym celem jest oszacowanie wcześniejszych oraz późniejszych dat rozpoczęcia i zakończenia czynności projektowych.
- ♦ **Metoda ścieżki krytycznej** (ang. *Critical path method*) dla każdej czynności szacuje optymistyczną (wcześniejszą) i pesymistyczną (późniejszą) datę rozpoczęcia i zakończenia. Szacunki te wykonane są bez uwzględnienia ograniczeń zasobowych. Następnie analizowane są wzajemne zależności między czynnościami. W rezultacie otrzymujemy informację o tym, w jakich granicach możemy przesuwając wykonanie poszczególnych czynności. Brak takiej swobody w stosunku do serii zadań określany jest mianem ścieżki krytycznej. Harmonogram może mieć kilka ścieżek krytycznych. Metoda ma na celu takie przemodelowanie planu, aby uzyskać maksymalnie dużą swobodę jego wykonania.
- ♦ **Metoda łańcucha krytycznego** (ang. *Critical chain method*) przyjmuje za punkt wyjścia zdefiniowane ścieżki krytyczne. Uzupełnia plany o ograniczenia zasobowe i tak zmodyfikowane ścieżki krytyczne uzyskują miano łańcucha krytycznego. W ramach tego procesu na końcu całego projektu dodawane są dodatkowe bufony czasu (ang. *the project buffer*), które mają zabezpieczyć projekt przed przekroczeniem terminów końcowych. Dodatkowo do łańcuchów zadań o największej niepewności dodawane są również dodatkowe bufony czasu (ang. *the feeding buffer*). Wykorzystując tę metodę, należy w trakcie realizacji projektu koncentrować się na właściwym stosowaniu buforów.
- ♦ **Równoważenie zasobów** (ang. *Resource leveling*) to technika, która zakłada duże ograniczenie w dostępności do zasobów. Przykładowo przy użyciu tego samego zasobu może realizować równocześnie dwa różne projekty, określona pula zasobów może być dostępna tylko pomiędzy konkretnymi datami na określonej długość. Tego typu ograniczenia mogą powodować znaczące zmiany w harmonogramie. Jeżeli np. okazuje się, że mamy osiem osób do długoterminowej dyspozycji, a wykres zaangażowania wygląda tak jak



W praktyce

Prawo Parkinsona⁸ mówi, że praca zawsze rozrasta się w taki sposób, aby zapłacić cały zaplanowany na nią czas. W związku z tym, nie ma sensu dodawać kolejnych buforów do każdej czynności, gdyż z pewnością zostaną zużyte. Warto dodać pewne bufory „na czarną godzinę” na końcu projektu poza konkretnymi zadaniami jako swoistą rezerwę strategiczną. Warto również motywować zespół do ich niewykorzystywania (np. premie).

na rysunku 13, to plany muszą zostać tak przeprojektowane, aby zrównoważyć wykorzystanie zasobów. Takie zmiany muszą zostać wykonane nawet wtedy, jeśli spowoduje to wydłużenie realizacji pewnych zadań.



Rysunek 13. Mechanizm równoważenia zasobów

- ♦ **Analiza scenariuszy „co, jeśli”** — w przypadku kilku wariantów realizacji projektu analizowane są konsekwencje każdego z nich.
- ♦ **Kompresja harmonogramu** to metoda skracania ścieżki krytycznej bez zmiany zakresu projektu. Wykorzystuje się do tego poniższe techniki.
- ♦ **Kruszenie** (ang. *crashing*) to technika, która analizuje koszty względem harmonogramu i próbuje uzyskać jak największą kompresję zadań za jak

⁸ Praca rozszerza się wprost proporcjonalnie do czasu wyznaczonego do jej wykonania (oryg. ang.: *Work expands as to fill the time available for its completion*).

najniższą cenę. Przykładowe sposoby stosowania tej techniki to nadgodziny, dodatkowe zasoby lub premie za realizację zadań na ścieżce krytycznej. Technika ta nie zawsze tworzy rzeczywistą alternatywę i może powodować zwiększone ryzyko projektowe.

- ♦ Szybkie śledzenie (ang. *Fast tracking*) to całkowite lub częściowe zrównoleglenie czynności, które normalnie byłyby wykonywane sekwencyjnie. Zrównoleglenie czynności powoduje konieczność ich końcowej integracji. Jest to pewien dodatkowy koszt i ryzyko, które należy wziąć pod uwagę przy okazji podejmowania tego typu decyzji.
- ♦ Oprogramowanie do zarządzania projektami (np. MS Project).

Techniki wspierające podejmowanie decyzji, stosowane m.in. w procesach 5.1. Zbieranie wymagań i 8.1. Planowanie jakości

- ♦ „Burza mózgów” (ang. *Brainstorming*) to swobodna dyskusja całego zespołu na kluczowe tematy projektowe, gdzie każdy ma równe prawo głosu i głowę otwartą na nieszablone pomysły. W trakcie tych sesji nie ma „głupich pomysłów”.



W praktyce

Rano po kawie, gdy umysł świeży, zabieramy członkinie i członków zespołu do salki konferencyjnej lub na spacer. Otwieramy tabliczkę czekolady, paczkę z paczkami lub kładziemy na stół bilety do teatru i pytamy, jak rozwiązać problem komiwojażera, choćby i w najbardziej oryginalny sposób... Pozostaje jedynie pobudzać umysły do twórczej dyskusji, rugać osoby, które dyskwalifikują cudze idee, i sumiennie notować najlepsze pomysły.

- ♦ **Technika grupy nominalnej** (ang. *Nominal group technique*) to „uczesana” wersja burzy mózgów, w której zebrana grupa jest zaznajomiona z problemem i samodzielnie zapisuje propozycje jego rozwiązania. Po spisaniu pomysłów każda osoba przedstawia swoje rozwiązanie reszcie zespołu, a następnie wszyscy dyskutują ze sobą, wyjaśniając i rozwijając warianty. Ostatecznie decyzja jest podejmowana w demokratycznym głosowaniu⁹.
- ♦ **Technika delficka** (ang. *The Delphi Technique*) — grupa ekspertów odpowiada na ankiety i udostępnia informację zwrotną, która umożliwia doprecyzowanie problemu. W następnej rundzie eksperci otrzymują kolejną propozycję rozwiązywania problemu wraz z listą anonimowych uwag i uzasadnień. Eksperti udzielają wtedy kolejnej serii odpowiedzi. Tego typu technika może zostać zastosowana do rozwiązywania kluczowych problemów biznesowych lub projektowych.

⁹ <http://www.joe.org/joe/2007february/iw1.shtml>

Wujek Dobra Rada — odcinek 3. „W zespole siła”

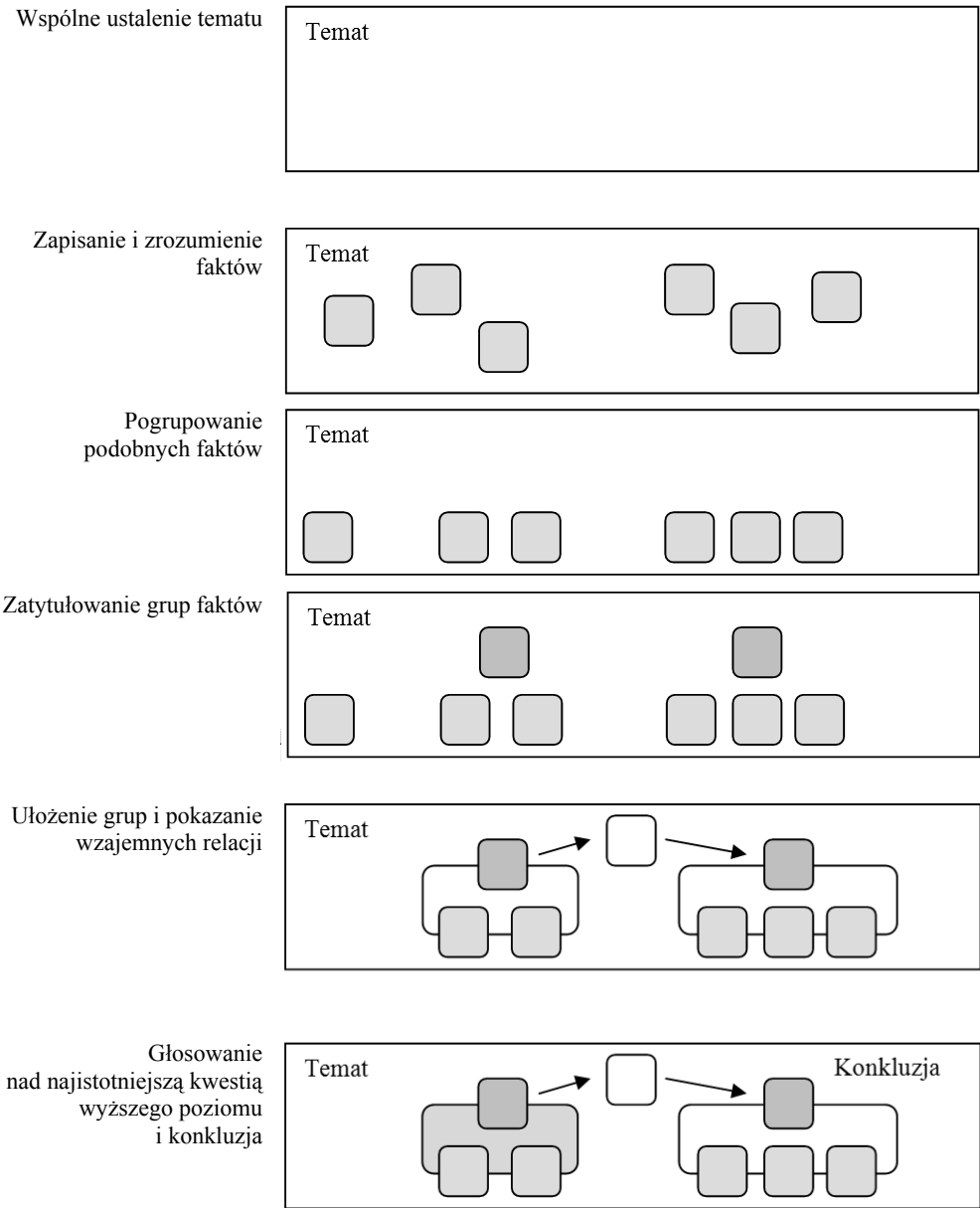


CO DWIE GŁOWY, TO NIE JEDNA — DEMOKRATYCZNY SPOSÓB PODEJMOWANIA DECYZJI
ZWIĘKSZA ZAANGAŻOWANIE CZŁONKÓW ZESPOŁU I ICH MOTYWACJĘ

- ♦ **Mapa pomysłów** (ang. *Idea/mind mapping*) to technika podobna do diagramów pokrewieństwa, która polega na umieszczeniu wszystkich pomysłów na jednej mapie, po to aby odnaleźć ich wzajemne różnice i części wspólne.
- ♦ **Diagramy pokrewieństwa** (ang. *Affinity diagram*) — metoda wymyślona w latach 60. ubiegłego wieku przez japońskiego antropologa Jiro Kawakitę. Jest to tablica, na której za pomocą żółtych karteczek wypisujemy wszystkie kwestie, grupujemy je, określamy między nimi relacje, nadajemy im priorytety i czynimy stosowne ustalenia na przyszłość, tak jak zaprezentowano na rysunku 14.
- ♦ **Analiza pola siły** (ang. *force field analysis*) to analiza sił działających na projekt, szczególnie użyteczna przy podejmowaniu kluczowych decyzji; jest to specjalizowana metoda analizy „za i przeciw”. Bazując na konkretnym projekcie, planie czy rozwiązaniu, należy określić siły działające na jego korzyść i niekorzyść oraz zdefiniować ocenę każdej z sił (rysunek 15).

Za pomocą takiego podejścia można podjąć bardziej trafną decyzję i zdefiniować, jakie siły należy wzmocnić, a jakie osłabić.

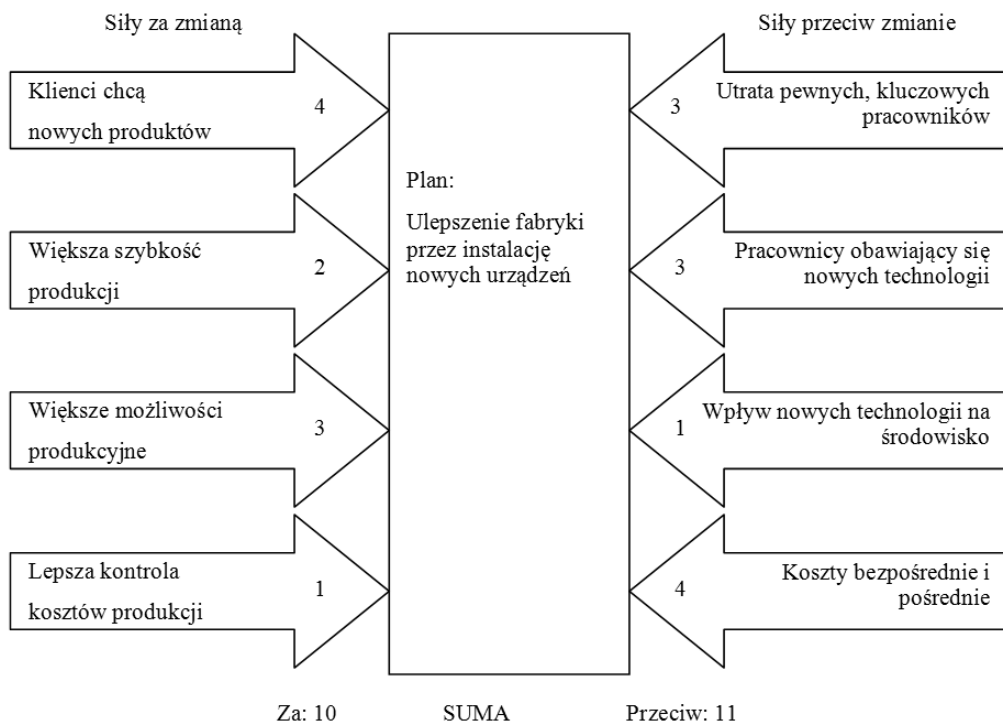
- ♦ **Diagram relacji** (ang. *interrelationship diagram*) to diagram relacji przyczynowo-skutkowych. Należy sformułować problem oraz kwestie z nim związane, a następnie zdefiniować związek przyczyna-skutek pomiędzy kwestiami



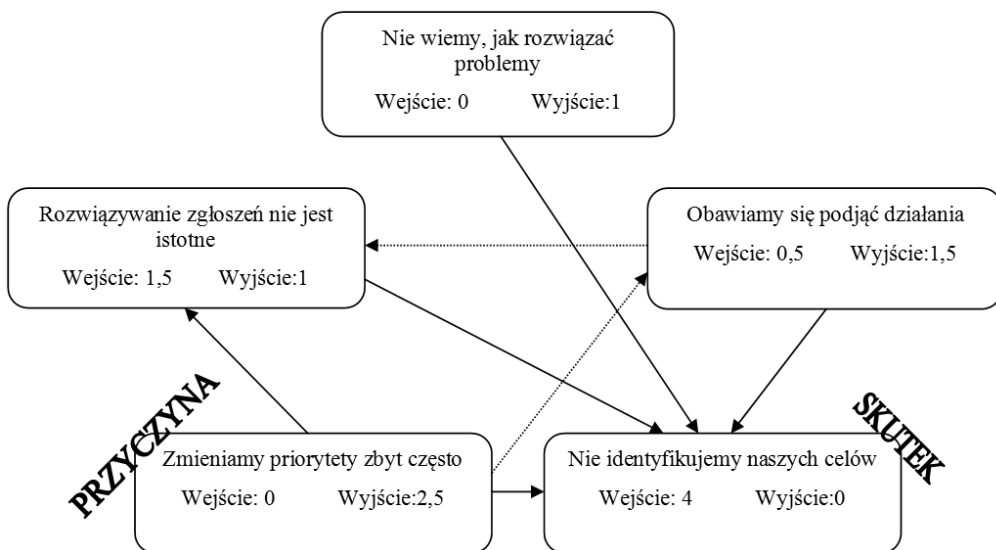
Rysunek 14. Przykład powstania diagramu pokrewieństwa

i zaprezentować go, rysując strzałki. W przypadku relacji słabej strzałka powinna być przerywana, a w przypadku relacji silnej — ciągła. Duża liczba strzałek wychodzących wskazuje główną przyczynę problemu¹⁰ (rysunek 16).

¹⁰ <http://web2.concordia.ca/Quality/tools/17interdiagram.pdf>



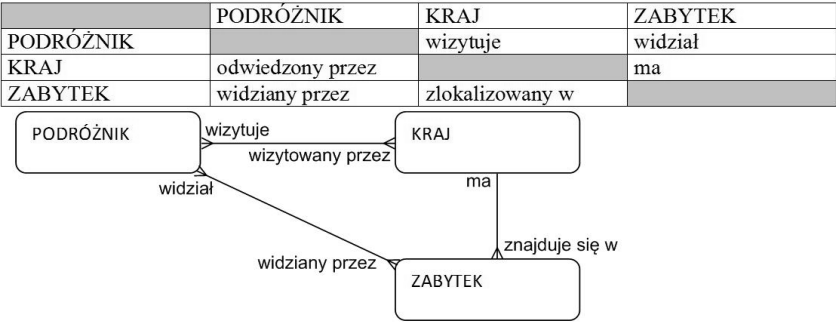
Rysunek 15. Przykład analizy pola siły



Rysunek 16. Przykład diagramu relacji: Dlaczego nie wykorzystujemy procesów rozwiązywania problemów?

- ♦ **Diagramy macierzowe** (ang. *matrix diagrams*) — porównywanie dwóch lub więcej grup pomysłów, określanie wzajemnych zależności i podejmowanie decyzji na bazie arkuszy Excela lub tablic „w kratkę”. Sposób może być wykorzystywany np. do zdefiniowania diagramu encji w bazie danych (rysunek 17).

Rysunek 17.
Przykład
diagramu
macierzowego



- ♦ **Diagramy przepływów** (ang. *Flowcharts*) to graficzna reprezentacja procesu wizualizująca czynności, punkty decyzyjne oraz kolejność procesowania. Takie podejście ma ułatwić możliwość wykrycia błędów lub przewidzenia, gdzie mogą potencjalnie wystąpić.
- ♦ **Matryce priorytetyzacyjne** (ang. *Prioritization matrices*) — na bazie arkuszy Excela lub tablic „w kratkę” należy wypisać w rzędach kryteria decyzyjne, a w kolumnach — możliwe opcje rozwiązania problemu. W każde z pól wewnątrz takiej tabeli trzeba wpisać siłę rozwiązania względem każdego z kryteriów. Dodatkowo możliwe jest uwzględnienie wagi każdego z kryteriów decyzyjnych. Następnie wystarczy wyliczyć sumę dla każdej opcji, aby wybrać najlepszą z nich¹¹. Przykładowe zastosowanie tej techniki zostało zaprezentowane w tabeli 1.

Tabela 1. Przykład matrycy priorytetyzacyjnej

	Waga (1 – 10)	A. Zakup gotowego rozwiązania	B. Samodzielne stworzenie własnego rozwiązania	C. Zlecenie wykonania rozwiązania
1. Koszt	8	10	6	8
2. Czas wykonania	5	10	4	6
3. Wewnętrzne kompetencje	4	0	10	3
Suma	(maks. 170)	130	108	106

¹¹ <http://www.maqin.org/brownbag/PrioritizationMatrixNov05.pdf>

Techniki wspierające proces 12.1. Planowanie dostaw

Jednym z głównych tematów związanych z planowaniem zaopatrzenia jest zaplanowanie kontraktów. Proces ten opisuje następujące typy kontraktów.

- ◆ Kontrakt o ustalonej cenie (ang. *Fixed price*) — kupujący specyfikuje wykonanie konkretnego zamówienia, które sprzedawca wykonuje za z góry ustaloną cenę.
- ◆ Standardowy kontrakt o ustalonej cenie (ang. *Firm Fixed Price Contracts FFP*) to najprostszy wariant, który zakłada, że każda zmiana przedmiotu zamówienia powoduje zmianę ceny.
- ◆ Kontrakt o ustalonej cenie z marginesem (ang. *Fixed Price Incentive Fee Contracts FPIF*) to wariant, w którym zawarto pewien margines płatności uzależniony od uzyskania ustalonych rezultatów (np. część płatności jest zależna od wydajności finalnego rozwiązania lub poziomu satysfakcji użytkowników). Ciekawym wariantem jest kontrakt, w którym partnerzy umawiają się na kwotę bazową plus udział w zyskach (ang. *revenue sharing*).
- ◆ Kontrakt o ustalonej cenie z opcją rewaloryzacji (ang. *Fixed Price with Economic Price Adjustment Contracts FP-EPA*) — w tym przypadku sprzedawca rezerwuje sobie prawo do rewaloryzacji ceny kontraktu np. wtedy, gdy jest on wieloletni. Taki kontrakt musi odnosić się bezpośrednio do konkretnych, zaufanych indeksów finansowych.
- ◆ Kontrakty zwrotu kosztów (ang. *Cost-reimbursable*) — kupujący zwraca sprzedawcy wszelkie koszty plus dodatkową kwotę reprezentującą zysk sprzedawcy.
- ◆ Koszt z ustaloną opłatą (ang. *Cost-Plus-Fixed-Fee CPFF*) to najprostszy wariant tego typu kontraktu, gdzie po zakończeniu prac sprzedawca uzyskuje dodatkową płatność będącą procentem kosztów bazowych.
- ◆ Koszt z marginesem (ang. *Cost-Plus-Incentive-Fee CPIF*) — w tym przypadku sprzedawca uzyskuje dodatkową płatność po uzyskaniu konkretnych celów ustalonych w kontrakcie. W tych kontraktach, jeżeli rzeczywisty koszt projektu jest inny od przewidywanego, sprzedawca i kupujący dzielą tę sumę według ustalonych w kontrakcie parytetów. Podział ten odbywa się według tego samego współczynnika, niezależnie od tego, czy różnica jest na minus, czy na plus.
- ◆ Koszt z premią (ang. *Cost-Plus-Award-Fee CPAF*) to system bardzo podobny do CPIF, ale większa część premii zależy od subiektywnej satysfakcji pewnego gremium. Subiektywność podejmowanej decyzji jest zapisana w kontrakcie i nie podlega zażaleniom.
- ◆ Kontrakt typu czas i materiał (ang. *Time and Material*) jest mieszanką dwóch powyższych. W tym przypadku sprzedawca sprzedaje kupującemu konkretne materiały i ludzi na określoną długość czasu. Pełna odpowiedzialność za zarządzanie tymi zasobami spoczywa na kupującym. W zależności od efektów, może on przyznać dodatkową marżę lub premię sprzedawcy. Kontrakt ten jest

często wykorzystywany do zdobycia niszowych zasobów szczególnie wtedy, jeżeli tego typu potrzeba jest krótkoterminowa i trudna do sprecyzowania w ramach odrębnego projektu.



W praktyce

W praktyce najpowszechniejsze na rynku są kontrakty o ustalonej cenie, ponieważ gwarantują uzyskanie efektu przy jasno określonej cenie. Coraz częściej spotyka się również umowy typu czas i materiał, szczególnie w przypadku projektów outsourcingowych.

Umowa o ustalonej, stałej cenie jest najpopularniejsza, ponieważ przenosi odpowiedzialność na wykonawcę. To on odpowiada za to, że prace zostaną wykonane w ramach budżetu i na czas. Umowa taka jest naturalnym wyborem, gdy można jasno określić jej przedmiot oraz kryteria powodzenia. Należy jednak pamiętać, iż prawdziwy sprawdzian takiej umowy następuje przy zamykaniu etapów i całego projektu. Dlatego właśnie płatności są uwarunkowane podpisaniem protokołu odbiorczego lub akceptacyjnego. Drugą stroną medalu jest to, że kupujący „lubi” tę procedurę czasami przeciągać. Dlatego też wiele umów szczegółowo specyfikuje mechanizm odbioru produktów.

Abstrahując od typu kontraktu, w trakcie jego podpisywania warto zawrzeć warunki gwarancji i serwisowania. W przypadku umowy serwisowej standardem jest już definiowanie kilku poziomów wsparcia (ang. *SLA* — *Service Level Agreement*) dla różnego typu zgłoszeń. Umowa taka winna definiować:

- ♦ mechanizmy rejestracji i zarządzania zgłoszeniami wraz z planem awaryjnym (co robić, jeśli podstawowy system rejestracji zgłoszeń zawiedzie),
- ♦ tryb ustalania zmian,
- ♦ mechanizmy komunikacji,
- ♦ okres świadczenia usługi (np. czy jest to 24/7/365, czy tylko w ustalonych godzinach pracy),
- ♦ możliwe priorytety zgłoszeń i ich szczegółowe definicje wykluczające subiektywną ocenę,
- ♦ czasy reakcji na zgłoszenia o poszczególnych priorytetach w ramach tabeli SLA,
- ♦ możliwe zasady eskalacji problemu (np. co będzie, jeśli wartości z tabeli SLA nie zostaną dotrzymane),
- ♦ dane kontaktowe do kluczowych osób w projekcie.

Tabela 2. to przykładowa tabela SLA.

W interesie kupującego jest ustalenie warunków takiej umowy, zanim jeszcze usługa lub produkt zostaną wdrożone. Negocjowanie odrębnej umowy tego typu po zrealizowaniu kontraktu nie jest dla niego korzystne, jest bowiem uzależniony od kupującego.

Tabela 2. Przykład tabeli SLA

Określenie czynności	Błąd krytyczny	Błąd poważny	Błąd drobny
Potwierdzenie przyjęcia zgłoszenia	W ciągu 4 godzin roboczych po otrzymaniu zgłoszenia	W ciągu 8 godzin roboczych po otrzymaniu zgłoszenia	W ciągu 2 dni roboczych po otrzymaniu zgłoszenia
Czas reakcji	W ciągu 1 dnia roboczego po otrzymaniu zgłoszenia	W ciągu 2 dni roboczych po otrzymaniu zgłoszenia	W ciągu 5 dni roboczych po otrzymaniu zgłoszenia
Czas naprawy	3 dni robocze lub ciągły wysiłek prowadzący do usunięcia błędu	5 dni roboczych lub ciągły wysiłek prowadzący do usunięcia błędu	Błąd zostanie rozwiązany w następnej wersji systemu. Wykonawca dołoży starań, aby rozwiązanie błędu nastąpiło w ciągu 30 dni od daty otrzymania zgłoszenia

Specyficznym przypadkiem kontraktów o stałej cenie są **przetargi publiczne** regulowane przez Prawo Zamówień Publicznych. Każdy potencjalny oferent przed złożeniem oferty musi zaznajomić się ze Specyfikacją Istotnych Warunków Zamówienia (SIWZ), która opisuje szczegółowo wszelkie kwestie dotyczące sposobu składania oferty oraz aspekty techniczne. Ustawa opisuje tryb zadawania pytań i udzielania odpowiedzi do tego dokumentu, a kluczową filozofią jest pełna jasność całego procesu. Oznacza to przykładowo, że wszyscy oferenci mają dostęp do wszystkich pytań, odpowiedzi oraz wszelkich pozostałych dokumentów powstałych w trakcie wyboru wygrywającej oferty. W dokumencie SIWZ zawarte są kryteria oceny oferty oraz wzór definiujący siłę oferty.



W praktyce

Cena nie zawsze jest jedynym kryterium wyboru i cała strategia przedstawiania ofert powinna bazować właśnie na kryteriach wyboru. Przyjmując pewne założenia wstępne, można przykładowo wyliczyć przybliżoną wartość jednego punktu, co będzie bardzo pomocne przy podejmowaniu konkretnych decyzji.

Należy ocenić, jaka będzie wartość cenowa oferty wygrywającej i na tej podstawie wycenić wartość jednego punktu w kategorii ceny. Jeżeli np. wiemy, że 1 pkt. jest wart około 10 tys. zł, a istnieje dodatkowe kryterium, które mówi, że wykonanie konkretnej funkcjonalności warte jest 10 pkt., możemy w jej realizację zainwestować do 70 tys. zł.

Oprócz wymagań „punktowanych”, czyli opcjonalnych, pojawia się bardzo wiele wymagań obligatoryjnych i nie zawsze są one przedstawione w sposób przejrzysty. Ustawa jasno zabrania ograniczania w SIWZ zakresu oferty do konkretnych produktów, usług lub producentów. Jeśli nawet w SIWZ pojawiają się konkretne nazwy producentów, zazwyczaj umieszczono tam kilka możliwych opcji albo sformułowanie „lub równoważ-

nych”. Jednak istnieje możliwość takiego sformułowania dokumentu SIWZ, aby tylko jeden podmiot w Polsce lub jeden produkt spełniał opisane wymagania. Bardzo dobrym przykładem jest tutaj oferta na dostawę „soków w szklanych butelkach z kapsłem z zawleczką”¹².

Wygrany przetarg jest jednak często dopiero połową sukcesu, ponieważ wielu konkurentów po przegraniu zgłasza protesty dotyczące ofert lub oferty wygrywającej. Zdarzały się przypadki, że w ten sposób wyeliminowano nawet pierwsze cztery oferty. Innym przypadkiem są protesty dotyczące samego przetargu. Bardzo często problemu nie udaje się załatwić w sposób polubowny i mamy do czynienia z wieloletnimi procesami, które mogą skutkować anulowaniem całego przetargu.

Gdy wreszcie ogłoszono nowy przetarg nieograniczony na Zautomatyzowany System Radarowego Nadzoru obejmujący całość granic morskich, do którego stanęło dziewięć konsorcjów, najniższą ofertę złożył holenderski Hitt (22 mln euro), ale rozminął się całkowicie ze Specyfikacją Istotnych Warunków Zamówienia. Hiszpański Amper Sistemas wycenił swoje usługi na 25 mln euro, Ster-Projekt na 28 mln euro, zaś Atlas na 29 mln euro. Następni byli: Allenia Marconi — 31 mln euro, Prokom Software — 34 mln euro, Sagem — 35 mln euro, EADS — 40 mln euro i Bumar — 43 mln euro. Kuzdumieniu wszystkich uczestników zwycięzcą zostało ponownie konsorcjum Atlas i WPRT.

Niewątpliwie perspektywa utraty 40 mln euro z Funduszu PHARE na ten system, co nastąpiłoby gdyby do 30 czerwca 2005 r. Straż Graniczna nie podpisała umowy z wykonawcą ZSRN, podziałała otępliwiejąco. W połowie czerwca, po serii protestów, unieważniono postępowanie i poproszono o oferty konsorcja: Bumar i Radwar; EDAS Deutschland, Alcatel i Atem; Prokom Software i Thales Naval; Emax, Safran i Sofrelog; SELEX Sistemi Integrati, Computex Telecommunication oraz Przemysłowy Instytut Telekomunikacji. O wyborze decydowała wyłącznie cena. 28 czerwca 2005 r. Komisja Przetargowa uznała, że najkorzystniejszą ofertę, ok. 30 mln euro, złożyło włoskie konsorcjum, którego liderem jest SELEX Sistemi Integrati (dawniej Alenia Marconi), należące do grupy Finmeccanica. To trzeci co do wielkości europejski koncern przemysłu obronnego, kosmicznego i elektroniki specjalnej. I z tym konsorcjum, do którego należą również Przemysłowy Instytut Telekomunikacji, producent systemów radarowych oraz Computex Telecommunication, Straż Graniczna podpisała umowę. System transmisji danych za 4,9 mln zł tworzy zaś olsztyński SPRINT, partner Motoroli.

Cóż z tego, skoro wkrótce prezes Urzędu Zamówień Publicznych podważył decyzję o ogłoszeniu negocjacji z powołaniem się na art. 5 PZP, który mówi o szczególnych sytuacjach w związku z obronnością kraju. Straży udało się cudem uniknąć nowego postępowania. Włoskie konsorcjum mogło zaś przygotowywać zintegrowaną sieć radarów i czujników elektrooptycznych wzdłuż polskiego wybrzeża, dwa narodowe centra kierowania (jedno będzie pełniło funkcję centrum zapasowego i szkoleniowego) oraz kilka centrów regionalnych. Całość miała zostać zintegrowana z pilotażowym systemem zbudowanym przez Atlas Elektronik i innymi już działającymi systemami ochrony wybrzeża i nadzoru ruchu statków. Termin zakończenia projektu to koniec 2006 r. Projekt się przedłużył. Dzisiaj Straż Graniczna tylko lakonicznie informuje, że

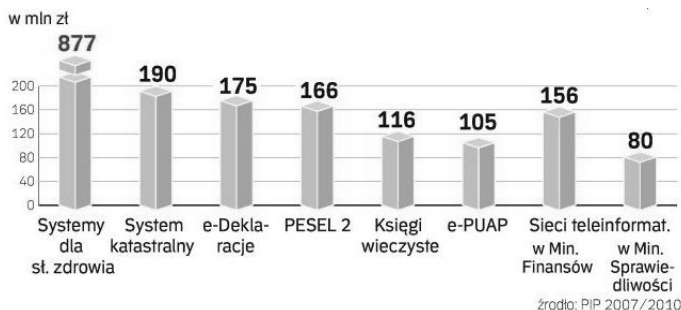
¹² Przemysław Gamdzyk, Sławomir Kosieliński, „Przetarg na tańsze państwo”, Computerworld 5-sty-01, archiwum elektroniczne.

„wybrane konsorcjum prowadzi zaawansowane prace nad budową systemu ZSRN. Termin zakończenia realizacji kontraktu przewidywany jest w roku bieżącym¹³.”

Pomimo tych trudności, wiele firm łakomym okiem patrzy na przetargi publiczne, ponieważ jest to spora część rynku IT (rysunek 18). Jeden wygrany przetarg może zapewnić wieloletnie, stałe przychody, perspektywę kolejnych zleceń oraz bardzo dobre referencje.

Rysunek 18.

Planu informatyzacji
państwa na lata
2007 – 2010



Znana jest opowieść o dwóch amerykańskich firmach, które startowały z tego samego punktu i wykonywały dokładnie takie same usługi. Jako że firmy te były właściwie zarządzane, jedna z nich po 5 latach posiadała już na własność przyzwoity dwupiętrowy budynek na przedmieściach miasta, a druga dziesięciopiętrowy budynek w centrum. Na czym więc polegała różnica między nimi? Skąd brała się taka różnica w zyskach? Otóż, pierwsza obsługiwała sektor prywatny, a druga — publiczny ☺.

Umowa typu czas i materiał jest używana często w sytuacji, gdy kupujący chce sam zrealizować konkretny projekt wewnętrznie, ale nie ma po temu wystarczających zasobów. Do takich umów należą też ramowe umowy wynajmu osób (ang. *body shopping*) od firm zewnętrznych. W stosunku do rekrutacji pracowników etatowych lub kontraktowych jest to rozwiązanie droższe, ale ma sens, gdy:

- ♦ brakuje zgody na dodatkowe etaty i kontrakty, przy równoczesnej zgodzie na dodatkowe koszty projektu,
- ♦ mamy do czynienia z projektem krótkoterminowym,
- ♦ istnieje konieczność stałej rotacji pracowników (np. testowanie rozwiązań bazujące na testerach nieznających aplikacji),
- ♦ poszukiwany jest doświadczony pracownik na część etatu (np. doświadczony administrator baz danych, którego obecność jest konieczna w każdy piątek),
- ♦ nie można znaleźć osób o niszowych specjalnościach,
- ♦ kluczową sprawą jest zdobycie kompetencji wewnątrz organizacji i pracownicy zewnętrzni mają jedynie wspierać firmę przy realizacji rozwiązania,
- ♦ umowa ramowa zawiera możliwość wynajęcia pracownika na okres stały po okresie przejściowym (forma leasingu).

¹³ Sławomir Kosieliński, „Pogranicze w ogniu (protestów)”, Computerworld 29-maj-07, nr 41.

Przygotowując umowę typu czas i materiał, warto zwrócić uwagę na kwestie związane z zarządzaniem zasobami ludzkimi.

- ♦ Czy pracownicy będą pracować na sprzęcie zleceniobiorcy, czy zleceniodawcy?
- ♦ Jakie będą „dodatki socjalne” i kto będzie je dostarczał? Jakie są możliwe lokalizacje pracy takich pracowników?
- ♦ Jakie są zasady raportowania i premiowania?
- ♦ Jakie są okresy rezygnacji z pracowników kontraktowych?
- ♦ Jakie są zasady ewentualnej „wymiany” pracowników, np. dwutygodniowa opcja „na zakładkę” umożliwiająca przekazanie zdobytych u klienta doświadczeń?
- ♦ Czy istnieje opcja podpisania kontraktu bezpośredniego między pracownikiem a kupującym po okresie przejściowym?
- ♦ Jakie są wymagania formalne, takie jak np. zaświadczenie o odbyciu służby wojskowej lub o niekaralności?
- ♦ Jakie są sposoby powiadamiania o chorobie, urlopach i zasady odpłatności za takie okresy?
- ♦ Jak rozwiązywane są kwestie decyzji i płatności za szkolenia oraz wyjazdy integracyjne (jeśli np. konieczne będzie dedykowane szkolenie dla danego pracownika, jak kupujący rozliczy się ze sprzedającym)?

W przypadku sprzętu i usług warto zastanowić się nad:

- ♦ opcją przyszłej integracji wynajmowanego sprzętu (usług) z istniejącym po stronie zleceniodawcy,
- ♦ poziomem niezawodności sprzętu (usług), np. procentem bezawaryjnego czasu działania w roku,
- ♦ zmianą lokalizacji sprzętu (usługi),
- ♦ okresem wypowiedzenia,
- ♦ zapewnieniem administracji sprzętem i jego bezpieczeństwa oraz możliwości audytu,
- ♦ zasadami backupu i odtwarzania systemów,
- ♦ zasadami upgrade’u sprzętu (usług),
- ♦ dodatkowymi wymaganiami zgodności ze standardami, np. ISO,
- ♦ zasadami odpłatności.

Procesy realizacji według PMBoK to wykonanie projektu.

- ♦ 4.3. Kierowanie i zarządzanie realizacją projektu — główny proces realizacyjny, w ramach którego kierownik projektu uruchamia i zamyka wszystkie pozostałe procesy z tej grupy.



W praktyce

Zarówno pracownicy „kontraktowi”, jak i pracownicy „na umowę o pracę” powinni być traktowani w maksymalnie podobny sposób.

- ♦ Powinni przejść normalny tryb rekrutacji.
- ♦ Muszą posiadać podobne prawa dostępu do systemów i wiedzy.
- ♦ Muszą raportować w podobny sposób.
- ♦ Powinni wyjeżdżać na te same imprezy integracyjne.

-
- ♦ 8.2. Zapewnienie jakości — mechanizm zapewnienia jakości w procesach związanych z projektem i w samym projekcie poprzez bieżącą kontrolę zgodności wykonanych prac z przyjętymi planami.
 - ♦ 9.2. Organizacja zespołu projektowego — czyli pozyskiwanie dla projektu osób poprzez rekrutację wewnętrzną i zewnętrzną.
 - ♦ 9.3. Rozwój zespołu projektowego — czyli budowanie umiejętności każdego pracownika z osobna i wszystkich razem. Proces ten opisuje m.in. okresową ocenę pracowników i mechanizmy motywacyjne.

Jeżeli chcesz zbudować statek, nie przywołuj mężczyzn, żeby zdobyć drewno i rozdzielić pracę, ale rozbudź w nich tęsknotę za otwartym, nieskończonym morzem.

Antoine de Saint Exupéry

- ♦ 9.4. Zarządzanie zespołem — czyli bieżąca praca operacyjna, na którą składa się m.in. przygotowywanie informacji zwrotnej o wydajności zespołu i rozwiązywanie sytuacji konfliktowych.
- ♦ 10.3. Dystrybucja informacji — mechanizmy informowania interesariuszy o postępach w projekcie.
- ♦ 10.4. Zarządzanie oczekiwaniami interesariuszy — aktywne zarządzanie oczekiwaniami interesariuszy.
- ♦ 12.2. Wybranie dostaw — proces selekcji i wyboru konkretnych sprzedawców z wykorzystaniem zdefiniowanych kryteriów wyboru.

Procesy kontroli według PMBoK to odrębna grupa procesów odpowiedzialna za weryfikację przebiegu projektu i monitorowanie ewentualnych odstępstw od planów. W ramach tej grupy możliwe jest ewentualne uruchomienie działań zapobiegawczych i (lub) naprawczych.

- ♦ 4.4. Monitorowanie i nadzór nad pracami projektu — główny proces kontrolny odpowiedzialny za stałe porównywanie stanu rzeczywistego względem planowanego, analizę trendów i rekomendowanie działań prewencyjnych oraz naprawczych.
- ♦ 4.5. Zintegrowane zarządzanie zmianami — centralny punkt, w którym weryfikowane są wszystkie przedstawione propozycje zmian i podejmowane kluczowe decyzje na ich temat.
- ♦ 5.4. Weryfikacja zakresu — proces formalnej akceptacji wykonanych produktów poprzez inspekcję z uwzględnieniem zdefiniowanych kryteriów akceptacji.
- ♦ 5.5. Kontrola zakresu — mechanizm kontrolowanego wprowadzania zmian do projektu, uwzględniający konieczność wprowadzenia zmian w obszarach czasu, budżetu, jakości, konfiguracji itp.
- ♦ 6.6. Kontrola harmonogramu — proces ten może być uruchomiony przez konieczność wprowadzenia zmiany; dodatkowo monitoruje bieżące wykonanie harmonogramu projektu.
- ♦ 7.3. Nadzór nad kosztami — analogicznie do pkt. 6.6. Kontrola harmonogramu proces ten jest uruchamiany przez konieczność wprowadzania zmian w budżecie; dodatkowo monitoruje bieżące wykonanie budżetu projektu. W obu przypadkach PMBoK specyfikuje bardzo ciekawą technikę uzyskanej wartości, szczegółowo opisaną poniżej.
- ♦ 8.3. Kontrola jakości — sprawdza konieczność wprowadzenia zmian do planów jakościowych i monitoruje jakość wytwarzanych produktów, wykorzystując do tego celu szereg ciekawych technik wymienionych poniżej.
- ♦ 10.5. Raportowanie wydajności — w ramach tego procesu zbierane są zagregowane dane na temat projektu i przygotowywany jest raport o wydajności jego realizacji. Raport ten przeznaczony jest głównie dla interesariuszy projektu.
- ♦ 11.6. Monitorowanie i nadzór nad ryzykami — proces stałego obserwowania listy ryzyk w trakcie trwania projektu.
- ♦ 12.3. Administrowanie dostawami — mechanizm kontroli realizacji dostaw.

Zarządzanie Wartością Wypracowaną (ang. *Earned Value Management* — EVM) związane z procesem „7.3. Nadzór nad kosztami”

W trakcie realizacji projektu jednym z największych wyzwań jest rzetelna weryfikacja, czy projekt postępuje zgodnie z planem, czy też nie. Do tego celu konieczna jest jednoczesna analiza wykorzystanego budżetu projektowego i stopnia realizacji zaplanowanych zadań. Technika EVM jest zazwyczaj stosowana w określonych punktach kontrolnych, czyli zaplanowanych momentach w trakcie trwania projektu. Ma to na celu sprawdzenie, czy stopień wydatkowanych środków jest adekwatny do postępu wykonanych prac względem planu bazowego. Oto kluczowe pojęcia.

- ♦ Wartość Zaplanowana (ang. *Planned Value* — PV) — planowane koszty częściowe, czyli budżet na poszczególne okresy rozliczeniowe (np. miesiące).

- ♦ Wartość Uzyskana (ang. *Earned Value* — EV) — przełożenie wykonanej pracy na wartość finansową. Wiąże ze sobą stopień wykonania harmonogramu z planowanymi kosztami. Jeżeli założymy, że w danym okresie rozliczeniowym planowaliśmy wydać kwotę 100 zł, a wszystkie założone prace zostały wykonane na czas, to $EV = 100$ zł. Jeżeli jednak do końca okresu wykonano jedynie połowę prac (np. opóźnienia), $EV = 50$ zł.
- ♦ Koszt Faktyczny (ang. *Actual Cost* — AC) — faktycznie poniesione koszty. Kontynuując powyższy przykład, przy wykonaniu 50% prac i wykorzystaniu całego zaplanowanego czasu może się okazać, że wydatkowano cały zaplanowany budżet. Koszt faktyczny nie jest bezpośrednio powiązany z Uzyskaną Wartością.
- ♦ Odchylenie Kosztowe (ang. *Cost Variance* — CV) — rozbieżność pomiędzy Wartością Uzyskaną a Kosztem Faktycznym ($EV - AC$). Jeżeli wartość ta jest równa 0, oznacza to, że poniesione koszty są adekwatne do stopnia wykonania prac. Jeżeli jest ujemna, znaczy, że stan faktyczny jest gorszy od zaplanowanego (prace kosztują więcej), a jeśli jest dodatnia, stan faktyczny jest lepszy od zaplanowanego (prace kosztują mniej).
- ♦ Odchylenie Harmonogramowe (ang. *Schedule Variance* — SV) — to rozbieżność pomiędzy Wartością Uzyskaną a Wartością Zaplanowaną ($EV - PV$). Jeżeli wartość jest ujemna, wskazuje to na opóźnienia, a jeśli dodatnia, oznacza przyspieszenie w stosunku do pierwotnego planu.
- ♦ Wskaźnik Efektywności Kosztowej (ang. *Cost Performance Index* — CPI¹⁴) — czyli Wartość Uzyskana podzielona przez Koszt Faktyczny (EV/AC). Rozbieżność kosztów przedstawia dokładnie te same informacje w nieco inny sposób; jeśli mamy wartość równą 1, poniesione koszty są adekwatne do stopnia wykonania prac, jeżeli poniżej 1, oznacza, że koszty przekraczają plan, a wartość powyżej 1 oznacza efektywność wydatkowania lepszą od planowanej.
- ♦ Wskaźnik Efektywności Harmonogramowej (ang. *Schedule Performance Index* — SPI) — faktyczna wydajność harmonogramu ($SPI = EV/PV$). Rozbieżność harmonogramu przedstawia dokładnie te same informacje w nieco inny sposób; rezultat poniżej 1 oznacza opóźnienia, a powyżej 1 — przyspieszenia.

Powyższe wartości występują też w postaci skumulowanej, m.in.:

- ♦ Budżet Całkowity (ang. *Budget At Completion* — BAC) — suma planowanych wartości (PV),
- ♦ Skumulowane Koszty Faktyczne (ang. *Actual Cost Cumulated* — ACC),
- ♦ Skumulowane Wartości Wypracowane (ang. *Earned Value Cumulated* — EVC),
- ♦ Skumulowane Odchylenia Kosztowe (ang. *Cost Variance Cumulated* — CVC = $EVC - ACC$),
- ♦ Skumulowane Odchylenia Harmonogramowe (ang. *Schedule Variance Cumulated* — SVC = $EVC - BAC$),

¹⁴ CPI może oznaczać również *Corporate Performance Indicator*, czyli kluczową miarę wydajności całej firmy.

- ♦ Skumulowany Wskaźnik Efektywności Kosztowej (ang. *CPI Cumulated* — $CPIC = EVC/ACC$),
- ♦ Skumulowany Wskaźnik Efektywności Harmonogramowej (ang. *SPI Cumulated* — $SPIC = EVC/BAC$).

Warto również zbadać trend tych zmian, bo może się okazać, że stan niższej konsumpcji zaplanowanych kosztów jest symptomem poważnych problemów, które wyjdą na jaw dopiero pod koniec projektu, dlatego technika ta zawiera również elementy **prognozowania**. Jeżeli okaże się, że wykonane prognozy znacząco odbiegają od planu bazowego, konieczne może być wykonanie nowego planu dla konkretnego etapu lub całego projektu. Wykorzystywane są tutaj następujące parametry.

- ♦ Szacunek Kosztu do Ukończenia (ang. *Estimate To Complete* — ETC) — koszty konieczne do poniesienia, aby zamknąć projekt (suma kosztów wszystkich czynności). Mogą zostać ustalone:
 - ♦ za pomocą nowych szacunków,
 - ♦ wyliczone na bazie odchylenia Budżetu Całkowitego od Skumulowanych Wartości Wypracowanych ($ETC = BAC - EVC$); jest to podejście, które powinno być stosowane, gdy efektywność realizacji projektu jest zgodna z planem: „Jeżeli budżet projektu to 100, a nasza Skumulowana Wartość Uzyskana to 60, przewidywany Koszt do Poniesienia wynosi 40”,
 - ♦ przy użyciu powyższych wyliczeń ważonych przez Skumulowany Wskaźnik Efektywności Kosztowej ($ETC = (BAC - EVC)/CPIC$ czyli $(BAC - EVC)/(EVC/ACC)$). Takie podejście bierze pod uwagę sytuację, w której projekt nie do końca idzie zgodnie z planem. Zakładając sytuację analogiczną do powyższej (budżet projektu to 100, a Skumulowana Wartość Uzyskana to 60) oraz dodatkową informację, że faktycznie wydatkowano nie 60, a 50, uzyskujemy $(100-60)/(60/50) = 33,3$, ponieważ efektywność realizacji projektu ($CPIC$) była wyższa od przewidywanej.
- ♦ Szacunkowy Budżet Całkowity (ang. *Estimate at completion* – EAC) — nowa prognoza całkowitego budżetu (BAC), czyli dotychczas poniesione koszty i nowe prognozy ($EAC = ACC + ETC$).

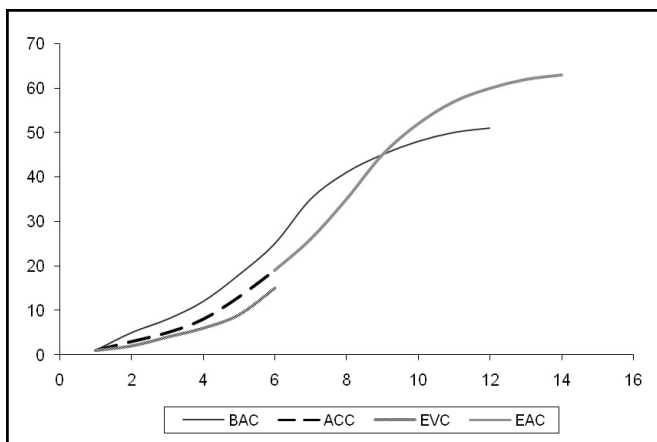
Przykład zastosowania techniki Zarządzania Wartością Wypracowaną

Kluczowe informacje przedstawiono na rysunku 19.

Projekt planowany był na dwanaście miesięcy, a jego budżet na 51 tys., zgodnie z linią BAC . W połowie projektu, w szóstym miesiącu została przeprowadzona zewnętrzna kontrola w postaci osądu eksperta, która wykazała znaczące rozbieżności. Okazało się, że zamiast planowanych 25 tys. (BAC) wydano 19 tys. (ACC), a wykonana praca odpowiada sumie zaledwie 15 tys. (EVC). Wyliczono, iż:

- ♦ Skumulowane Odchylenie Kosztowe ($CVC = EVC - ACC$) wynosi minus 4 tys.; oznacza to, że z punktu widzenia pierwotnego planu wypracowana wartość jest niższa od wydatkowanych kwot o 4 tys.

Miesiące	Planowane Koszty Częstkowe (PV)	Skumulowany Koszt Planowany (BAC)	Skumulowany Koszt Rzeczywisty (ACC)	Skumulowana Wartość Uzyskana (EVC)	Skumulowany Przewidywany Koszt (EAC)
1	1	1	1	1	
2	4	5	3	2	
3	3	8	5	4	
4	4	12	8	6	
5	6	18	13	9	
6	7	25	19	15	19
7	10	35			26
8	6	41			35
9	4	45			45
10	3	48			52
11	2	50			57
12	1	51			60
13					62
14					63
			CPIC	0,79	
			ETC	44	
			EAC	63,00	



Rysunek 19. Przykład zastosowania Zarządzania Wartością Wypracowaną (ang. *Earned Value Management — EVM*)

- ♦ Skumulowany Wskaźnik Efektywności Kosztowej ($CPIC = EVC/ACC$) wynosi 0,79; koszt wykonania prac jest o 21% wyższy niż przewidywano albo, inaczej mówiąc, efektywność jest o 21% niższa niż przewidywano.
- ♦ Skumulowana Rozbieżność Harmonogramu ($SVC = EVC - BAC$) wynosi minus 10 tys.; o tyle mniej udało się wypracować względem planów w połowie projektu.

- ♦ Skumulowany Indeks Rozbieżności Harmonogramu (SPIC = EVC/BAC) wynosi tylko 0,6; wykonano zaledwie 60% zaplanowanej pracy!

Postanowiono opracować plan naprawczy, który zakładał wydłużony o dwa miesiące czas trwania projektu, oraz wyliczono, że do zakończenia projektu konieczne będzie poniesienie dodatkowych nakładów w kwocie 12 tys. ($ETC = (BAC - EVC) / CPIC$), co dało nowy plan budżetu (linia EAC) na poziomie 63 tys.

Należy zwrócić uwagę, że niektóre publikacje operują nieco innym nazewnictwem. Oto przykłady.

- ♦ Budżetowy Koszt Planowanej Pracy (ang. *Budgeted Cost of Work Scheduled* — BCWS) = Wartość Zaplanowana (ang. *Planned Value* — PV).
- ♦ Rzeczywisty Koszt Wykonanej Pracy (ang. *Actual Cost of Work Performed* — ACWP) = Koszt Faktyczny (ang. *Actual Cost* — ACC).
- ♦ Budżetowy Koszt Wykonanej Pracy (ang. *Budgeted Cost of Work Performed* — BCWP) = Wartość Wypracowana (ang. *Earned Value* — EV).
- ♦ Nowe Przewidywania (ang. *Forecast To Go* — FTG) = Szacunek Kosztu do Ukończenia (ang. *Estimate To Complete* — ETC).
- ♦ Przewidywany Skumulowany Koszt (ang. *Forecast of remaining work* — FCST) = Szacunkowy Budżet Całkowity (ang. *Estimate At Completion* — EAC).

Wersja czwarta PMBoK wprowadza również pojęcie „indeksu niezbędnej wydajności do zakończenia projektu” (ang. *To-Complete Performance Index* — TCPI). Wartość ta opisuje minimalną wartość CPI, która umożliwi realizację projektu zgodnie z planem. Podstawowa wersja formuły to:

$$TCPI = (BAC - EV) / (BAC - AC)$$

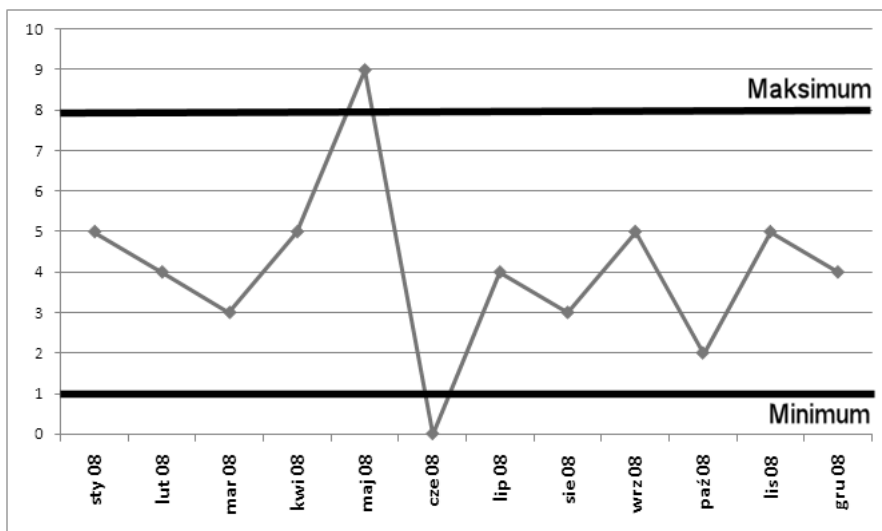
Widać więc, że bazując na „Skumulowanym Koszcie Planowanym (BAC)”, TCPI określa możliwą wartość odchylenia „Wartości Uzyskanej (EV)” w stosunku do kosztów (AC). Im więcej zapasów w projekcie, tym niższe wymagania w stosunku do TCPI.

Parametr ten może zostać wykorzystany przy przeplanowaniu projektu, podczas wyliczenia Skumulowanego Przewidywanego Kosztu (EAC). W takim przypadku cel wydajnościowy będzie wyliczany ze wzoru:

$$TCPI = (BAC - EV) / (EAC - AC)$$

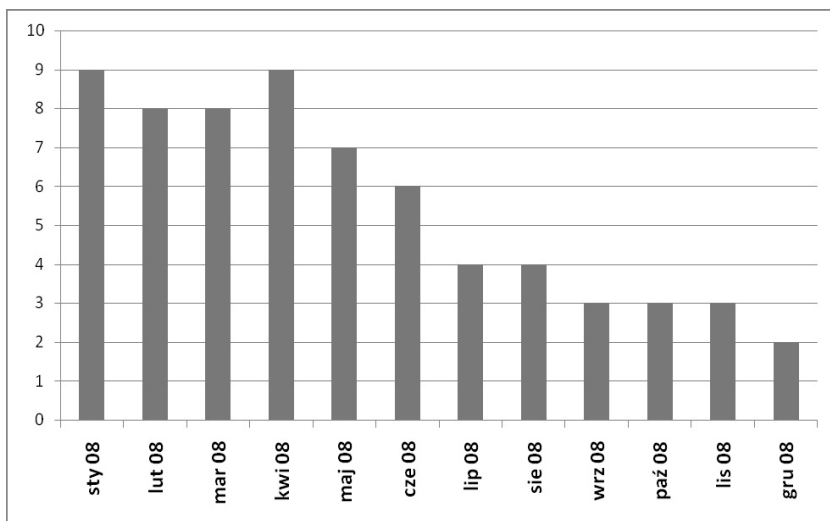
Techniki związane z procesem „8.3. Kontrola jakości”

- ♦ **Wykresy kontrolne** umożliwiają obserwowanie, czy proces jest stabilny, tzn. czy zawiera się pomiędzy zdefiniowanymi limitami górnym i dolnym. Głównym celem tej techniki jest wczesne alarmowanie o nieprawidłowym przebiegu procesu. Przykład na rysunku 20. przedstawia wykres kontrolny dla liczby błędów w projekcie. Problemem jest zarówno sytuacja, gdy błędów jest za dużo (maj), jak i sytuacja, kiedy błędów nie ma wcale (czerwiec), bo może to świadczyć o złym sposobie mierzenia błędów lub niewykonaniu procedur kontrolnych.



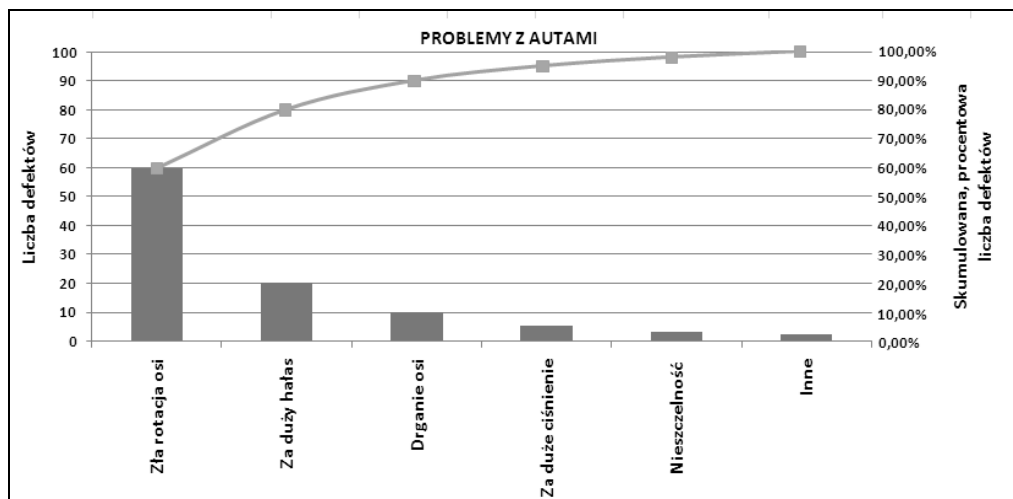
Rysunek 20. Przykład wykresu kontrolnego

- ♦ **Histogram**, czyli prezentacja wartości parametru w czasie. Przykładowy histogram wykorzystania zasobów ludzkich w projekcie można zobaczyć na rysunku 21.



Rysunek 21. Przykład histogramu

- ♦ **Wykres Pareto** — istnieje zasada, że 80% problemów jest powodowanych przez 20% przyczyn i w związku z tym warto analizować zaistniałe problemy oraz koncentrować wysiłki naprawcze na tych kategoriach, które generują najwięcej problemów (rysunek 22).



Rysunek 22. Przykład wykresu Pareto

- ♦ **Wykres Postępu** (ang. *Run Chart*) pokazuje, w jaki sposób przebiegał proces w czasie, jakie wystąpiły w nim trendy i warianty, co pozwala zdefiniować możliwe udoskonalenia w procesie. Kluczową różnicą między wykresem próby a histogramem jest dodatkowa analiza trendów.
- ♦ **Diagram Rozrzutu** (ang. *Scatter Diagram*) umożliwia wskazanie związku pomiędzy dwoma zmiennymi na osiach X i Y oraz badanie ewentualnej, wzajemnej korelacji. Firma Gartner znana jest z przedstawiania trendów rynkowych w taki właśnie sposób.
- ♦ **Próby Statystyczne** to wykonanie badań na wybranym zbiorze danych (populacji).
- ♦ **Inspekcja** to badanie rezultatów prac przez pryzmat ich zgodności ze standardami.
- ♦ **Przegląd Napraw Defektów** to weryfikacja, że błędy zostały właściwie naprawione.

Procesy zamknięcia według PMBoK są uruchamiane, gdy następuje formalne zakończenie projektu lub jego etapu. Możliwe jest tutaj wykorzystanie dwóch procesów.

- ♦ 4.7. Zamknięcie projektu — formalne zamknięcie wszystkich procesów projektowych oraz udokumentowanie stopnia realizacji celów, które zostały zdefiniowane w procesach inicjacji.
- ♦ 12.6. Zamknięcie kontraktów — zamknięcie wszelkich dostaw z podwykonawcami w kontekście danego projektu oraz skompletowanie dokumentacji na temat tych dostaw.

Dostosowanie do potrzeb organizacji

Należy jasno zaznaczyć, że *nie ma* obowiązku wykonania wszystkich procesów i kierownik projektu lub organizacja są zobligowani do właściwej skali zastosowania metodyki. PMBoK definiuje pewne obszary wiedzy, które wkraczają w metody organizacji pracy danej firmy, takie jak obszary związane z zaopatrzeniem lub zarządzaniem zasobami ludzkimi. W związku z tym, oprócz kierownika projektu, również i cała firma musi zdefiniować, które z praktyk wymienianych przez metodykę będą stosowane, a które nie.

PMBoK jest przystosowany do „wyrywkowego” zastosowania, szczególnie w kontekście technik. Przykładowo z powodzeniem można w umowie serwisowej zawrzeć definicję poziomu świadczenia usług (SLA) bez konieczności wdrażania któregośkolwiek z procesów. Jeżeli więc któraś z wymienionych technik jest remedium na istniejące problemy, można ją zastosować bez większego zachodu i bez konieczności implementowania całego procesu.

Certyfikacja

Za kwestię szkoleń i certyfikacji odpowiedzialny jest Project Management Institute (PMI) posiadający narodowe oddziały na całym świecie, także w Polsce. Istnieją trzy certyfikaty:

- ♦ *Project Management Professional (PMP®)* jest najbardziej rozpoznawalnym i rozpowszechnionym certyfikatem PMI, który funkcjonuje od 1984 roku. Do marca 2008 roku wydano już ponad 260 tys. certyfikatów PMP, z czego 150 tys. w USA. Wymagane jest odbycie trzydziestu pięciu godzin szkoleń w autoryzowanym centrum szkoleniowym i trzydzieści sześć miesięcy udokumentowanego doświadczenia w projektach w okresie sześciu lat dla osób z wykształceniem wyższym lub sześćdziesiąt miesięcy udokumentowanego doświadczenia w projektach w okresie ośmiu lat dla osób z wykształceniem średnim. Od 2005 roku wymaga się udzielenia w przeciągu czterech godzin poprawnej odpowiedzi na sto sześć z dwustu pytań, ale dwadzieścia pięć losowo wybranych pytań nie jest ocenianych, zatem wymagana jest skuteczność na poziomie 61%. Egzamin odbywa się wyłącznie w języku angielskim. Co trzy lata certyfikat wymaga potwierdzenia, polegającego na zebraniu 60 punktów PDU poprzez odbycie autoryzowanych przez PMI szkoleń, udział w konferencjach itp. Przykładowo szkolenie firmy IBM o nazwie „Analiza i projektowanie obiektowe z wykorzystaniem języka UML” za 1350 USD daje 24 PDU. Certyfikat PMP upoważnia do prowadzenia projektów zgodnie z metodyką PMBoK.
- ♦ *Certified Associate in Project Management (CAPM)* to stosunkowo nowy certyfikat, rodzaj „małego PMP” dla osób pragnących udokumentować znajomość metodyki. Charakteryzuje się niższymi wymaganiami wstępnymi: dwadzieścia trzy godziny szkoleń w autoryzowanym centrum szkoleniowym lub dwanaście miesięcy udokumentowanego doświadczenia w projektach. W przeciągu trzech godzin wymaga się poprawnej odpowiedzi na osiemdziesiąt sześć ze stu

pięćdziesięciu pytań, ale piętnaście losowych pytań nie jest w ogóle ocenianych, co daje próg 61% skuteczności. Egzamin wyłącznie w języku angielskim. Certyfikat jest ważny przez pięć lat i po tym okresie wymagane jest ponowne zdanie egzaminu. Certyfikat CAMP upoważnia do uczestnictwa w projektach zgodnych z PMBoK.

- ♦ *Program Management Professional (PgMP®)* to najświeższy pomysł PMI, który został uruchomiony 1 października 2007 roku i jest przeznaczony dla najbardziej doświadczonych kierowników projektów oraz kierowników programów. Wymaga się co najmniej czterech lat doświadczenia w kierowaniu projektami i od czterech do siedmiu lat kierowania programami, w zależności od poziomu edukacji. Następnie w przeciągu czterech godzin należy odpowiedzieć na sto siedemdziesiąt pytań, z czego dwadzieścia losowych pytań nie jest ocenianych. Na końcu aranżowana jest ocena kandydata online przez dwunastu wskazanych, dotychczasowych współpracowników, w formie oceny typu 360 stopni — system Multi-Rater Assessment (MRA). Nie ma wymogu posiadania certyfikatu PMP lub CAPM. Certyfikat wymaga potwierdzenia co trzy lata; polega ono na zebraniu 60 punktów PDU i poniesieniu kosztu dodatkowej opłaty 150 – 170 USD (ang. *Continuing Certification Requirements* — CCR).

Oto autoryzowane centra szkoleniowe w Polsce na kwiecień 2008 roku: IBM, Infovide-Matrix, Instytut ZTI, Management Training & Development Center, Kepner-Tregoe, PM Experts oraz Thomson Prometric Test Center, który jest międzynarodowym, elektronicznym systemem zdawania egzaminów, a w Polsce można z niego skorzystać w Polsko-Japońskiej Szkole Technik Komputerowych.

Koszty związane ze zdobyciem certyfikatów na kwiecień 2008 roku to:

- ♦ PMP — obligatoryjne szkolenie za około 4 tys. złotych plus egzamin za 535 – 555 USD,
- ♦ CAPM — egzamin to koszt około 300 USD,
- ♦ PgMP — egzamin to koszt około 1630 – 1800 USD, a ocena to dodatkowe 400 USD.

Podsumowanie

PMBoK i Prince2 to metodyki konkurencyjnie w dziedzinie zarządzania projektami. O ile Prince2 koncentruje się bardziej na przygotowaniu szablonu procesów do zastosowania, a dobre praktyki są niejako „efektem ubocznym”, to PMBoK koncentruje się na zdefiniowaniu dobrych praktyk w określonych obszarach wiedzy i dodatkowo oferuje pewien szablon procesów. PMBoK, w porównaniu z Prince2, umożliwia nieco większą swobodę w doborze procesów, ale jest to broń obosieczna.

W porównaniu z Prince2, PMBoK posiada nieco więcej technik związanych z praktycznymi aspektami prowadzenia projektu, takich jak EVM lub techniki prac grupowych. Metodyka ta stara się opisać całą wiedzę związaną ze sposobem realizacji

projektów IT, nawet w takich obszarach jak zarządzanie kontraktami i zasobami ludzkimi. PMBoK jest bez wątpienia bardzo ciekawym i praktycznym podejściem, czego dowodem jest rosnąca popularność tej metodyki na świecie.

W rozdziale „Przykład PMBoK i MSF — wdrożenie systemu BI” opisany został przykład projektu realizowanego zgodnie z metodykami PMBoK i MSF.

Część II

Metodyki wytwórcze a praktyka

Metodyki zarządcze i metodyki wytwórcze nie wykluczają się nawzajem, a nawet zawierają pewne części wspólne. Niemniej jednak istnieją między nimi dość znaczne różnice.

- ◆ Metodyki zarządcze koncentrują się na procesie decyzyjnym, a szczególnie na kwestiach związanych z uruchomieniem projektu, zdefiniowaniem zakresu poszczególnych etapów oraz jego zamknięciem.
- ◆ Metodyki wytwórcze skupiają się na procesie produkcyjnym i definiują pewne techniczne sposoby tworzenia produktów, takie jak fazy analizy czy stabilizacji.

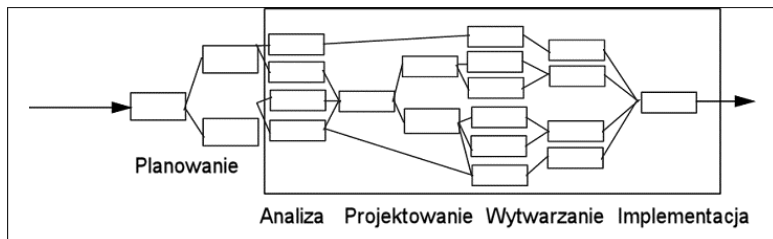
W praktyce każda organizacja może przyjąć różne modele formalizacji obu procesów. Oto one:

- 1) wybrane metodyki zarządcza i (lub) wytwórcza obowiązujące ogólnie w całej organizacji (np. Prince2 i [lub] RUP),
- 2) własna metodyka obowiązująca ogólnie (najczęściej bazuje ona na jednej lub kilku powszechnie znanych metodykach),
- 3) zdefiniowana pula możliwości do wyboru przez kierownika projektu,
- 4) brak formalizacji, czyli każdy kierownik projektu sam decyduje o ewentualnym wykorzystaniu metodyki.

Opisując różne metodyki, nie można pominąć dyskusji na temat modeli kaskadowego, spiralnego i iteracyjnego. Te przymiotniki to pewien zestaw cech, które dana metodyka może posiadać. Część metodyk stara się łączyć te modele i np. w MSF mówi się o połączeniu modeli spiralnego i iteracyjnego. Inne metodyki wskazują na konkretne „źródło inspiracji”; właśnie RUP wywodzi się z modelu spiralnego.

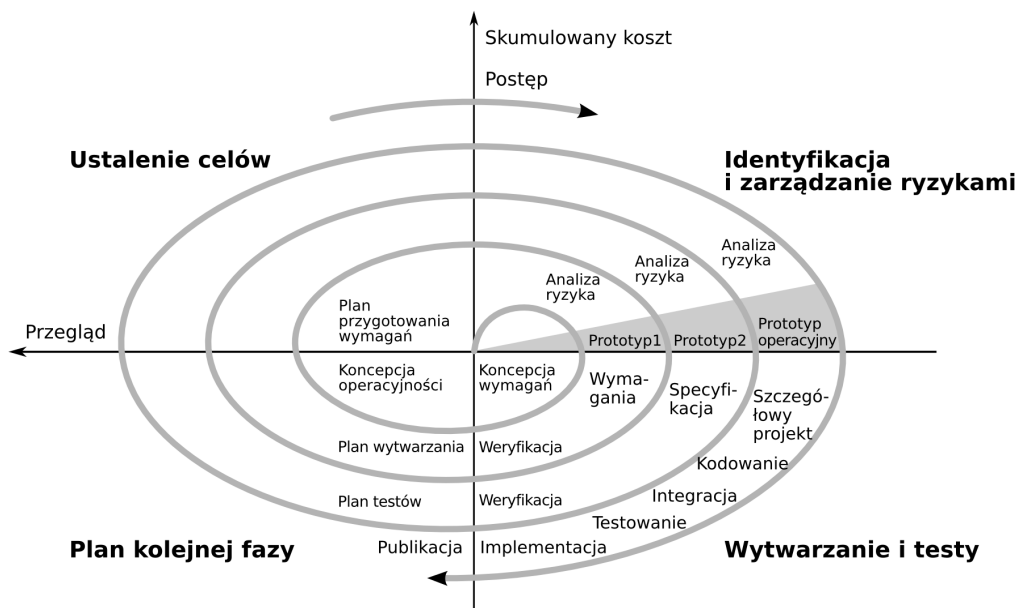
Podejście kaskadowe oznacza, że cały projekt musi zostać najpierw całkowicie zaplanowany, a następnie w całości przeanalizowany, zaprojektowany, zakodowany i zainstalowany. Poszczególne fazy nie mogą na siebie zachodzić i muszą zostać w całości wykonane (rysunek 23). W szczególności oznacza to, że w przypadku wieloetapowego projektu

Rysunek 23.
Model kaskadowy



faza analizy musi opisywać dokładnie każdy etap. Podejście kaskadowe jest z lubością przeciwstawiane metodykom adaptacyjnym jako stare i złe; to klasyczny „chłopiec do bicia”. Żadna metodyka „nie przyznaje” się już dzisiaj, że jest kaskadowa, chociaż wciąż istnieją firmy, które z powodzeniem stosują to podejście w praktyce. Przykładowo model kaskadowy może być zastosowany w kontraktach z ustaloną ceną.

Klasyczna „spiralność” została zaproponowana przez Barry’ego Boehma w 1988 roku w artykule „A Spiral Model of Software Development and Enhancement”¹. Model bazuje na założeniu przyrostowego dochodzenia do finalnego rozwiązania. Pierwszą fazą jest stworzenie koncepcji, drugą — wymagań, a ostatnia to wytworzenie. Każda z faz zaczyna się od określenia celów, a kończy zaplanowaniem kolejnej fazy. Unikalną cechą tego modelu jest to, że w trakcie każdej fazy tworzony jest lub rozbudowywany prototyp rozwiązania (rysunek 24). Metodyka RUP jasno odnosi się do tego właśnie modelu.



Rysunek 24. *Model spiralny*

¹ <http://www.scribd.com/doc/12597552/a-spiral-model-of-software-development-and-enhancement>

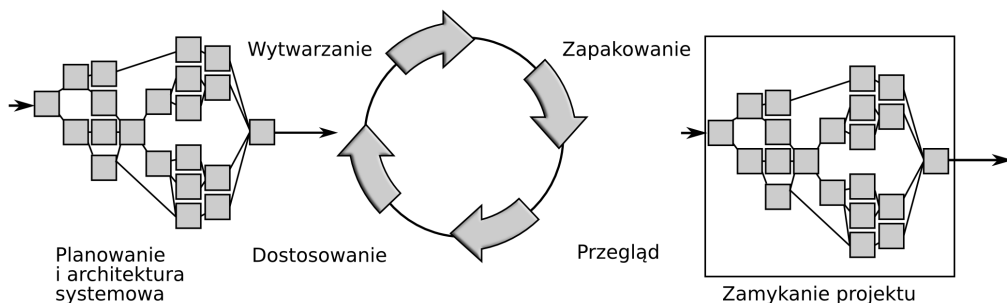
Model iteracyjny jest połączeniem modeli kaskadowego i spiralnego. Tak samo jak model spiralny, zakłada on przyrostowe dochodzenie do finalnego rozwiązania. Zamiast idei prototypowania zakłada tworzenie kolejnych, coraz bardziej dojrzałych wersji. Każda z nich jest wytwarzana dokładnie w taki sam sposób, począwszy od analizy, a skończywszy na testowaniu (rysunek 25). Metodyka MSF jasno powołuje się właśnie na ten model.

Rysunek 25.

Model iteracyjny



Zupełnie inne podejście zakładają metodyki adaptacyjne, które koncentrują się wyłącznie na pracach wytwórczych. W zakres metodyk adaptacyjnych nie wchodzi wstępna analiza wymagań. Model ten jest uruchamiany dopiero w momencie, gdy ustalona została biznesowa wizja rozwiązania, np. po podpisaniu umowy. Model adaptacyjny nie specyfikuje również etapów, z jakich składa się proces wytwórczy, i kwestie te pozostawia do ustalenia zespołowi (rysunek 26).



Rysunek 26. *Model adaptacyjny*

Do tego modelu dostosowuje się wiele istniejących metodyk, takich jak Agile MSF, Agile Unified Process, (kuzyn metodyki RUP), a nawet tak oryginalne rozwiązania jak eXtreme Prince (XPrince), promowane przez grupę Jerzego Nawrockiego z Politechniki Poznańskiej.

Rozdział 3.

Rational Unified Process (RUP)

RUP jest szablonem przyrostowego sposobu wytwarzania oprogramowania i wywodzi się wprost z modelu spiralnego. Zakłada, że jest już ustalona pewna wizja projektu i wstępne zezwolenie na uruchomienie jego analizy biznesowej. Z jednej strony, wyróżnikiem tej metodyki jest szeroka możliwość jej dostosowania do potrzeb projektu, dzięki doborowi właściwych komponentów. Z drugiej, jest to gotowe do użycia rozwiązanie, które pomaga w ustaleniu kluczowych kamieni milowych projektu oraz zasobów niezbędnych do jego realizacji. RUP jest znanym bardzo powszechnie standardem, wykorzystywany przez tysiące firm na całym świecie.

Szczypta historii

Metodyka RUP jest nierozzerwalnie związana z językiem modelowania UML (ang. *Unified Modelling Language*). Metodyka została pierwotnie stworzona przez firmę Rational Software, powszechnie znaną z doskonałego onegdaj narzędzia Rational Rose. Służyło ono właśnie do modelowania diagramów UML. RUP 5.0 została opublikowana w 1998 roku. Było to pierwsze rozwiązanie, które umożliwiało przejście od „rysunku” do szkieletu klas. Można wręcz zaryzykować stwierdzenie, że RUP była opakowaniem dla UML, który miał zapewnić, że zestaw diagramów przerodzi się w pełnoprawną, niezależną metodykę.



W praktyce

W 2003 roku IBM wykupił firmę Rational Software za 2,1 miliarda USD, wykorzystując pęknięcie eBańki. Od tego czasu następuje pewne upolitycznienie tej metodyki i pojawiają się problemy ze wsparciem dla narzędzi firmy Microsoft — perturbacje z linią produktową XDE. Dzieje się tak, ponieważ IBM, ze swoim tandemem RUP/Eclipse,

konkuruje z MSF/Visual Studio.Net. Microsoft nigdy nie próbował nawet zintegrować się z metodyką RUP i dość zaczepnie publikuje szereg materiałów porównujących tę metodykę z MSF.

W praktyce, niewiele firm w Polsce wytwarza projekty, utrzymując stałą i automatyczną synchronizację warstwy logicznej w UML z fizyczną w kodzie. Automatyczne generowanie szkieletu klas i zapewnienie kompatybilności wstecznej (ang. *re-engineering*) należy do rzadkości. Dzieje się tak, ponieważ wytwarzanie projektów w taki sposób wiąże się z wyższymi kosztami. Zazwyczaj zadowalające jest wygenerowanie odpowiedniej dokumentacji z diagramami UML, a szkielet rozwiązania jest generowany niezależnie. Do dobrej praktyki należy aktualizacja diagramów UML po zakończeniu wytwarzania rozwiązania fizycznego. Jest to jednak jedno z tych zadań, o których „lubi się zapominać” ☺.

Proces

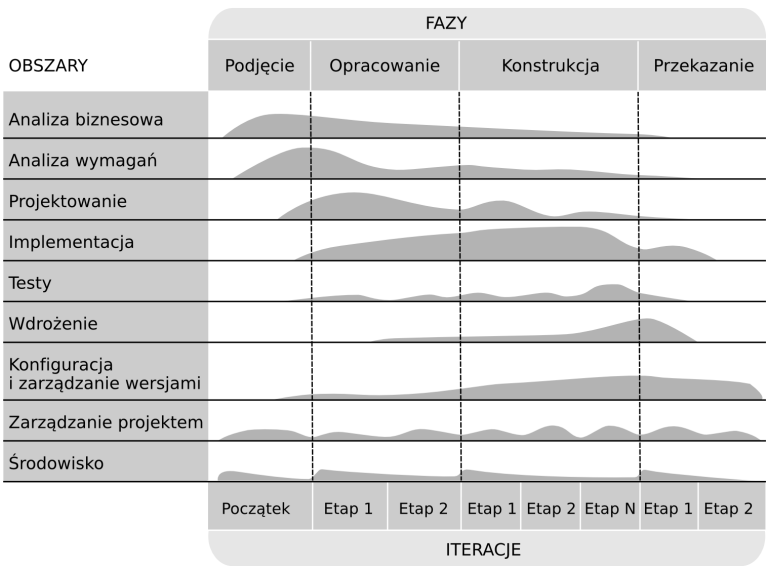
Główną ideą metodyki RUP jest uruchomienie prac na trzech poziomach: wysokim poziomie abstrakcji (ang. *High Level Definition*), niskim poziomie abstrakcji (ang. *Low Level Definition*) i prac wytwórczych. Tak jak zostało przedstawione to na rysunku 27., RUP wyróżnia:

- ♦ dziewięć dyscyplin projektowych, które powinny zostać „obsłużone” przez poszczególnych członków zespołu,
- ♦ cztery fazy projektu, z których każda może zawierać jedną bądź wiele iteracji. Każda z iteracji kończy się osiągnięciem wyraźnego kamienia milowego oraz podjęciem decyzji o rozpoczęciu kolejnej iteracji. Kolejna iteracja może oznaczać przejście do następnej fazy lub kolejną iterację w ramach tej samej fazy (np. poprawę przygotowanej analizy).

Oto poszczególne fazy RUP.

- ♦ **Faza Podjęcia** (ang. *Inception*) to faza analizy biznesowej. Głównymi produktami tej fazy są kluczowe przypadki użycia (ang. *use cases*) oraz czynniki powodzenia projektu (ang. *success factors*). Czynnikiem takim może być model zyskowności albo kryteria akceptacyjne. Wykonana analiza biznesowa musi zostać zaakceptowana przez kluczowego użytkownika końcowego i sponsora projektu przed uruchomieniem kolejnej fazy (ang. *Lifecycle Objective Milestone*). W przeciwnym przypadku faza ulega powtórzeniu, czyli mamy do czynienia z kolejną iteracją fazy.
- ♦ **Faza Opracowania** (ang. *Elaboration*) to faza analizy wymagań. Głównymi jej produktami są:
 - ♦ wszystkie istotne przypadki użycia (około 80%),
 - ♦ architektura całego rozwiązania,

Rysunek 27.
Architektura RUP



- ♦ plan projektu,
- ♦ plan jakości,
- ♦ listy ryzyk.

Faza Opracowania ma na celu weryfikację możliwości realizacji projektu przy przyjętych założeniach biznesowych. Faza kończy się formalną decyzją o uruchomieniu Fazy Konstrukcji, zatrzymaniu lub powtórzeniu całej fazy (ang. *Lifecycle Architecture Milestone*).

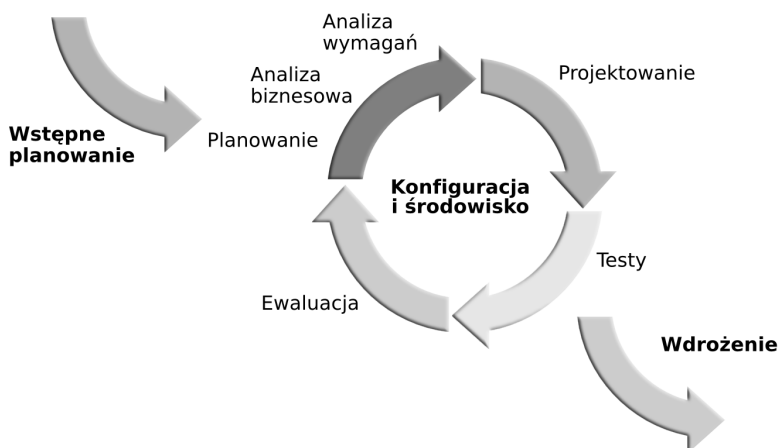
- ♦ **Faza Konstrukcji** (ang. *Construction*) to faza wykonania rozwiązania. Zawiera dodatkowe prace przygotowawcze, kodowanie, testowanie oraz inne działania konieczne do wytworzenia finalnego produktu. Zaleca się co najmniej kilka iteracji kodowania i sukcesywną prezentację rezultatów końcowemu użytkownikowi (ang. *Initial Operational Capability Milestone*).
- ♦ **Faza Przekazania** (ang. *Transition*) jest fazą przekazania finalnego rozwiązania do systemu produkcyjnego. Może zawierać:
 - ♦ przygotowanie instalatorów systemu,
 - ♦ poszerzenie zakresu konfigurowalności systemu,
 - ♦ końcowe testy akceptacyjne, walidujące zgodność ze wstępnymi założeniami biznesowymi,
 - ♦ szkolenia wewnętrzne i zewnętrzne,
 - ♦ testy alfa i beta,
 - ♦ przygotowanie dokumentacji.

Możliwe jest kilkukrotne powtórzenie (iteracja) całej fazy, np. ze względu na konieczność naprawy błędów lub dodatkowe zmiany. Cała faza kończy się odebraniem rozwiązania (ang. *Product Release Milestone*).

Dyscypliny RUP

RUP, oprócz faz, specyfikuje również dyscypliny, które są pewnymi obszarami specjalizacji. Tak jak w dobrej orkiestrze, każda z nich „włącza” się w konkretnym momencie partytury, którą przedstawia rysunek 28.

Rysunek 28.
*Przepływ pracy
w metodyce RUP*



- ♦ **Modelowanie biznesowe** to wstępne definiowanie zakresu projektu przeprowadzane ze sponsorami projektu i użytkownikami końcowymi. W trakcie tej analizy powinny zostać określone:

- ♦ główne źródła wartości dodanej całego systemu,
- ♦ typy użytkowników,
- ♦ formuła biznesowa całego przedsięwzięcia,
- ♦ główne moduły biznesowe.

Większość tych prac jest wykonana w Fazie Podjęcia. Po akceptacji przez użytkowników końcowych, w trakcie Fazy Konstrukcji, konieczne mogą być poprawki i zmiany. Im bliżej końca tej fazy, tym bardziej restrykcyjna powinna być kontrola zakresu projektu.

- ♦ **Wymagania** to szczegółowe wypunktowanie kluczowych funkcjonalności systemu oraz opisanie ich w sposób, który będzie zrozumiały dla programistów. Można powiedzieć, że jest to pomost między kwestiami biznesowymi a technicznymi. Dyscyplina ta zawiera prace równoległe w stosunku do modelowania biznesowego. Wpływ na zdefiniowane wymagania mogą mieć również prace analityczne i projektowe, które przynoszą rewizję wstępnych założeń.

- ♦ **Analiza i projektowanie** to dyscyplina odpowiedzialna za przygotowanie dokumentacji niezbędnej do wykonania oprogramowania w konkretnym środowisku programistycznym. Dyscyplina ta zajmuje się również określeniem architektury technicznej systemu poprzez wyszczególnienie modułów i komponentów. Większość prac analitycznych i projektowych jest wykonana w Fazie Rozwinięcia, ale po rozpoczęciu implementacji mogą być nanoszone poprawki i zmiany.
- ♦ **Implementacja** to działania związane z procesem wytwórczym, polegające m.in. na:
 - ♦ właściwym zorganizowaniu środowiska programistycznego,
 - ♦ oprogramowaniu rozwiązania,
 - ♦ testach jednostkowych i integracyjnych,
 - ♦ kontroli wersji,
 - ♦ budowaniu rozwiązania.

Wskazane jest, aby najtrudniejsze i najbardziej ryzykowne elementy zostały wstępnie zbadane w ramach dowodów wykonalności już w trakcie Fazy Opracowania. Realizacja architektury technicznej to głównie Faza Konstrukcji.

- ♦ **Testowanie** to walidacja jakości wytwarzanego oprogramowania oraz zgodności wykonanego oprogramowania ze zdefiniowanymi wymaganiami i przyjętymi założeniami biznesowymi. Testowanie może być „manualne” bądź zautomatyzowane i zawierać m.in. testy akceptacyjne, funkcjonalne, wydajnościowe, destrukcyjne, bezpieczeństwa czy platformowe. W trakcie Fazy Opracowania powinny zostać przygotowane scenariusze akceptacyjne oraz kluczowe scenariusze testowe. Same testy wykonywane są w Fazie Konstrukcji. W Fazie Przekazania użytkownicy końcowi mogą wymagać wykonania dodatkowych testów lub konieczna będzie analiza wykrytych przez nich błędów.
 - ♦ **Wdrożenie** to wszelkie działania konieczne do przeniesienia rozwiązania z środowisk programistycznego i testowego do środowiska produkcyjnego. Zawiera m.in. przygotowanie:
 - ♦ instalatorów rozwiązania,
 - ♦ dokumentacji dla poszczególnych grup użytkowników,
 - ♦ planu wdrożenia,
 - ♦ finalnych testów akceptacyjnych,
 - ♦ uruchomienia wsparcia powdrożeniowego.
- Wdrożenie to esencja Fazy Przekazania, ale przygotowania do niego zaczynają się już na końcu Fazy Konstrukcji.
- ♦ **Zarządzanie zmianą i konfiguracją** to dyscyplina związana z:
 - ♦ wersjonowaniem finalnego produktu (ang. *release*),

- ♦ wersjonowaniem kodu,
- ♦ wersjonowaniem dokumentacji,
- ♦ konfiguracją systemów w środowiskach programistycznym, testowym oraz produkcyjnym.

W ramach tej dyscypliny zawiera się również zdefiniowanie konfigurowalności systemu — zbyt duża liczba opcji spowoduje problemy z przetestowaniem każdej możliwości, a zbyt mała niepotrzebnie „usztynwi” system. W obrębie tej dyscypliny znajdują się również procedury obsługi zgłoszeń, bez względu na to, czy są to zgłoszenia dotyczące błędów w systemie, czy propozycji zmian. Warto pokusić się o wykorzystanie narzędzi do ewidencjonowania zgłoszeń i błędów (ang. *Bug Tracking Systems* — BTS).

Wujek Dobra Rada — odcinek 4. „Lista przebojów”



NIEZAREJESTROWANY BŁĄD LUB PROPOZYCJA ZMIANY LUBIĄ GINAĆ.
JUŻ SAM FAKT NADANIA NUMERU SPRAWIE MA „MAGICZNY” SKUTEK

Dyscyplina ta „włącza” się w Fazie Konstrukcji, kiedy konieczne jest zarządzanie wersjami wytwarzanego kodu oraz konfiguracją. Odgrywa również ważną rolę w Fazie Przekazania, kiedy konieczny jest nadzór nad wersjami finalnego produktu oraz jego konfiguracją u klienta.

Zarządzanie projektem to wszelkie prace zarządcze, takie jak zdefiniowanie i monitorowanie ryzyk, plan projektu i zarządzanie kryzysowe.

Zarządzanie środowiskiem to wszelkie prace administracyjne związane z utrzymaniem środowisk programistycznych, testowych i produkcyjnych.

Załącznik B — Specyfikacja dyscyplin RUP z Rational Method Composer (RMC) — zawiera szczegółową specyfikację wszystkich, wymienionych powyżej dyscyplin.

W praktyce dyscypliny te mogą należeć do obowiązków pracowników pełniących różne role w organizacji, oto dwa przykłady.

Scenariusz A

- ♦ Analityk biznesowy — modelowanie biznesowe i wymagania
- ♦ Projektant — analiza i projektowanie
- ♦ Programista — implementacja
- ♦ Tester — testowanie, zarządzanie zmianą i konfiguracją
- ♦ Wdrożeniowiec — wdrożenie
- ♦ Kierownik projektu — zarządzanie projektem
- ♦ Administrator — zarządzanie środowiskiem

Scenariusz B

- ♦ Konsultant — modelowanie biznesowe, wymagania, wdrożenie
- ♦ Projektant (programista) — analiza i projektowanie, implementacja
- ♦ Tester — testowanie
- ♦ Kierownik projektu — zarządzanie projektem
- ♦ Administrator — zarządzanie zmianą i konfiguracją, zarządzanie środowiskiem

Abecadło metodyki RUP

Metodyka RUP powstała na bazie badań projektów, które zakończyły się fiaskiem. W rezultacie zdefiniowano zbiór sześciu dobrych praktyk (ABCDEF) projektowania, które są sterowane celami biznesowymi (ang. *Business driven development*). Nie są to li tylko puste hasła ☺...

- ♦ **Adaptuj proces** (ang. *Adapt the process*) — dopasuj proces wytwórczy do projektu i nigdy nie stosuj zasad na ślepo. Główną zaletą RUP jest elastyczność. Jest ona tak duża, że niektórzy postrzegają tę cechę jak wadę. Oto, co na ten temat napisał Martin Fowler:

Moje doświadczenie z RUP jest takie, że jego nieograniczona dostosowywalność stwarza problemy. Napotykałem przypadki użycia RUP od modelu kaskadowego z iteracjami analitycznymi do pełnego procesu Agile. Uderzyło mnie to, że promowanie RUP jako pojedynczego procesu doprowadziło do tego, że ludzie mogą zrobić wszystko i nazwać to RUP, co prowadzi do tego, że RUP staje się nic nieznaczącym słowem¹.

Sytuacja ta jest wynikiem przemyślanej strategii firmy IBM, która udostępnia szereg dokumentów opisujących sposób integracji metodyki RUP z innymi

¹ Martin Fowler, The New Methodology, <http://www.martinfowler.com/articles/newMethodology.html>

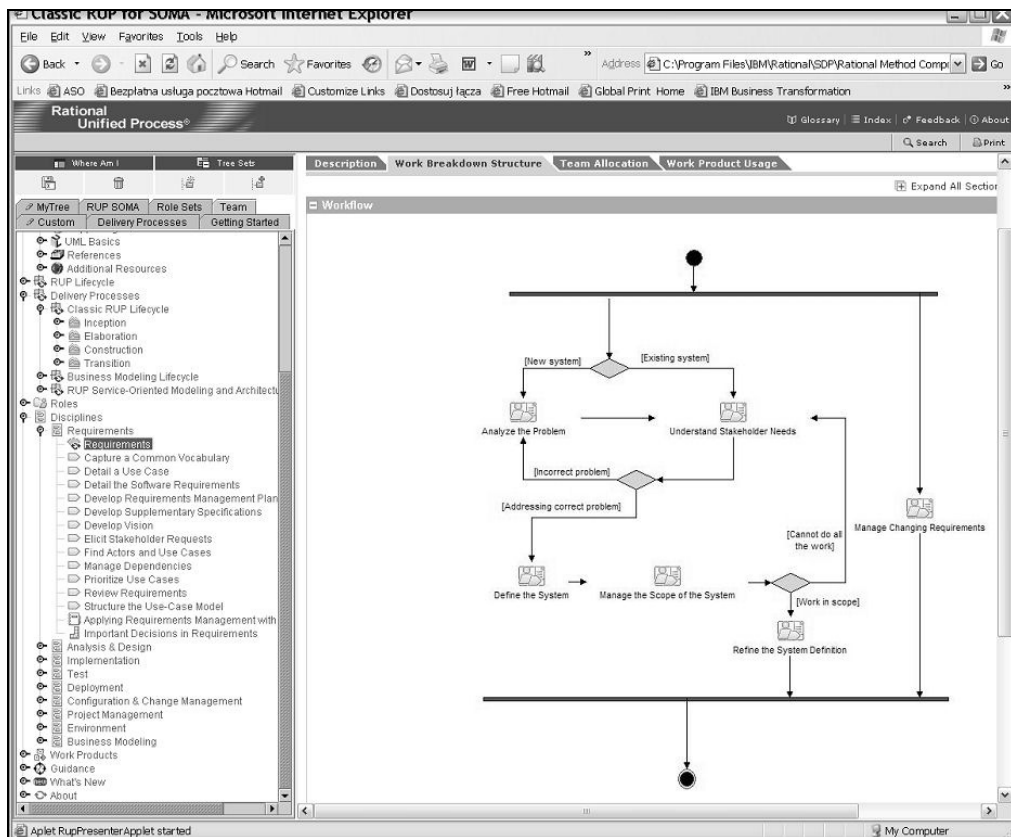
metodykami. Mogą to być zarówno metodyki zarządcze, takie jak Prince2 i PMBoK, jak i inne metodyki wytwórcze, takie jak adaptacyjny Scrum.

- ♦ **Balansuj** pomiędzy priorytetami decydentów (ang. *Balance stakeholder priorities*) — szukaj kompromisu między wymaganiami decydentów a techniczną opłacalnością wykonania. Warto zauważyć, że w praktyce decydenci rezygnują z wielu „fanaberii”, kiedy pokazany zostanie ich koszt i pojawi się pytanie o planowane zyski z tych funkcjonalności. Sformalizowany proces zarządzania zmianą może pomóc przy filtrowaniu nieprzemyślanych pomysłów nowych funkcjonalności.
- ♦ **Comunikuj** (dbaj o komunikację) zespoły między sobą (ang. *Collaborate across teams*) — wszystkie projekty angażują ludzi o różnych specjalnościach, z różnych „światów” i finalny produkt będzie *tylko* lub *aż* tak dobry jak poziom wzajemnego zrozumienia. RUP bardzo szczegółowo definiuje poszczególne dyscypliny oraz wskazuje, która z nich odgrywa kluczową rolę w konkretnej fazie projektu.
- ♦ **Demonstruj** rozwiązania iteracyjnie (ang. *Demonstrate value iteratively*) — należy zawsze jak najszybciej zmierzać do pokazania finalnym użytkownikom choć części funkcjonalności, nawet jeśli nie jest jeszcze w pełni przetestowana (np. wersje alfa). Jest to możliwe dzięki iteracjom w Fazie Opracowania i Fazie Konstrukcji. Co ciekawe, RUP nie definiuje roli użytkownika końcowego i nie wskazuje żadnej osoby odpowiedzialnej za weryfikację, czy tworzone wymagania są zgodne z oczekiwaniami. Nieco zaskakujące jest np. to, że osoby odpowiedzialne za modelowanie biznesowe i wymagania „nie wykazują” zwiększonego zaangażowania na końcu Fazy Konstrukcji i Fazy Przekazania.
- ♦ **Elastycznie** zwiększaj poziom abstrakcji (ang. *Elevate the level of abstraction*) — konieczne jest utrzymanie jednej lub wielu warstw abstrakcji między klientem końcowym a kodem w postaci specyfikacji. Tak jak zostało wspomniane na początku, esencją RUP jest tworzenie specyfikacji wysokiego i niskiego poziomu, zanim jeszcze zostaną uruchomione prace programistyczne.
- ♦ **Fokusuj się** (skupiaj się) stale na jakości (ang. *Focus continously on quality*) — w trakcie każdego etapu należy stale zapewniać kontrolę jakości każdego z wytwarzanych i wytworzonych elementów. RUP-owa dyscyplina testowania jest odpowiedzialna właśnie za to zadanie.

Adaptacja RUP do potrzeb organizacji

Przy wdrażaniu RUP w obrębie danej organizacji bardzo pomocne może okazać się narzędzie *Rational Method Composer* (RMC). Umożliwia ono właściwe zdefiniowanie odpowiedzialności poszczególnych osób. Odbywa się to poprzez określenie ról, dyscyplin oraz produktów prac. Następnie elementy te są powiązane z konkretnymi procesami. Główną wartością dodaną RMC jest bogata biblioteka szablonów ról, dyscyplin, produktów, prac i procesów (rysunek 29).

RMC jest rozwijane na bazie wersji open source projektu Eclipse Process Framework. Umożliwia on wykorzystanie kolekcji predefiniowanych procesów tworzenia oprogramowania oraz kreowania nowych procesów. Projekt ten zawiera m.in. dedykowaną część dotyczącą metodyk adaptacyjnych — Open Unified Process.



Rysunek 29. Przykładowy ekran narzędzia Rational Method Composer (RMC)

Podsumowanie RUP

Z jednej strony, wybór metodyki RUP jest dobrym pomysłem dla wielu organizacji, ze względu na bardzo dużą elastyczność i możliwość łączenia z innymi metodykami. Z drugiej strony, wdrożenie RUP oznacza konieczność szczegółowego zdefiniowania wszystkich jego procesów. Jest to konieczne, gdyż różne osoby mogą odmiennie interpretować RUP. IBM oferuje możliwość zdobycia certyfikatu w obszarze tej metodyki², ale nie jest

² Od stycznia 2007 roku jest to egzamin *IBM Certified Solution Designer — Rational Unified Process 7.0*

on zbyt rozpowszechniony na rynku. Najprawdopodobniej dzieje się tak, ponieważ testy zawierają raczej pytania na wysokim poziomie ogólności, takie jak: „Proszę zaznaczyć 3 zalety podejścia iteracyjnego”³.

Rozmowa z...

Marcin Kumorek jest konsultantem, który od trzech lat pracuje w katowickim oddziale ABG. Brał udział w wielu dużych projektach dla sektora publicznego, telekomunikacyjnego i innych firm usługowych, głównie w obszarze systemów billingowych. Specjalizuje się w tematyce integracji systemów, migracji baz danych i ich projektowaniu.

Adam Koszla (AKO): Marcinie, jesteś konsultantem w ABG. Na czym właściwie polega twoja praca?

Marcin Kumorek (MKU): Większość moich obowiązków wiąże się z tworzeniem nowych architektur. Od jakiegoś czasu przejmuję też obowiązki kierownika projektu. To dość częsta sytuacja — niekoniecznie dobra — gdy pomysłodawca (architekt) dostaje do wykonania swój projekt.

AKO: Większość osób, które chcą robić coś sensownego, musi się „nauczyć nosić wiele czapek”). Jak realia tworzenia dużych projektów informatycznych w przypadku takiej firmy jak ABG mają się do metodyk, takich jak RUP czy Scrum?

MKU: Część prac to utrzymanie naszych dużych rozwiązań. Tutaj już od dawna trudno mówić o projektach, bo jest to raczej proces. Część projektów, z racji wymagań zamawiającego, prowadzona jest w bardziej formalny sposób, ale nie widziałem w ABG projektu prowadzonego ściśle wg metodyki RUP. Przeżyłem to wcześniej. Trzeba „przyciąć” ideę do konkretnego projektu.

AKO: Metodyka powinna służyć jedynie jako szablon, rama... Wiedza na temat dobrego prowadzenia projektu zawsze wyszłizgnie się z mniej (zbiór dobrych praktyk) czy bardziej formalnego (zbiór procesów) opisu.

MKU: Jest kilka podstawowych spraw, które są istotne przy każdym projekcie, i z nimi metodyka powinna dawać sobie radę.

AKO: Nie zmienia to jednak faktu, że — moim zdaniem — RUP to coś więcej niż tylko fajny, trzyliterowy skrót.

MKU: RUP wprowadził do projektów użycie UML-a. Reszta to — jak sam mówisz — opis dobrych praktyk. Uważam, że kluczowe jest, aby w każdym projekcie pojawiło się kilka zasadniczych faz, takich jak:

³ Przykładowy test 839 dostępny na stronach IBM <http://www6.software.ibm.com/cgi-bin/pwdown/public/httpd/certify/sam839.pdf>

- ♦ **definicja wymagań**, która jest podstawą każdego projektu i przyczyną największej ilości katastrof; metodyka musi wspierać poprawne ich definiowanie, zmiany, weryfikację wykonania,
- ♦ **analiza i projekt**, które są często zaniedbywane przy napiętych terminach i zbyt szybko przechodzi się do implementacji; metodyka w tym miejscu musi dość elastycznie reagować na zmiany projektu.

AKO: Moim zdaniem, RUP jest właśnie jedną z najbardziej elastycznych metodyk. Dlaczego mam wrażenie, że miałeś jakieś negatywne doświadczenie z nim związane...

MKU: Brałem kiedyś udział w projekcie, który został „zaRUPowany” na śmierć. Nie była to wina samej metodyki RUP, tylko tego, że był to pierwszy projekt z wykorzystaniem tej metodyki i w ogóle UML-a.

AKO: I jakie wnioski wyciągnęliście na przyszłość?

MKU: Pierwsze fazy bardzo nam się podobały — sformalizowane podejście oraz czas na przygotowanie analizy i projektu. Niestety, po przejściu do fazy konstrukcji pojawiły się kłopoty. Duża ilość zmian w projekcie wymagała sporego nakładu pracy przy poprawianiu przygotowanych wcześniej diagramów. W pewnym momencie nie robiliśmy niczego innego, tylko poprawialiśmy to, co już zostało zrobione.

AKO: Czyli projekt nie był „zaRUPowany”, tylko „zaUMLowany”?

MKU: Wtedy nie było jeszcze narzędzi, które wspierałyby automatyczną synchronizację modeli logicznego i fizycznego. Dziś myślę, że należało jednak wykonać to, co było zaprojektowane, a dokumentację poprawiać dopiero w następnej iteracji.

AKO: Poza tymi kwestiami, jakie widzisz inne kluczowe czynniki sukcesu projektów informatycznych?

MKU: Moje doświadczenie mówi mi, że kluczem do sukcesu nie są jakieś trzy czy więcej literek w nazwie przyjętej metodyki. Zwykle kluczem do sukcesu są ludzie.

AKO: Szczególnie małe 5 – 10 osobowe zespoły...

MKU: Lubię małe zespoły :). Optymalnie dobrana wielkość pozwala na to, by:

- ♦ zespół mógł zbierać się łatwo i często,
- ♦ członkowie zespołu mogli się łatwo i często komunikować,
- ♦ nie potrzeba było nikogo więcej, aby wykonać pracę.

AKO: Spotkałem się z opinią, że nie ma sensu organizować zespołów większych niż dziesięć osób, bo w takim przypadku trzeba rozbudowywać hierarchię w górę...

MKU: Zgadzam się z tą opinią. Trudno nadzorować pracę więcej niż dwunastu ludzi. Kilkunastu ludziom trudno się porozumieć. Jeśli trzeba więcej ludzi, zadanie należy podzielić i przydzielić małym zespołom.

AKO: Co oprócz wielkości zespołu?

MKU: Ci ludzie, których właśnie zebraliśmy, to jeszcze nie zespół :). Zespół trzeba zbudować. Najprostszym sposobem jest zgromadzenie ich w jednym miejscu i określenie celu do osiągnięcia. Jedność miejsca, czasu i celu buduje już jakieś poczucie „my-zespół”. Zazwyczaj na początku praca będzie toczyć się miło. Wiesz, nowy projekt, dużo sił i chęci. W tej fazie warto czasem sprawdzić, czy w tym entuzjazmie robimy to, co trzeba.

AKO: Potem nadchodzi okres, kiedy robota „jest ustawiona i maszyna pracuje”...

MKU: Druga faza to faza frustracji, czyli pierwsze niepowodzenia i konflikty. Maszyna jeszcze „nie pracuje” — właśnie się dociera. Wbrew pozorom jest to dość wartościowy moment. Teraz właśnie można zobaczyć, co można jeszcze poprawić, a czego nie było widać pod wcześniejszym entuzjazmem. Gdy maszyna „została już ustawiona”, na jakiś czas jest dobrze. Trzeba tylko uważać, żeby nie popaść w samouwielbienie ;). Ostatnim etapem powinno być świętowanie. W długich projektach (trwających więcej niż kilka miesięcy) bardzo ważne jest akcentowanie zakończenia kolejnych etapów.

AKO: A co w sytuacjach kryzysowych? Np. wyraźnego konfliktu między dwoma osobami w zespole?

MKU: W sytuacjach konfliktowych kierownik musi poznać ich przyczynę. Często można znaleźć rozwiązanie, z którego wszyscy będą zadowoleni. Dużo zależy od sposobu, w jaki komunikują się członkowie zespołu. W idealnym świecie koniec projektu oznacza dostarczenie założonych produktów. Po zakończeniu projektu pracownicy również powinni być lepsi, a zespół — coś zyskać.

AKO: A w nieidealnym z większym lub mniejszym bólem projekt jest przekazywany do działu wsparcia albo — co gorsza — utrzymywany przez ten zespół jako dodatkowe obowiązki. Zdarza się jeszcze czasami, że tenże zespół jest okrawany.

MKU: Koszmar, ale — niestety — czasem prawdziwy. W idealnym świecie, jeśli udało się stworzyć zespół, warto go utrzymać i zrobić nim kolejny projekt ☺. Należy tutaj zwrócić uwagę, aby przy końcu każdego projektu zebrać doświadczenia w taki sposób, by możliwe było ich korzystanie w przyszłości. Świetnym pomysłem są tutaj firmowe bazy wiedzy, portale intranetowe, wiki...

AKO: I spotkania post mortem. Tak, ten temat często przewija się na początku, ale często też na końcu projektu, ludzie są już tak zmęczeni, że nie mają ochoty analizować, co się tak naprawdę wydarzyło.

MKU: Rzeczywiście, mało uwagi przykładą się do skończonych projektów; zwykle jest już coś innego do roboty :(.

AKO: W trakcie realizacji projektu trzeba jednak w jakiś sposób „rozliczać” zespół z realizacji?

MKU: Tu podoba mi się podejście z metodologii Scrum, czyli krótkie spotkanie rano, w trakcie którego omawia się, co zrobiono wczoraj, co mamy dziś, i kanapki

zachęcające tych, co nie lubią spotkań ;). Warto też przewidzieć w punktach wyznaczonych przez kamienie milowe czas (1 – 2 dni) na raportowanie postępów projektu dla „góry”.

AKO: Kanapki sponsorowane przez firmę?

MKU: Są takie firmy :). Nie szukając daleko, np. DRQ.

AKO: Wiem, pracowałem w takiej, która w lodówce trzymała piwo dla pracowników. Są też takie, które wstawiły stół bilardowy, i pozwalają ludziom grać w godzinach pracy... problem w tym, że nie ma godzin pracy i często ludzie „żyją” w firmie.

MKU: To ma zalety i wady. Z jednej strony, ludzie lubią być w pracy i nie czują, że praca to przymus, a z drugiej, jak siedzą tam cały czas, brakuje im dystansu, jaki może zapewnić np. rodzina.

AKO: Poza tym, mamy do czynienia z nierównością społeczną i istnieje dość wyraźny podział na pracowników z rodziną i bez. No i tu pojawia się dość poważny problem, jak godzić rodzinę z pracą...

MKU: Tak naprawdę nie ma problemu. To tylko kwestia priorytetów. Są ludzie, którzy pracują, żeby żyć, i są ludzie, którzy żyją, żeby pracować :).

AKO: Najlepsze, że człowiek uczy się tej maksymy z wiekiem ;).

MKU: A najgorsze, że nie wszyscy potrafią się tego nauczyć.

Przykład Prince2 i RUP — BlogSerwis

W przykładzie przedstawiony został kontrakt na wykonanie usługi programistycznej podpisany między dużą korporacją realizującą projekty, zgodnie z Prince2, a mniejszą firmą wykonującą oprogramowanie, zgodnie z RUP.

Zleceniodawca „Książęta” — Prince2	Zleceniobiorca „Rapersi” — RUP
Dział handlowy dużej, międzynarodowej korporacji „Książęta”, która sprzedaje kompleksowe, sprzętowe rozwiązania sieciowe, zwrócił uwagę na fakt, iż konieczne jest zbudowanie społeczności internetowej wokół istniejących produktów firmy. Forum dyskusyjne firmy odnotowuje dużą ilość wejść, ale wiadomo, iż istnieje spora część specjalistów, którzy o wiele chętniej dzieliliby się wiedzą „pod własnym nazwiskiem”. Pojawia się grupa osób, która postuluje stworzenie blogosfery, w której będzie istniała możliwość założenia własnego bloga. W ramach organizacji pojawił się również bardzo ciekawy pomysł integracji tej usługi z wewnętrzną bazą wiedzy.	Średniej wielkości firma programistyczna „Rapersi” specjalizuje się w dedykowanych rozwiązaniach i posiada spore doświadczenie w obszarze rozwiązań webowych oraz bogatą listę referencji. Firma ma już w swoim dorobku wykonanie dużego portalu intranetowego dla jednej z większych korporacji. Portal ten dawał m.in. możliwość założenia wewnątrzfirmowego bloga każdemu pracownikowi firmy.

Kierownicy projektów utworzyli swoje dzienniki projektów na wzór dziennika kapitańskiego...

Zlecniodawca „Książęta” — Prince2

Zlecnio biorca „Rapersi” — RUP

4 stycznia 2008 (piątek)

Przygotowanie dokumentu o nazwie Zlecenie Przygotowania Projektu (ZPZ)

Cały proces zgodny z Prince2 zaczął się od przygotowania przez dział handlowy dokumentu Zlecenie Przygotowania Założeń Projektu (ZPZP), który opisywał powyższe powody uruchomienia projektu oraz szkic uzasadnienia biznesowego. W ZPZP pojawiły się prognozy liczby specjalistycznych błogów w perspektywie pięciu lat od uruchomienia usługi.

Planowana jest większa kampania mająca na celu aktywizację społeczności związanej z produktami firmy i projekt ten ma udostępnić m.in. internetową platformę komunikacyjną tej społeczności.

Od strony biznesowej głównym celem jest lepsze rozpowszechnienie wiedzy na temat produktów firmy. Dzięki temu planuje się zwiększenie sprzedaży wśród obecnych klientów. W trakcie zamykania sprzedaży dział handlowy jest zobowiązany dowiedzieć się, skąd klient czerpał kluczowe informacje na temat kupowanego produktu. Lista możliwych opcji zostanie wzbogacona właśnie o nowo powstałą blogosferę. Dzięki temu zapewnione zostaną mechanizmy informacji zwrotnej i analiza skuteczności przeprowadzanej kampanii.

Pewien czas po wdrożeniu systemu planowana jest redukcja personelu odpowiedzialnego za utrzymanie wewnętrznej bazy wiedzy (podobne informacje będą publikowane i utrzymywane przez ekspertów zewnętrznych). Osoby te zostaną przesunięte do innych zadań.

Wyczenia wskazują, że łącznie bezpośrednie, planowane korzyści finansowe to co najmniej 60 tys. zł rocznie. Przychód na takim poziomie będzie możliwy przez okres co najmniej trzech lat, co daje łączną kwotę planowanych zysków na poziomie 180 tys. zł.

Zleceniodawca „Książęta” — Prince2**Zleceniobiorca „Rapersi” — RUP****7 stycznia 2008 (poniedziałek)****Akceptacja ZPZ**

Pomysł został zaprezentowany wiceprezesowi firmy i został przez niego wstępnie zaakceptowany. Jako że już wcześniej miał on do czynienia z metodyką Prince2, szybko ustanowił Komitet Sterujący, który składał się z niego samego, kierownika działu handlowego i kierownika działu projektowego; tego ostatniego zobowiązano do wyznaczenia kierownika projektu.

11 stycznia 2008 (piątek)**Przygotowanie dokumentu Podstawowe Założenia Projektu (PZP)**

Nowy kierownik projektu pan Krzysztof szybko zorientował się w temacie i w ciągu tygodnia przygotował dokument Podstawowe Założenia Projektu (PZP). Zaproponował w nim realizację projektu przez jedną z firm zewnętrznych, posiadającą doświadczenie technologiczne w tym obszarze. Ze względu na planowaną integrację z istniejącymi rozwiązaniami postanowiono utrzymywać całą usługę wewnątrz firmy. Na przygotowania do projektu zaplanowano trzy tygodnie, w trakcie których:

1. Firma zewnętrzna dokona analizy wykonania I etapu z działem handlowym,
2. Zostanie zdefiniowana formuła odbioru wykonanej aplikacji (testy),
3. Firma zewnętrzna proponuje konfigurację sprzętową,
4. Zostanie podjęta decyzja, czy zakupiony sprzęt zostanie umiejscowiony w jednej z serwerowni firmy, czy tymczasowo w firmie zewnętrznej,
5. Zostaną oszacowane koszty utrzymania rozwiązania,
6. Zostaną ustanowione reguły wsparcia rozwiązania,
7. Zostanie zaplanowany harmonogram i budżet przedsięwzięcia.

Pojawił się pierwszy sygnał, że potencjalny zleceniodawca — „Książęta” — planuje wykonanie usługi BlogSerwis. Ze względu na duży potencjał współpracy dyrektor firmy „Rapersi” podjął decyzję, że prace analityczne zostaną uruchomione natychmiast i mianował wewnętrznego kierownika projektu, pana Radka.

Zleceniodawca „Książęta” — Prince2**14 stycznia 2008 (poniedziałek)****Przygotowanie DIP**

Komitet sterujący wraz z kierownikiem projektu Krzysztofem na półgodzinnym spotkaniu podjął decyzję o zainicjowaniu projektu. Kierownik działu projektowego poinformował, że do odbiorów konieczne jest zaangażowanie wewnętrznego działu testów. Oprócz testów przeprowadzonych przez firmę zewnętrzną, dział testów ma sprawdzić drugi raz jakość

finalnego rozwiązania, jego wydajność i zweryfikować zaproponowaną konfigurację środowiska produkcyjnego. Wstępnie ustalono, że zakupiony sprzęt powinien wystarczyć na pięć lat działalności, ale musi istnieć możliwość łatwego „dostawienia” kolejnych serwerów bez konieczności wielotygodniowych migracji. Ustalono, iż na okres inicjacji zespół projektowy podlegający kierownikowi projektu Krzysztofowi będzie składał się z:

- ♦ kluczowego użytkownika (dział handlowy) — 30% jego czasu,
- ♦ testera (dział projektowy) — 20% jego czasu,
- ♦ administratora (dział projektowy) — 10% jego czasu.

Po 3 tygodniach planowana jest pierwsza wersja Dokumentu Inicjującego Projekt, a po miesiącu — finalna.

Kierownik projektu zmobilizował firmę zewnętrzną, która w ciągu pierwszych dwóch tygodni pięciokrotnie spotykała się z kluczowym użytkownikiem, by doprecyzować zakres prac do wykonania. Kierownik projektu Krzysztof stworzył pierwszą wersję struktury produktowej i rejestr ryzyk, w którym wyszczególniono m.in. takie zapisy.

- ♦ Zleceniobiorca nie wykona prac na czas — zdecydowano, że punkty kontrolne będą zrealizowane jako wizyty kierownika projektu Krzysztofa z kluczowym użytkownikiem u zleceniobiorcy w celu weryfikacji wykonanych prac.
- ♦ Pierwsza instalacja systemu nie będzie w pełni płynna — w ramach umowy zobowiązano zleceniobiorcę do wykonania wewnętrznych testów instalacyjnych i instrukcji instalacyjnej oraz dezinstalacyjnej. W przypadku błędnej instalacji konieczne będzie proste przywrócenie środowiska do stanu pierwotnego.
- ♦ System nie będzie skalowalny — zleceniobiorcę zobowiązano do wykonania testów dla więcej niż jednego serwera produkcyjnego oraz zawarcia w instrukcji administracyjnej instrukcji procedury przejścia z modelu jednoserwerowego do wieloserwerowego.
- ♦ Błędy systemu produkcyjnego nie będą weryfikowalne po stronie zleceniobiorcy — zleceniobiorca zobowiązał się do utrzymania kopii systemu po swojej stronie, do celów testowych. Dodatkowo przed wygenerowaniem łąty systemowej przetestuje on we własnym zakresie jej jakość oraz sposób instalacji. Ustalono, że ze względu na niski poziom tajności danych cała baza produkcyjna może być okresowo kopiowana do zleceniobiorcy. Obie firmy mają podpisaną klauzulę o zachowaniu poufności.
- ♦ Możliwe problemy z utrzymaniem usługi produkcyjnej ze względu na brak doświadczenia administratorów, kiepskie łącza oraz kwestie bezpieczeństwa — podjęto decyzję o hotelowaniu serwerów poza firmą.

Za moment uruchomienia projektu ustalono datę podpisania umowy między zleceniobiorcą a zleceniodawcą. Oto inne kluczowe punkty harmonogramu.

- ♦ W czasie trzech miesięcy od momentu podpisania umowy zleceniobiorca dostarczy pierwszą wersję wewnętrznie przetestowanego funkcjonalnie rozwiązania.

- ♦ W ciągu kolejnych dwóch tygodni zleceniodawca wykona własne testy, zgodnie ze scenariuszem akceptacji, i przygotuje protokół odstępstw oraz zlecenie zakupu sprzętu na potrzeby środowiska produkcyjnego.
- ♦ W ciągu kolejnych dwóch tygodni zleceniobiorca jest zobowiązany do naprawy wszystkich „odstępstw”.
- ♦ W ciągu kolejnego tygodnia nastąpi instalacja systemu na serwerach produkcyjnych zleceniodawcy, we wskazanej firmie udostępniającej serwerownie (hotelującą serwery).

Po upływie czterech i pół miesiąca powinna zostać udostępniona pierwsza wersja rozwiązania produkcyjnego (ang. *Release Candidate*). Po odbiorze aplikacji zostanie podpisana oddzielna umowa, która będzie zobowiązywała zleceniobiorcę do wykonania określonych prac w ciągu kolejnych pięciu lat. Wstępnie ustalono tabelę poziomu świadczonych usług (ang. *Service Level Agreement*).

Tabela 3. Tabela SLA w projekcie BlogSerwis

	Zgłoszenia krytyczne	Zgłoszenia istotne	Pozostałe zgłoszenia
	Problem, który uniemożliwia normalną pracę z systemem lub blokuje jego funkcjonalności.	Problem, który powoduje duże niedogodności w wykorzystaniu systemu.	Wszystkie zgłoszenia, które nie należą do dwóch poprzednich grup.
Reakcja na zgłoszenie — zleceniobiorca odpowiada na e-mail i rozpoczyna pracę.	2 godz.	4 godz.	1 dzień
Naprawa — zleceniobiorca dostarcza rozwiązanie, które umożliwia pracę z systemem na dotychczasowych zasadach (ang. <i>work around</i>).	1 dzień	2 dni	3 dni
Rozwiązanie docelowe — w przypadku rozwiązań tymczasowych zleceniobiorca dostarcza finalną wersję przetestowanej łąty systemowej.	3 dni	1 tydzień	2 tygodnie

- ♦ Przekroczenie czasów reakcji będzie skutkowało karą w wysokości 0,1% za każdy dzień zwłoki.
- ♦ Ustalono wstępnie pewną stałą kwotę opłaty za jeden rok usługi serwisowej na powyższych warunkach, pierwszy rok świadczenia takiej usługi jest darmowy.

Wujek Dobra Rada — odcinek 5. „Zbrodnia i kara”



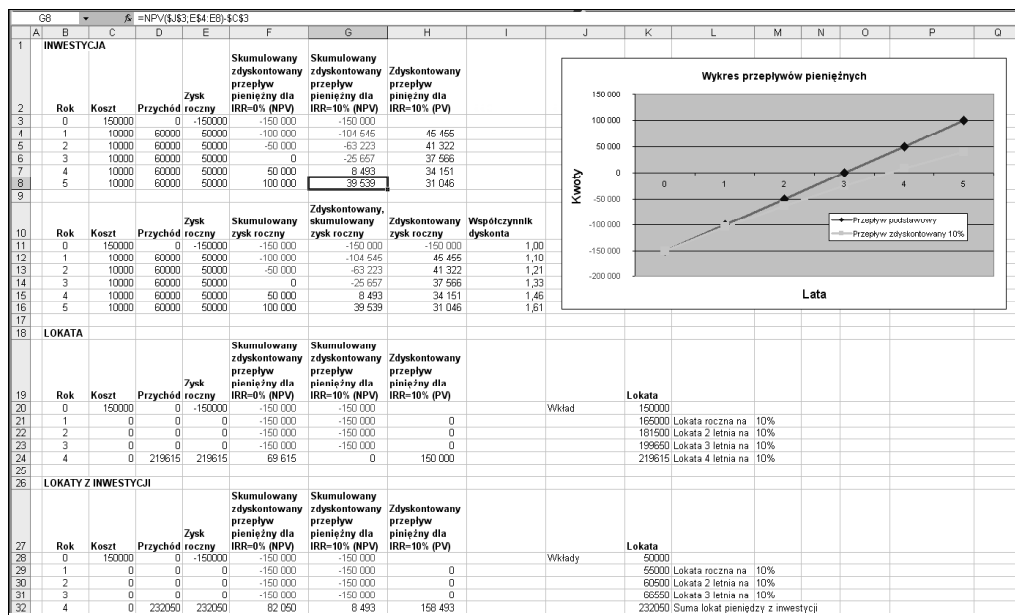
**PRAWNY ZAKAZ NIE MA SENSU BEZ OKREŚLENIA KARY SĄDOWEJ,
A ŚLĄ BEZ KAR FINANSOWYCH MOŻE STAĆ SIĘ MARTWYM ŚLĄ**

- ♦ Wśród zleczanych prac rozgranicza się zgłoszenia polegające na naprawie błędów i propozycje zmian. Propozycje zmian są rozliczane po stawkach zawartych w odrębnym aneksie umowy.
- ♦ Ustalono i zaakceptowano mechanizm akceptacji szacunków oraz raportowania czasu realizacji prac zleconych.
- ♦ Ustalono, że komunikacja będzie odbywać się drogą e-mailową. Zleceniobiorca udostępni e-mail support@firma.com.pl, który będzie odbierany przez co najmniej przez dwie osoby. Zleceniodawca ze swojej strony wskaże dwie osoby decyzyjne z działu handlowego, które będą miały prawo rejestrowania błędów i propozycji zmian oraz akceptacji lub renegocjacji przedstawionych wycen.
- ♦ Wszelkie prawa do kodów źródłowych przechodzą na rzeczy zleceniodawcy, ale nie może on wprowadzać do nich żadnych zmian pod groźbą utraty gwarancji.
- ♦ Budżet przedsięwzięcia został oszacowany na 150 tys. zł, w tym:
 - ♦ 90 tys. to usługi zewnętrzne,
 - ♦ 20 tys. to koszt zakupu sprzętu dla środowiska produkcyjnego,
 - ♦ 10 tys. to zakup licencji oprogramowania dla środowiska produkcyjnego,
 - ♦ 20 tys. to rezerwa budżetowa,

- ♦ 10 tys. to budżet zmian.
- ♦ Każdy kolejny rok utrzymania rozwiązania będzie kosztował 10 tys. zł, w tym:
 - ♦ 5 tys. to koszty utrzymania sprzętu,
 - ♦ 5 tys. to koszt kolejnego roku gwarancji zleceniobiorcy.

Uwaga: Zgodnie z polityką finansową zleceniobiorcy, w budżecie projektu nie są brane pod uwagę koszty własne.

Do dokumentu DIP dołączono diagram Zaktualizowanej Wartości Netto (ang. *Net Present Value* — NPV) — rysunek 30.



Rysunek 30. Diagram Zaktualizowanej Wartości Netto (ang. *Net Present Value* — NPV)

Zestawienie przelewów pieniężnych stworzono przy założeniu, że pierwszego roku ponoszone są główne koszty realizacji projektu, który jeszcze nie przynosi zysku, ale pierwszy rok gwarancji jest darmowy. Od drugiego roku pojawiają się planowane zyski, ale konieczne jest również poniesienie rocznych kosztów utrzymania. Grupa komórek F3:G7 została wyliczona z wykorzystaniem excelowej funkcji NPV. Przykładowo komórka G7 zawiera formułę:

$$= \text{NPV}(\$I\$2;E\$3:E7) - \$C\$2$$

Gdzie:

\$I\$2 to stopa dyskonta na poziomie 10%,

E\$3:E7 to poszczególne przepływy roczne,

\$C\$2 to pierwotny koszt inwestycji.

Należy zwrócić uwagę, że w przypadku kosztów inwestycyjnych w roku zerowym konieczne jest uwzględnienie takiego parametru poza standardową funkcją Excela NPV. Funkcja ta uwzględnia wyłącznie koszty w trakcie realizacji projektu.

Analogiczne obliczenia wykonano bez wykorzystania funkcji NPV w grupie komórek G10:I16.

Przepływ podstawowy (NPV dla $IRR = 0\%$) pokazuje zysk przyrastający w cyklu rocznym (zsumowany zysk roczny) i widać wyraźnie, że w takim przypadku inwestycja „zwróci się” równo po trzech latach.

Przepływ zdyskontowany bierze pod uwagę zmianę wartości pieniądza w czasie, przy założeniu, że traci on na wartości 10% rocznie (np. inflacja). Nawet w takim przypadku po nieco mniej niż czterech latach okaże się, że nasz projekt jest zyskowny (wartość NPV jest większa od zera dla $IRR = 10\%$).

Metoda NPV służy również do analizy opłacalności inwestycji w lokatę długoterminową. W przypadku odłożenia 150 tys. otrzymalibyśmy wartości zawarte w komórkach K20:K24. W komórce G24 widać wyraźnie, że dla wartości 219 615 NPV wyliczona dla czterech lat i $IRR = 10\%$ wynosi dokładnie 0.

Po czterech latach inwestycji, która zwraca rokrocznie 50 tys., NPV jest większa niż 0, czyli jest to lepsza inwestycja niż czteroletnia lokata na 10%. Co ciekawe, inwestycja zwraca łącznie 200 tys., a lokata 219 615; mimo to inwestycja jest opcją lepszą. Dlaczego? To właśnie esencja techniki NPV. Gdyby inwestycja przyniosła 200 tys. po czterech latach, byłaby mniej opłacalna niż lokata długoterminowa (NPV byłaby ujemna). Inwestycja przynosi jednak zyski rokrocznie i w grupie komórek H4:H8 widać wyraźnie, że 50 tys. po pierwszym roku jest o wiele więcej warte niż te same 50 tys. po czterech latach. Gdybyśmy corocznie zyski inwestowali w 10% lokaty długoterminowe, po czterech latach uzyskalibyśmy 232 050, a nie 200 tys. (lokata 3-letnia, 2-letnia, 1-roczna i zysk z czwartego roku). NPV z 232 050 w skali czterech lat dla $IRR = 10\%$ wynosi właśnie 8493 (komórka G32). O tyle po czterech latach inwestycja jest lepsza od lokaty długoterminowej.

Należy pamiętać, że z jednej strony, metoda NPV bardzo często demonstruje idealny przebieg wypadków, istniejący w subiektywnej wyobraźni twórcy i nie bierze pod uwagę ryzyk. Z drugiej strony, metoda NPV jako jedyna uwzględnia „koszt pieniądza” i umożliwia łatwe porównywanie wielu różnych inwestycji od strony opłacalności finansowej.

Do niedawna w branży IT trzyletnie inwestycje wewnętrzne dające $NPV > 0$ w skali trzech lat dla $IRR = 7\%$ były traktowane jako ciekawa inwestycja. Kryzys na rynku w latach 2008 i 2009 przyniósł jednak znaczącą zmianę i bardzo często planuje się już nie w perspektywie trzech lat, ale roku lub wręcz kwartału.

Zleceńbiorca „Rapersi” — RUP

14 stycznia 2008 (poniedziałek)

— uruchomienie Fazy Podjęcia

Uruchomiona została Faza Podjęcia projektu. Konsultant firmy „Rapersi” wybrał się pięciokrotnie w ciągu dwóch tygodni do kluczowego użytkownika firmy. W pierwszym i ostatnim spotkaniu brał udział kierownik projektu ze strony firmy „Rapersi” — Radek.

RUP nie definiuje sposobu zarządzania projektami, ale kierownik projektu Radek, zgodnie ze standardami firmy, założył kartę projektu jako excelowy plik.

Pierwsza zakładka tego pliku to strona tytułowa opisująca kluczowe cechy projektu, takie jak:

- ♦ nazwa,
- ♦ kierownik projektu,
- ♦ data rozpoczęcia,
- ♦ typ projektu,
- ♦ osoby zaangażowane,
- ♦ uwagi.

Druga zakładka to ewidencja dokumentacji projektowej, trzecia — rejestr ryzyk, a czwarta — kosztorys, który został zaprezentowany w tabeli 4.

Firma „Rapersi” wylicza marżę na trzech poziomach.

Marża pokrycia I stopnia analizuje zyski względem bezpośrednich kosztów związanych z realizacją projektu. W tym projekcie są to wyłącznie koszty związane z pensjami pracowników.

Marża pokrycia II stopnia uwzględnia dodatkowo koszty związane z kosztami obsługi prawnej oraz promocją i sprzedażą danego produktu. Klient w jakiś sposób dowiedział się przecież o tym, że firma „Rapersi” posiada kwalifikacje w obszarze rozwiązań webowych i został przekonany o tym, że jest to partner godny zaufania. Dział handlowy już od dłuższego czasu prowadził rozmowy z kluczowymi osobami w firmie „Książęta” na temat możliwości nawiązania współpracy.

Marża pokrycia III stopnia bierze pod uwagę wszelkie inne koszty niezwiązane bezpośrednio z realizacją poszczególnych projektów. Są to przykładowo koszty administracyjne, takie jak prowadzenie księgowości lub wynajem biura. Koszty te są równomiernie alokowane pomiędzy wszystkie projekty realizowane w firmie.

W danym roku firma „Rapersi” przyjęła pożądaną marżę pokrycia I stopnia na poziomie minimum 35%, co przekłada się średnio na marżę II stopnia na poziomie około 20%, a na marżę III stopnia — na poziomie około 10%. Dopiero ta ostatnia wartość jest rzeczywistym zyskiem netto.

Tabela 4. Przykładowy kosztorys zleceniobiorcy w karcie projektu BlogSerwis na 14 stycznia 2008 roku**ZESTAWIENIE SUMARYCZNE**

	Razem	sty 08	lut 08	mar 08	kwi 08	maj 08
Przychody	90000	0	0	0	0	90000
Koszty	55500	4500	8000	11000	13000	19000
Zysk	34500	-4500	-8000	-11000	-13000	71000
Marża pokrycia I stopnia	38,3%					

PRZYCHODY

Nr	Opis	Razem	sty 08	lut 08	mar 08	kwi 08	maj 08
		90000	0	0	0	0	90000
1	Sprzedaż oprogramowania	90000					90000

KOSZTY

Nr	Opis	Razem	sty 08	lut 08	mar 08	kwi 08	maj 08
		55500	4500	8000	11000	13000	19000
1	Konsultant	6000	3000				3000
2	Architekt programista	16000		4000	4000	4000	4000
3	Programista	21000		3000	6000	6000	6000
4	Tester	6000				2000	4000
5	Kierownik projektu	6500	1500	1000	1000	1000	2000

W tym projekcie marża pokrycia I stopnia wynosi 38,3%, więc kierownik Radek otrzymał oficjalną akceptację na rozpoczęcie realizacji projektu.

Stworzone zostało repozytorium na dokumentację projektową.

Zleceniodawca „Książęta” — Prince2**Zleceniobiorca „Raparsi” — RUP**

28 stycznia 2008 (poniedziałek)

Faza Podjęcia (RUP)

Po dwóch tygodniach została wykonana analiza biznesowa, która zawierała pełen model logiczny aplikacji i kluczowe wymagania. Analiza ta została zweryfikowana przez kierownika projektu Radka („Raparsi”) i przekazana do kierownika projektu Krzysztofa („Książęta”).

Po konsultacjach z architektami Radek opracował harmonogram realizacji projektu.

Zleceniodawca „Książęta” — Prince2**Zleceniobiorca „Rapersi” — RUP****11 lutego 2008 (poniedziałek)****Akceptacja DIP (Prince2)**

Okazało się, że osoba decyzyjna jest na zagranicznej konferencji, więc akceptacja dokumentu DIP została przełożona na czwartek.

14 lutego 2008 (czwartek)**Akceptacja DIP (Prince2)****Uruchomienie Fazy Opracowania (RUP)**

Zauważono, iż firma po raz pierwszy współpracuje z tym zleceniobiorcą, ale duże doświadczenie merytoryczne i niski koszt realizacji projektu są wystarczającym uzasadnieniem. Uzgodniono, że należy jak najszybciej podpisać umowę ze zleceniobiorcą. Podjęto również ostateczną decyzję, że zakupione serwery będą hotelowane w firmie zewnętrznej. Zaakceptowano dokument DIP i przyjęto próg tolerancji dla kierownika projektu Krzysztofa na poziomie 20%.

25 lutego 2008 (poniedziałek)**Podpisanie umowy**

Po dwóch seriach poprawek z obu stron udało się wreszcie podpisać umowę ze zleceniobiorcą. Dzięki temu, że kierownik projektu Krzysztof wiedział, iż spora część prac to gotowe komponenty, oraz podkreśleniu, że jest to początek długofalowej współpracy, udało się zbić ostateczną cenę z 90 tys. do 84 tys. złotych. Ustalono szczegóły umowy głównej oraz pięcioletniej umowy ramowej na wsparcie wraz ze stawkami za prace zleczone (propozycje zmian). Umowa regulowała również zasady zgłaszania, wyceny i akceptacji propozycji zmian.

Dzień podpisania umowy uznano za datę rozpoczęcia projektu. Kierownik projektu Krzysztof określił zarządcze punkty kontrolne, czyli daty weryfikacji planów projektowych.

11 marca 2008 (poniedziałek)**Plan Bazowy (RUP)**

Otrzymano wstępne informacje, że umowa z firmą „Książęta” jest w trakcie negocjacji — kierownik projektu Radek podjął decyzję o uruchomieniu drugiej fazy RUP, Fazy Opracowania. Zaangażowano do niej architekta i programistę, którzy zaczęli zapoznawać się z przygotowaną analizą. Wykryli kilka miejsc, które wymagały weryfikacji technicznej, i zaplanowali w tych obszarach wykonanie dowodów wykonalności.

PODPISALIŚMY UMOWĘ!!!

... na dostarczenie rozwiązania w ciągu trzech miesięcy. Kierownik projektu Radek uzgodnił już wcześniej z zarządem, że marża pokrycia I stopnia spadnie z 38% do 34%, ale za to udało się podpisać pięcioletnią umowę na wsparcie.

Zakończono prace nad projektem technicznym rozwiązania, który zawierał:

- ♦ architekturę rozwiązania,
- ♦ opis wykonanych dowodów wykonalności,
- ♦ plan projektu z deklaracją wykonania rozwiązania w czasie dwóch miesięcy,

Zleceniodawca „Książęta” — Prince2	Zleceniobiorca „Rapersi” — RUP
	<ul style="list-style-type: none"> ♦ plan jakości, który opisywał sposób wykonania testów i konieczne zasoby, ♦ listę możliwych i rozwiązanych ryzyk technologicznych. <p>Od 14 lutego wykonano również szereg prac programistycznych, które rozwiązywały kluczowe, potencjalne problemy.</p> <p>Kierownik projektu Radek podjął decyzję o weryfikacji tych prac przez innego architekta.</p>
<p>14 marca 2008 (piątek) Uruchomienie Fazy Konstrukcji (RUP)</p>	<p>Wykonana dokumentacja po drobnych zmianach została pozytywnie zweryfikowana i zatwierdzona przez drugiego architekta. Właściwe prace programistyczne były już w toku, ale dopiero tego dnia oficjalnie rozpoczęto Fazę Konstrukcji.</p>
<p>17 marca 2008 (poniedziałek) Przygotowanie planu testów (Prince2)</p> <p>Testerzy dostarczyli plan testów ze wstępnymi wymogami środowiska testowego, scenariusz akceptacyjny produktu oraz cztery scenariusze testowe przy zastrzeżeniu, że wykonywane będą również testy typu ad hoc.</p> <p>Kierownik projektu Krzysztof przekazał wszystkie materiały do wiadomości zleceniobiorcy.</p>	
<p>21 marca 2008 (piątek) Sterowanie Etapem (Prince2)</p> <p>Miała miejsce pierwsza wizyta kontrolna u zleceniobiorcy, który zaprezentował wykonany model logiczny, specyfikację techniczną oraz wykonane prace przygotowawcze.</p>	<p>W obecności kierownika Radka („Rapersi”) miała miejsce wizyta delegacji zleceniodawcy (kierownik Krzysztof i użytkownik kluczowy). Zaprezentowano wykonane prace.</p>
<p>11 kwietnia 2008 (piątek) Pierwszy szkielec aplikacji (RUP)</p>	<p>Zbudowano pierwszy, zintegrowany szkielec aplikacji i uruchomiono pierwszą iterację testów. Kierownik projektu Radek dokonał przeglądu wykrytych błędów i wraz z zespołem podjął decyzję, które z nich są naprawiane natychmiast, a które będą poprawione w końcowej fazie projektu.</p>

Zleceniodawca „Książęta” — Prince2	Zleceniobiorca „Rapersi” — RUP
25 kwietnia 2008 (piątek) Druga wersja aplikacji (RUP)	<p>Zbudowano drugą wersję aplikacji, która zawierała około 80% planowanej funkcjonalności. Uruchomiono kolejną iterację testów. Kierownik projektu Radek podjął decyzję o stopniowym włączaniu do testów konsultanta, który przygotowywał analizę systemu.</p>
28 kwietnia 2008 (poniedziałek) Sterowanie Etapem (Prince2) <p>Ze względu na urlop kierownika projektu Krzysztofa, sześć dni po planowanym terminie miała miejsce druga wizyta kontrolna, na której zaprezentowano pierwszą, kompletną wersję produktu. W trakcie prezentacji pojawiło się kilka błędów, ale ogólna koncepcja aplikacji była zgodna z oczekiwaniami.</p>	<p>W obecności kierownika projektu Radka miała miejsce druga wizyta kontrolna, na której zaprezentowano pierwszą, kompletną wersję produktu, z zaznaczeniem, że jest jeszcze na początku procesu testowego.</p>
9 maja 2008 (piątek) Zakończenie wykonywania funkcjonalności (RUP)	<p>Wykonano 100% planowanej funkcjonalności i wszystkie testy na systemie. Odnotowano dużą liczbę błędów. Po zapoznaniu się z charakterem i czasochłonnością naprawy kierownik projektu Radek oszacował, że nie będzie konieczności przesuwania terminu przekazania projektu.</p> <p>Tester został zobligowany do przygotowania instrukcji użytkownika przy współpracy z konsultantem.</p>
16 maja 2008 (piątek) Naprawa błędów (RUP)	<p>Naprawiono wszystkie kluczowe błędy w systemie i utworzono kolejną wersję oprogramowania.</p>
23 maja 2008 (piątek) Zamykanie Fazy Konstrukcji (RUP)	<p>Zakończono testy przyrostowe systemu i część testów wydajnościowych. Wykonano też pewne testy destrukcyjne, które miały na celu sprawdzenie odporności na „złośliwe” działania użytkowników. Dostępna była w pełni przetestowana wersja 0.1 (wiadomo było, że istnieją w niej pewne drobne błędy funkcjonalne) i wersja 0.2, w której błędy te zostały naprawione, ale jeszcze nieprzetestowane.</p>

Zleceniodawca „Książęta” — Prince2	Zleceniobiorca „Rapersi” — RUP
	<p>Po dyskusji z architektem, testerem i konsultantem kierownik projektu Radek podjął decyzję o dostarczeniu wersji 0.1 oprogramowania wraz z informacją o wykrytych i nienaprawionych błędach. Zadeklarował też dostarczenia wersji 0.2 w ciągu kolejnych dwóch tygodni.</p>
<p>26 maja 2008 (poniedziałek) Zarządzanie Wytwarzaniem Produktów (Prince2) Rozpoczęcie Fazy Przekazania (RUP)</p>	
<p>Zleceniobiorca zgodnie z ustaleniami dostarczył wyspecyfikowane rozwiązanie, ale zastrzegł, że posiada ono jeszcze pewne znane, drobne błędy funkcjonalne. Została przedstawiona lista tych błędów oraz poinformowano, że poprawiona wersja jest w trakcie testów wewnętrznych.</p> <p>Sprawdzono, że wykazane błędy nie mają istotnego wpływu na główną funkcjonalność systemu. Podziękowano za uczciwe i otwarte postawienie sprawy. Poinformowano również, że planowane są testy przed podpisaniem protokołu odbioru.</p>	<p>Konsultant pojawił się u zleceniodawcy z oprogramowaniem w wersji 0.1. Zleceniodawca poinformował o przyjęciu aplikacji do testów wewnętrznych i poinformował, że wersja 0.2 powinna zawierać również ewentualne poprawki błędów odnalezionych w trakcie tych testów.</p> <p>Oficjalnie zakończyła się Faza Konstrukcji i zaczęła Faza Przekazania.</p>
<p>27 maja 2008 (wtorek) Zarządzanie Wytwarzaniem Produktów (Prince2) Faza Przekazania (RUP)</p>	
<p>Miała miejsce pierwsza instalacja systemu i pojawiły się pierwsze niespójności, gdyż zleceniobiorca pracował na Windows 2003, a zleceniodawca oczekiwał platformy Windows 2000. Wykryto brak specyfikacji wersji systemu Windows. W związku z powyższym, konieczna była reinstalacja systemu testowego. Zwrócono uwagę na brak instrukcji instalacji systemu.</p>	<p>Konsultant zasygnalizował problemy z wersją systemu operacyjnego. Kierownik projektu Radek podjął decyzję o zaangażowaniu testera w reinstalację systemu testowego, tak aby w pełni odpowiadał środowisku produkcyjnemu. Tester został również zobligowany do przygotowania instrukcji instalacji systemu przy okazji tych działań.</p>
<p>29 maja 2008 (czwartek)</p>	
<p>Udało się z powodzeniem zainstalować pierwszą wersję systemu i zleceniobiorca wykonał krótkie testy, aby zweryfikować konfigurację.</p>	<p>Konsultant poinformował o pozytywnej instalacji systemu.</p>
<p>2 czerwca 2008 (poniedziałek)</p>	
<p>Postanowiono o zaangażowaniu zewnętrznego konsultanta, który dokona audytu bezpieczeństwa całego rozwiązania, aby zredukować ryzyko włamania. Osoba ta była związana z firmą ramową umową o współpracy i w związku z tym już następnego dnia miało miejsce wstępne spotkanie.</p>	

Zleceniodawca „Książęta” — Prince2	Zleceniobiorca „Rapersi” — RUP
5 czerwca 2008 (czwartek)	
Zewnętrzny konsultant, który wykonał audyt bezpieczeństwa całego rozwiązania, przekazał swoje zalecenia działowi testów.	
11 czerwca 2008 (środa)	
W międzyczasie wystąpiło kilka krytycznych sytuacji, które wymagały naniesienia łat systemowych, aby można było zakończyć cały proces testów. Równocześnie na bieżąco dostarczano w formie plików excelowych listę wszystkich wykrytych błędów. 11 czerwca oficjalnie zakończono prace nad protokołem rozbieżności oraz wynikami testów wydajnościowych. W testach uczestniczył również kluczowy użytkownik, który — oprócz rozbieżności — zdefiniował konieczność wykonania zmian funkcjonalnych. Kierownik projektu Krzysztof wynegocjował wykonanie tych zmian za cenę dodatkowych 4 tys. zł. Po stronie zleceniobiorcy konieczna była praca w weekend.	Zakończono pierwszą iterację testów po stronie zleceniodawcy, w trakcie której naprawiono krytyczne błędy i ostatecznie zainstalowano wersję 0.2 oprogramowania. Utworzono również wersję 0.3, w ramach której wykonano dodatkowo naprawę drobnych błędów funkcjonalnych oraz wprowadzono zaakceptowane zmiany. Wersja 0.3 została przekazana do testów wewnętrznych wykonywanych przez testera i konsultanta.
Po konsultacjach z administratorem złożono zapotrzebowanie sprzętowe na system produkcyjny.	
16 czerwca 2008 (poniedziałek)	
Dzięki wcześniejszym sygnałom o błędach programiści zleceniobiorcy mogli natychmiast naprawić wszystkie wykryte nieścisłości i dwa dni przed czasem wysłać kolejną wersję oprogramowania.	Zakończono pozytywnie wszystkie testy wewnętrzne i wersję 0.3 przekazano do zleceniodawcy.
18 czerwca 2008 (środa)	
Zarządzanie Zakresem Etapu (Prince2)	
Faza Przekazania (RUP)	
Niestety, okazało się, że tylko 2/3 nieścisłości zostało naprawionych właściwie. Niektóre z nich nie zostały naprawione, a implementacja innych jest dyskusyjna. Oprócz tego pojawiło się kilka dodatkowych zmian funkcjonalnych. Wynegocjowano wykonanie ich za dodatkowy 1 tys. zł.	Otrzymano informację zwrotną od klienta na temat wersji 0.3 i uruchomiono prace nad wersją 0.4.
23 czerwca 2008 (poniedziałek)	
Pojawiła się kolejna wersja oprogramowania oraz sprzęt dla środowiska produkcyjnego.	Po przyrostowych testach wersję 0.4 przekazano zleceniodawcy.

Zleceniodawca „Książęta” — Prince2	Zleceniobiorca „Rapersi” — RUP
25 czerwca 2008 (środa)	
W wersji 0.4 stwierdzono — co prawda — pewne drobne nieścisłości, ale dział testów zdefiniował tę wersję jako stabilną. Kierownik projektu Krzysztof postanowił o dopuszczeniu tejże wersji do eksploatacji.	Zleceniodawca zaakceptował przekazaną wersję i oficjalnie zakończono Fazę Przekazania.

Zleceniobiorca „Rapersi” — RUP

26 czerwca 2008 (czwartek)

Zakończenie Fazy Przekazania

Kierownik projektu Radek zaaranżował w salce konferencyjnej krótkie spotkanie robocze i poinformował o podpisaniu protokołu odbioru przez zleceniodawcę. Powiedział o rozpoczęciu procedury wsparcia, która będzie realizowana przez ten sam zespół w mniejszym wymiarze czasu. Przekazał podziękowania za efektywną pracę i poinformował, że projekt przyniósł firmie zysk i istnieje szansa premii za wykonanie.

Po spotkaniu z zespołem kierownik projektu uzupełnił kartę projektu włącznie z szablonem kosztorysu (tabela 5).

Tabela 5. Przykładowa realizacja kosztorysu zleceniobiorcy w karcie projektu BlogSerwis na 26 czerwca 2008 roku

ZESTAWIENIE SUMARYCZNE

	Razem	sty 08	lut 08	mar 08	kwi 08	maj 08	cze 08
Przychody	89000	0	0	0	0	84000	5000
Koszty	64100	4500	4000	14700	13000	18400	11500
Zysk	24900	−4500	−4000	−14700	−13000	65600	−6500
Marża pokrycia I stopnia	28,0 %						

PRZYCHODY

Nr	Opis	Razem	sty 08	lut 08	mar 08	kwi 08	maj 08	cze 08
		89000	0	0	0	0	84000	5000
1	Sprzedaż oprogramowania	89000					84000	5000

KOSZTY

Nr	Opis	Razem	sty 08	lut 08	mar 08	kwi 08	maj 08	cze 08
		64100	4500	4000	14700	13000	18400	11500
1	Konsultant	6900	3000				2400	1500
2	Architekt programista	16000		2000	4000	4000	4000	2000
3	Architekt 2	1200			1200			
4	Programista	24000		1500	6000	6000	6000	4500
5	Tester	8000				2000	4000	2000
6	Kierownik projektu	8000	1500	500	1500	1000	2000	1500

Marża na projekcie była niższa od przewidywanej, co wynikało z obniżonej wartości kontraktu i przedłużenia prac o jeden miesiąc. Kierownik projektu Radek zaznaczył w trakcie rozmowy ze swoim przełożonym, że sam wynegocjował dodatkową sumę za wykonanie pewnych zmian, co częściowo zrekompensowało fakt późniejszego odbioru projektu. Poza tym firma „Rapersi” powinna spodziewać się dodatkowych zysków z tytułu pięcioletniej, ramowej umowy wsparcia. Kierownik projektu zaproponował przeznaczenie jednej dziesiątej zysków na premie dla członków zespołu, które żartobliwie zostały nazwane napiwkami (kieszonkowe na piwo). Zaproponowano następujący podział tej kwoty:

Architekt programista — 1000 zł,

Programista — 700 zł,

Tester — 400 zł,

Konsultant — 300 zł.

Przełożony wyraził swoje zadowolenie z wykonania projektu, ale nie zgodził się na przyznanie premii, enigmatycznie uzasadniając to „ogólnym stanem finansowym całej firmy”.

Kierownik projektu Radek zarchiwizował całą dokumentację projektową na współdzielonym dysku oraz własnym dysku przenośnym i wybrał się na zasłużony urlop.

Zakończenie procesu wytwórczego po stronie zleceniobiorcy nie oznaczało zakończenia procesu po stronie zleceniodawcy.

Zleceniodawca „Książęta” — Prince2

26 czerwca 2008 (czwartek)

Sterowanie Etapem

Poinformowano firmę hotelującą serwery o planowanej dostawie systemu w dniu jutrzejszym oraz rozpoczęto instalację produkcyjną systemu.

27 czerwca 2008 (piątek)

Zainstalowano serwery w docelowym miejscu.

30 czerwca 2008 (poniedziałek)

Nastąpił start usługi w wersji beta — do testów zaproszono dziesięciu pracowników firmy oraz pięciu pracowników z firm partnerskich.

31 lipca 2008 (czwartek)

Zamykanie Projektu

Kierownik projektu Krzysztof wraz z kluczowym użytkownikiem i zleceniobiorcą podsumowali miesięczny czas pracy usługi beta. W tym okresie osoby, które uzyskały dostęp do systemu, założyły dwanaście blogów (jedna z osób założyła dwa blogi) i umieściły

czterdzieści dwa posty, zamieszczono także trzydzieści cztery komentarze. Krótka sonda wśród uczestników testów beta wskazywała na dobry odbiór inicjatywy i przyniosła wiele propozycji zmian, spośród których wybrano powtarzające się zgłoszenia.

W początkowym okresie zleceniobiorca po otrzymaniu e-maila oddzwaniał po około godzinie, aby omówić problem. Następnie, średnio po dwóch godzinach przysyłał e-mailem informację o rozpoczęciu prac lub odrzuceniu zgłoszenia. Z czasem zaczęto wykorzystywać współdzielony arkusz za pomocą serwisu GoogleDocs, w którym każde zgłoszenie zawierało następujące parametry:

- ♦ unikalny identyfikator,
- ♦ czas zgłoszenia,
- ♦ czas rozpoczęcia prac,
- ♦ czas decyzji o podjęciu,
- ♦ czas instalacji,
- ♦ czas zamknięcia,
- ♦ status zgłoszenia,
- ♦ tytuł,
- ♦ uwagi.

Plik ten był okresowo archiwizowany, mimo faktu iż serwis Google utrzymuje i udostępnia wszystkie wersje historyczne pliku. Bardzo pozytywnie odebrano również możliwość równoczesnej pracy dwóch użytkowników na jednym arkuszu. Jeżeli osoba A edytuje wartość w konkretnej komórce, osoba B widzi po prostu, że jej kolor się zmienia. Po zakończeniu edycji przez osobę A osoba B natychmiast widzi w tej komórce nową wartość.

Dzięki tym i innym funkcjonalnościom obie strony zaakceptowały ten sposób rejestracji zgłoszeń. Ze względu na relatywnie niską liczbę błędów stwierdzono, że nie ma sensu podejmowanie prób instalacji bardziej zaawansowanych narzędzi ani rozszerzanie czasu pracy serwisu poza zwyczajowe godziny, tym bardziej, że system nie jest krytycznym systemem firmy.

Realizacja całego projektu trwała około ośmiu miesięcy, a główne prace wytwórcze zaplanowane na cztery i pół miesiąca przeciągnęły się do pięciu miesięcy.

Z 150 tys. zaplanowanego budżetu wydano 141 tys. zł:

- ♦ 84 tys. to koszt podstawowych usług zewnętrznych,
- ♦ 5 tys. to koszty dodatkowych zmian pierwotnego zakresu zamówienia,
- ♦ 26 tys. to koszt zakupu sprzętu produkcyjnego (konieczny był dodatkowy zakup firewalla i routera),
- ♦ 16 tys. to zakup licencji oprogramowania na system produkcyjny — nie uwzględniono, że procesory będą wielordzeniowe i wpłynęło to na koszt licencji oraz oprogramowania zabezpieczającego, takiego jak antywirusy, automatyczne

detektory intruzów itp.; różnicę tę pokryto z budżetu zapasowego, decyzję podjął kierownik projektu Krzysztof,

- ♦ 3 tys. to koszt usług prawniczych,
- ♦ 5 tys. to koszt dwóch audytów bezpieczeństwa — pełnego i uproszczonego,
- ♦ 2 tys. to koszty szkolenia jednego z testerów i kluczowego użytkownika.

4 sierpnia 2008 (poniedziałek)

Kierownik projektu Krzysztof przedstawił raport z wykonania pierwszego etapu projektu i poprosił o pełne uruchomienie serwisu, na co otrzymał zgodę. W ramach kampanii mającej na celu aktywizację społeczności związanej z produktami firmy uwzględniono promocję nowej blogosfery i powiązano ją ze specjalnymi rabatami na wybrane produkty.

Komitet Sterujący podjął decyzję o tymczasowym zamrożeniu rozwoju systemu do momentu weryfikacji zyskowności całego przedsięwzięcia. Dział handlowy został zobowiązany do przygotowania sumarycznego raportu biznesowego obejmującego rok działalności projektu. Pomimo decyzji o wstrzymaniu rozwoju, zaakceptowano kilka drobnych propozycji zmian, które mają zostać pokryte z budżetu zmian. Przykładowo jedna z propozycji zmian dotyczyła ulepszonej grafiki portalu. Pozostała kwota pieniędzy została rozdzielona na premie — kierownik projektu Krzysztof otrzymał połowę tej puli, a druga połowa ma zostać rozdzielona przez niego między członków zespołu.

5 sierpnia 2008 (wtorek)

Kierownik projektu Krzysztof podpisał protokół odbioru całej aplikacji i wynegocjował 2 tys. zł za wykonanie dodatkowych, drobnych zmian.

8 sierpnia 2008 (piątek)

Kierownik projektu Krzysztof zamknął całą dokumentację projektową i wykonał dwie kopie na płycie CD — jedną do własnego użytku, a drugą dla biura projektowego. Następnie zaprosił wszystkich członków zespołu oraz kilku najaktywniejszych beta-testerów na uroczystą kolację, którą zasponsorował ze swojej premii, ze względu na skomplikowane procedury rozliczeniowe w firmie „Książęta”. W trakcie tego wydarzenia podziękował wszystkim obecnym i przy lampce szampana zainicjował krótką dyskusję, co można było zrobić lepiej, a co wypadło świetnie. Zadał również o to, aby w trakcie drugiej połowy wieczoru odbiec od tematów zawodowych. Na koniec spotkania pożegnał każdego z osobna i wręczył mu kopertę z osobiście podpisanym listem referencyjnym oraz premią:

- ♦ kluczowy użytkownik — 2 tys. zł,
- ♦ tester — 1 tys. zł,
- ♦ nagroda specjalna dla najaktywniejszego beta-testera — 0,5 tys. zł.

2 marca 2009 (poniedziałek)

Po pół roku od momentu produkcyjnego uruchomienia systemu podsumowano wyniki uruchomienia serwisu. Odnotowano, że akcja promocyjna, która polegała na udzielaniu zniżek najbardziej aktywnym członkom blogosfery, zadziałała i serwis posiada już około stu pięćdziesięciu aktywnych użytkowników, którzy opublikowali łącznie około pięćset artykułów. Jakość i charakter tych artykułów są bardzo różne, ale na razie tylko kilka razy pojawiały się sytuacje, w których dochodziło do konieczności ich cenzurowania. Członkowie społeczności samodzielnie zwołali już jedno spotkanie w świecie rzeczywistym, po godzinach pracy. Zgodnie stwierdzono, że udało się stworzyć dość ciekawą społeczność internetową, która zaczęła „żyć własnym życiem”.

Od strony handlowej podsumowano rachunek zysków (nowe sprzedaże) i strat (udzielone rabaty). Po pół roku działalności udało się odnotować zysk na poziomie około 20 tys. złotych. Jest to zysk nieco mniejszy od przewidywanego (60 tys. w skali roku), ale ku zaskoczeniu odnotowano, że blogosfera powoduje napływ pewnej grupy nowych klientów. Jeden z ankietowanych, nowych klientów napisał, że każde narzędzie posiada słabe i mocne strony. Dzięki publicznemu forum pojawia się możliwość dowiedzenia czegoś więcej niż to, co oferują standardowe materiały reklamowe, i można zorientować się, jak wygląda rzeczywista praca z oferowanymi narzędziami. Przyjęta polityka pełnej transparentności zwiększa zaufanie do oferowanych produktów.

W związku z powyższym Komitet Sterujący wyasygnował nowy budżet na rozwój blogosfery i poprosił o przygotowanie planu reformy komórki zarządzającej wewnętrzną bazą wiedzy.

Rozdział 4.

Microsoft Solution Framework (MSF)

MSF to metodyka wytwórcza, która łączy cechy modeli iteracyjnego i spiralnego. W praktyce różnice między podejściem RUP a MSF nie są tak duże, jak mogłoby się zdawać. Co więcej, metodyki te będą się prawdopodobnie do siebie upodabniać i główną różnicą pozostanie podstawowe środowisko programistyczne, czyli Eclipse dla RUP i Visual Studio dla MSF. Należy zaznaczyć, że MSF może być wykorzystywana w różnego rodzaju projektach i to nie tylko tych, które mają związek z środowiskiem Visual Studio. Nic nie stoi na przeszkodzie, aby zastosować tę metodykę w projektach wykonywanych w środowisku Eclipse. Dodatkowo MSF może być użyty w pracach infrastrukturalnych, wdrożeniowych itp.

Szczypta historii

MSF jest metodyką, która powstała w 1994 roku na bazie doświadczeń pracowników z pionu Microsoft Consulting Services. W początkowej fazie nie posiadała żadnych narzędzi wspierających, oprócz MS Project i Visio. To drugie narzędzie umożliwia tworzenie diagramów UML, ale — niestety — nie jest wygodne w praktycznym zastosowaniu. W przypadku generacji modeli fizycznych pewne sukcesy można odnotować jedynie przy modelowaniu struktury bazodanowej. O wiele bardziej uniwersalnym i praktycznym rozwiązaniem jest narzędzie Enterprise Architect firmy Sparx Systems.

W czerwcu 2003 roku pojawiła się trzecia wersja metodyki MSF, która jest w pełni dojrzałym modelem wytwarzania produktów. Jest to metodyka bardzo ciekawa, ale stosowana prawie wyłącznie przez osoby wykorzystujące narzędzia firmy Microsoft.

Wraz z wprowadzeniem na rynek linii produktowej Microsoft Visual Studio Team System pojawiła się czwarta wersja metodyki z dodatkowymi szablonami i pluginami, takimi jak:

- ♦ MSF dla metodyk adaptacyjnych (ang. *MSF for Agile Software Development*),
- ♦ MSF dla modelu CMMI (ang. *MSF for Capability Maturity Model Integration, Process Improvement*),
- ♦ Edytor szablonów procesowych (ang. *Process Template Editor*).

Najszybszym sposobem na wdrożenie tych rozwiązań jest zapoznanie się z istniejącymi szablonami oraz ich dostosowanie. Stworzenie własnego szablonu jest dość kosztownym przedsięwzięciem. Jeżeli mamy do czynienia z zespołem rozproszonym geograficznie lub standaryzacja jest celem nadrzędnym, warto zastanowić się nad MSF4. Inaczej „miękkie” wdrożenie MSF3 może być zupełnie wystarczające. Takie podejście jest atrakcyjne również ze względu na dużo niższe koszty, ponieważ w kwietniu 2009 roku Visual Studio Team System to wydatek około 26 tys. zł netto. Jest to cena o wiele wyższa od typowo wykorzystywanej wersji Visual Studio 2008 Professional, która kosztuje około 2,3 tys. zł netto. To właśnie dlatego postrzega się MSF w wersji czwartej jako „monetyzację” sukcesu całej metodyki¹.

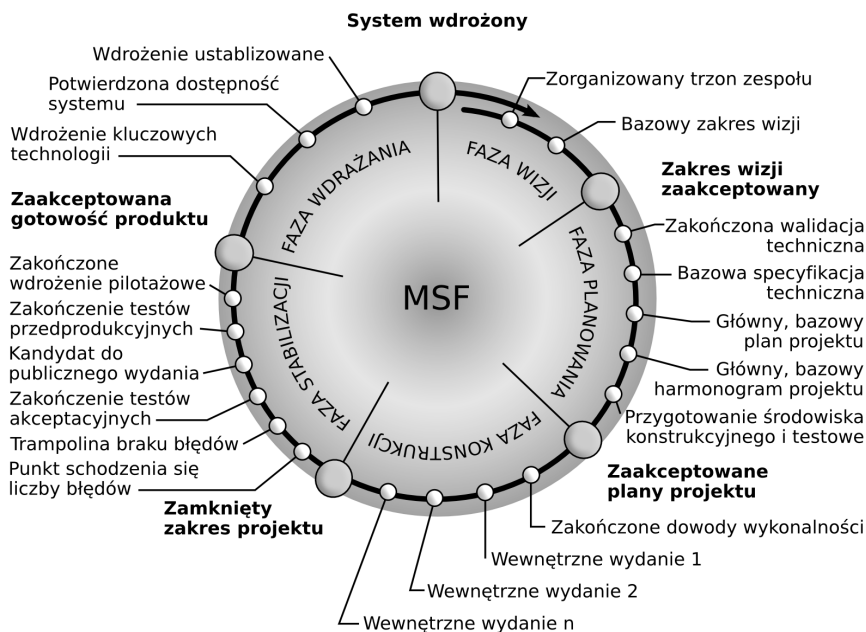
Proces

Wersja trzecia MSF składa się z pięciu głównych faz, z których każda kończy się realizacją kluczowego kamienia milowego. Oto one.

- ♦ Faza Wizji (ang. *Envisioning*) — okres organizacji zespołu i ustalenia wspólnej wizji projektu. W tym czasie określany jest jednoznacznie biznesowy zakres projektu.
- ♦ Faza Planowania (ang. *Planning*) — czas szczegółowych przygotowań do realizacji projektu, który zawiera opracowanie dowodów wykonalności i specyfikacji technicznej, określenie architektury rozwiązania, planu projektu i przygotowanie koniecznych środowisk pracy.
- ♦ Faza Konstrukcji (ang. *Developing*) — okres przyrostowego tworzenia rozwiązania.
- ♦ Faza Stabilizacji (ang. *Stabilizing*) — okres, w którym wszystkie siły skoncentrowane są na uzyskaniu właściwej jakości finalnego rozwiązania. W tym czasie funkcjonalność systemu nie powinna już być rozbudowywana.
- ♦ Faza Dostarczenia (ang. *Deploying*) — okres wdrożenia rozwiązania w docelowym środowisku (tzw. środowisku produkcyjnym).

Na rysunku 31. można zobaczyć, że każda z faz zawiera grupę cząstkowych kamieni milowych.

¹ Jest to blogowy głos Pawła Brodzińskiego, <http://blog.brodzinski.com/2006/07/msf-version-40-versus-version-31.html>



Rysunek 31. Architektura metody MSF

Model zespołu

MSF to także sześć kluczowych ról w projekcie, które odpowiadają sześciu kluczowym specjalizacjom.

- ♦ Zarządzanie programem (ang. *Program Management*) — *Kierownik Programu* jest odpowiednikiem kierownika projektów. W stosunku do innych metodyk nieco bardziej uwypuklona jest koordynacja kwestii technicznych.
- ♦ Konstruowanie (ang. *Development*) — *Konstruktor* jest odpowiedzialny za budowanie specyfikacji i prace wytwórcze.
- ♦ Testowanie (ang. *Testing*) — *Tester* odpowiada za nadzorowanie i akceptację jakości produktów.
- ♦ Projektowanie interfejsów użytkownika (ang. *User Experience*) — *Projektant Interfejsów Użytkownika* zajmuje się zaprojektowaniem ergonomicznego graficznego interfejsu użytkownika (ang. *Graphical User Interfaces Experience* — *GUI*), tak aby był przyjazny dla osób, które będą z niego korzystać (ang. *userfriendly*).

W wersji MSF4 rola ta uzyskała nową nazwę *Edukatora Użytkownika* (ang. *User Education*).

- ♦ Wdrażanie (ang. *Release Management*) — *Wdrożeniowiec* jest odpowiedzialny za instalację systemu w środowisku testowym i produkcyjnym, jego konfigurację i nadzór nad kolejnymi wydaniem.

W wersji MSF4 rola ta uzyskała nową nazwę *Zarządzania logistycznego* (ang. *Logistics Management*).

- ♦ Zarządzanie produktem (ang. *Product Management*) — *Kierownik Produktu* troszczy się o wartość biznesową finalnego produktu, określa docelową grupę klientów i ich wymagania.

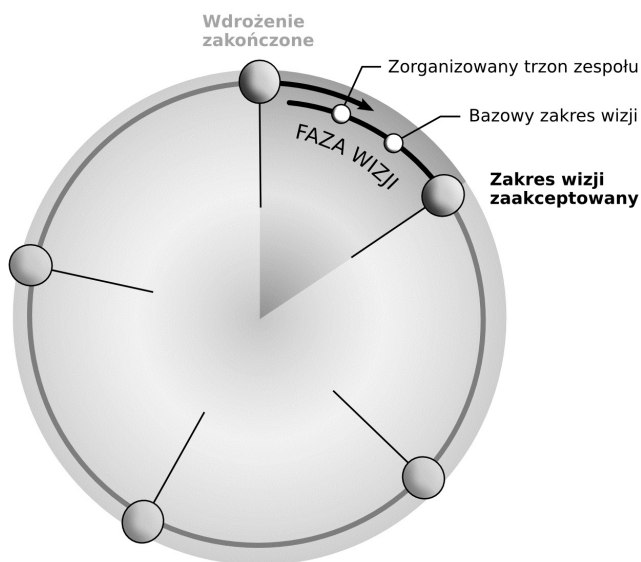
W poszczególnych fazach projektu każda z tych osób odgrywa nieco inną rolę.

Faza Wizji

W tym etapie (rysunek 32) muszą zostać zdefiniowane następujące elementy:

- ♦ Zakres wizji — wymagania biznesowe,
- ♦ Struktura projektowa — sposób zarządzania projektem, kwestie administracyjne, standardy, procesy, takie jak system kontroli wersji i zarządzania zmianą, zasoby i ograniczenia projektowe.
- ♦ Zarządzanie ryzykiem — wstępna identyfikacja i analiza ryzyk związanych z projektem oraz sposób zarządzania ryzykami.

Rysunek 32.
*Architektura Fazy
Wizji w MSF*



Oto przykładowa lista pytań, na które w tej fazie należy odpowiedzieć.

- ♦ Jakie są cele biznesowe?
 - ♦ Obniżenie kosztów utrzymania (ang. *Total Cost of Ownership* — TCO)?
 - ♦ Czy planowany jest zwrot z inwestycji (ang. *Return On Investment* — ROI)?
- ♦ Jaka jest ogólna misja całego przedsięwzięcia?
- ♦ Czy są narzucone jakieś ograniczenia, a jeśli tak, to jakie (np. czas realizacji, dostępność rozwiązania, niezawodność, bezpieczeństwo, zarządzalność, zgodność ze standardami)?
- ♦ Jaka jest ogólna koncepcja rozwiązania, jakie wymagania i czy określony jest strategiczny zakres projektu?
- ♦ Jakie problemy mają zostać rozwiązane?
- ♦ Jakie są profile użytkowników końcowych?
- ♦ Jaki jest stan obecny — aplikacje, bazy danych, infrastruktura, dokumentacja, środowiska, użytkownicy?
- ♦ Jeśli konieczna jest migracja, jaka jest jej strategia (np. okresowe współlistnienie dwóch rozwiązań, stopniowe zastępowanie nowym rozwiązaniem, plan przełączenia ze starego rozwiązania na nowe w „jeden weekend”)?
- ♦ Kto pełni kluczowe role w projekcie i jakie one są?
- ♦ Jakie są ryzyka projektowe wysokiego poziomu?

Na tym etapie kluczowe role to Kierownik Produktu (biznes) i Kierownik Programu (sprawy techniczne). To oni sterują całym etapem. Prace analityczne są wykonywane głównie przez Projektanta Interfejsów Użytkownika, który zbiera wymagania i przypadki użycia od przyszłych użytkowników systemu. Konstruktorzy pełnią raczej rolę konsultacyjną, głównie pomagając w definiowaniu stanu obecnego, z wyjątkiem sytuacji, gdy konieczne jest określenie strategii migracyjnej. Wdrożeniowiec sprawdza, czy przyjęte założenia będą możliwe do zrealizowania w praktyce.

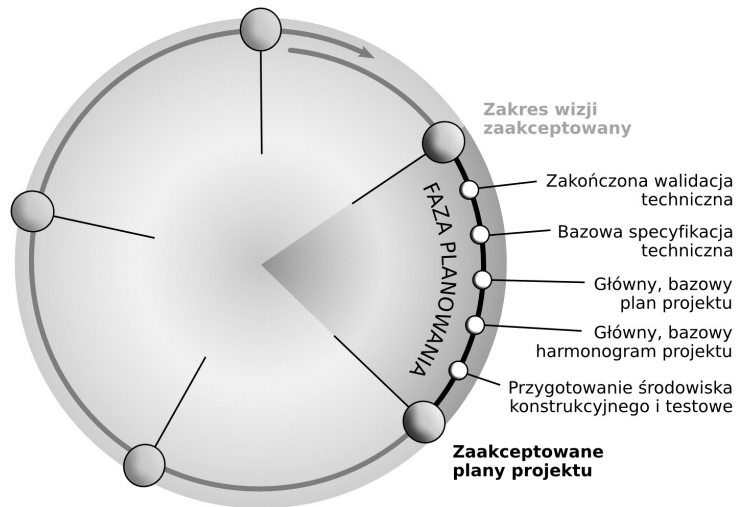
Faza Planowania

Po akceptacji wizji projektu zaczyna się okres szczegółowego planowania systemu (rysunek 33). W tej fazie główne cele to stworzenie:

- ♦ Specyfikacji funkcjonalnej, która zawiera wszelkie kluczowe kwestie projektowe, takie jak architektura rozwiązania; specyfikacja ta jest kontraktem na wykonanie pomiędzy zespołem projektowym a użytkownikami końcowymi oraz sponsorami projektu,
- ♦ Harmonogramu projektu,
- ♦ Zaktualizowanego rejestru ryzyk.

Rysunek 33.

*Architektura Fazy
Planowania w MSF*



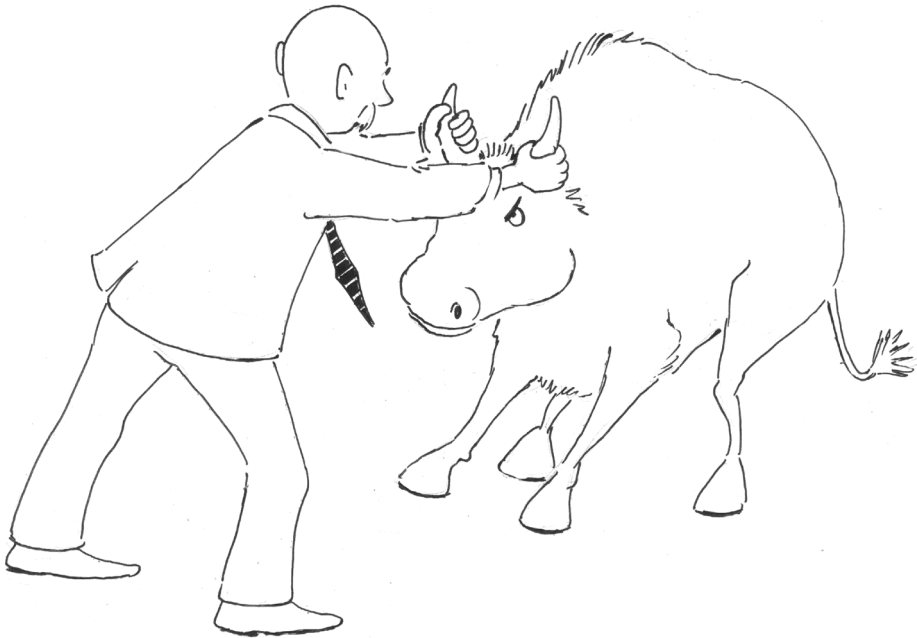
W fazie tej można wyróżnić trzy podetapy. Oto one.

- ♦ Projektowanie koncepcyjne — uszczegółowienie projektu z punktu widzenia biznesu i użytkownika,
(jest to odpowiednik modelowania na wysokim poziomie abstrakcji w metodyce RUP, czyli HLD).
- ♦ Projektowanie logiczne — stworzenie spójnego, logicznego modelu składającego się z obiektów i usług, który abstrahuje od fizycznego wykonania,
(jest to odpowiednik modelowania na niskim poziomie abstrakcji w metodyce RUP, czyli LLD).
- ♦ Projektowanie fizyczne — wstępny plan fizycznej realizacji projektu poprzez wybór właściwej technologii i komponentów oraz dowodów wykonalności, jakie należy wykonać.

W trakcie Fazy Planowania należy przeanalizować kwestie skalowalności, dostępności, niezawodności, wydajności i otwartości na współdziałanie (ang. *interoperability*). Warto wesprzeć się tutaj dokumentacją z biblioteki wzorców i praktyk Microsoftu (ang. *Patterns & Practices*).

Oto przykładowa lista pytań, na które w fazie tej należy odpowiedzieć.

- ♦ Jeżeli konieczna jest migracja, jaki dokładnie jest stan obecny i planowany model przejścia do nowej wersji?
- ♦ Jaka jest architektura planowanego rozwiązania?
- ♦ Dlaczego wybrano te technologie, a nie inne?
- ♦ Jakie badania wykonano w celu weryfikacji, czy zaproponowane architektura i technologie są najlepszymi z możliwych?

Wujek Dobra Rada — odcinek 6. „Dowody wykonalności”

IM TRUDNIEJSZY WYDAJE SIĘ PROBLEM PROJEKTOWY,
TYM SZYBCIEJ NALEŻY ZASTOSOWAĆ METODĘ DWÓCH „UD”:
ALBO SIĘ UDA ☺, ALBO SIĘ NIE UDA ☹.

- ♦ Czy w trakcie procesu planowania pojawiły się jakieś wątpliwości dotyczące możliwości realizacji konkretnych koncepcji? Jakiego wykonano dowody wykonalności (ang. *Proof Of Concepts*)?
- ♦ Czy wiadomo, kto, co i kiedy będzie robił?
- ♦ Czy członkowie zespołu jasno zadeklarowali, że są w stanie wykonać otrzymane zadania w obrębie przydzielonego czasu i zasobów, czy może wymagają jakiegos dodatkowego wsparcia?
- ♦ Czy konieczne (planowane) są szkolenia członków zespołu w trakcie trwania projektu?
- ♦ Czy w planie projektu uwzględniono urlopy, święta i inne dni wolne oraz niezbędne zapasy czasu?
- ♦ Jaki jest plan konstrukcji, testów i wdrożenia?
- ♦ Jakie rodzaje testów, w jakiej kolejności i ilu iteracjach będą wykonane?
- ♦ Czy dostępny jest (będzie) sprzęt konieczny do konstrukcji i testów rozwiązania (środowisko testowe)?
- ♦ Czy zostało stworzone środowiska konstrukcyjne, czyli w przypadków projektów programistycznych środowisko programistyczne?

- ♦ Jaki jest plan backupów kodów źródłowych, środowisk i danych?
- ♦ Czy planowany jest okres pilotażowy, a jeśli tak, jakie są jego szczegóły?
- ♦ Czy zostały stworzone środowiska testowe i konstrukcyjne?
- ♦ Jakie wsparcie użytkowników końcowych jest planowane — help desk, call center, bazy wiedzy, szkolenia?
- ♦ Jakie są ryzyka technologiczne?

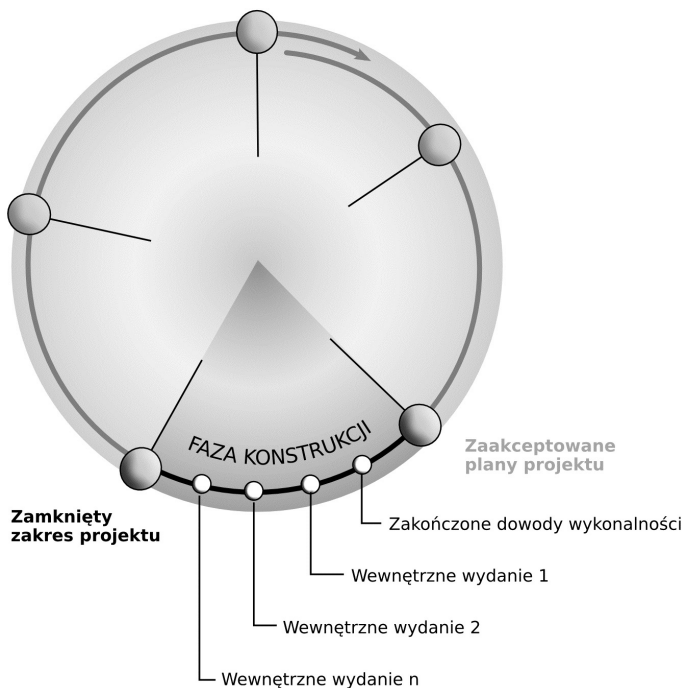
W tej fazie Kierownik Produktu (biznes) i Kierownik Programu (sprawy techniczne) płynnie oddają pałeczkę pierwszeństwa Konstruktorom, którzy przygotowują się do kolejnej fazy, czyli Fazy Konstrukcji. Są oni czynnie wspierani przez Projektanta Interfejsów Użytkownika i Testera, którzy kontynuują swoje prace analityczne. Wdrożeniu — więc uszczegółowia plany wdrożenia rozwiązania.

Faza Konstrukcji

Po akceptacji planów projektowych rozpoczyna się etap przyrostowej realizacji projektu (rysunek 34), zgodny z przygotowanym wcześniej harmonogramem.

Rysunek 34.

*Architektura Fazy
Konstrukcji w MSF*



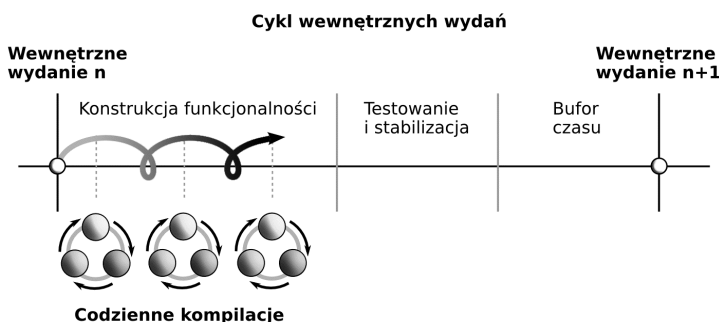
Kluczowe zadania na tym etapie to:

- ♦ Uruchomienie cykli produkcyjnych,
- ♦ Zbudowanie prototypu rozwiązania,
- ♦ Opracowanie kluczowych komponentów rozwiązania,
- ♦ Wykonanie narzędzi testowych i testów,
- ♦ Budowa rozwiązania docelowego,
- ♦ Przeglądy kodu (dla projektów programistycznych),
- ♦ Stałe zarządzanie ryzykami projektowymi.

W projektach programistycznych, oprócz decyzji dotyczących ilości wewnętrznych wydań rozwiązania, wskazane jest codzienne, automatyczne tworzenie kompletnych publikacji rozwiązania (ang. *daily builds*) — rysunek 35.

Rysunek 35.

Plan codziennych kompilacji w MSF



Idealne, codzienne tworzenie publikacji spełnia następujące założenia.

1. Proces budowania publikacji jest tworzony za pomocą skryptów napisanych w Maven, (N)Ant lub Cruise Control. Inną opcją jest sterowanie procesem budowania aplikacji z bardziej rozbudowanych środowisk, takich jak Automated Build Studio.
2. Weryfikowane jest wykonanie testów jednostkowych.
3. Weryfikowane jest wykonanie zautomatyzowanych testów funkcjonalnych.
4. Generowana jest dokumentacja techniczna z wykorzystaniem takich narzędzi jak NDoc czy JavaDoc.
5. Proces budowania aplikacji zawiera wygenerowanie instalatorów, które są dedykowanymi projektami stworzonymi za pomocą darmowych narzędzi lub aplikacjami, które powstały na odpłatnych platformach, takich jak InstallShield lub Wise.
6. Wdrożono narzędzia do statycznej analizy kodu (weryfikacja stylu pisania kodu, sprawdzenie, czy nie pojawiły się najczęściej popełnianie błędy lub złe praktyki) — fxCop, PMD, FindBugs.
7. Konkretna osoba przynajmniej raz dziennie weryfikuje wykonanie całego procesu i pielęgnuje go.

Codzienne tworzenie publikacji umożliwia testerom i innym zainteresowanym osobom pobranie najnowszego wydania rozwiązania. Daje to większą przezroczystość wykonywanych prac i możliwość ciągłej weryfikacji wykonanych zadań. Znacząco skraca się czas wykrycia usterki, a wiadomo, że im więcej czasu upłynęło od momentu popełnienia błędu do momentu jego wykrycia, tym gorzej. Sytuację tę można porównać do zdiagnozowania choroby we wczesnym stadium — w takim przypadku szanse wyleczenia są większe. Budowanie codzienne to przeświecanie stanu systemu każdego dnia!

Paradoksalne pytanie brzmi: „Czy system jest tego wart?”.

W projektach z otwartym kodem źródłowym i społecznością internetową, która tworzy taki projekt, wybór wydaje się oczywisty. Większość rozwiązań z Apache Foundation i wiele dostępnych w serwisie SourceForge funkcjonuje w takim właśnie reżimie.

Równocześnie większość systemów z nieotwartym kodem źródłowym nie posiada wdrożonych mechanizmów codziennego budowania kodu. Testerzy „wkraczają do akcji”, gdy programiści zasygnalizują koniec tworzenia grupy nowych funkcjonalności. Tego typu rozwiązanie może być najlepszym optimum między pożądanym kosztem a jakością.

Każdorazowo należy więc odpowiedzieć sobie na pytanie, czy opłaca się automatyzowanie procesu budowania i do jakiego stopnia. Jeżeli planujemy wykonanie dwóch do pięciu publikacji, to być może nie ma sensu. Jeśli jednak planujemy większą liczbę publikacji, warto zautomatyzować proces budowania do jednego kroku nawet wtedy, jeśli nie będziemy tworzyć publikacji codziennie. Dzięki temu uda nam się znacząco skrócić czas „od pomysłu do przemysłu” i zredukować koszt tworzenia kompletnej publikacji. Takie podejście przy budowaniu aplikacji zmniejsza również szansę popełnienia błędu krytycznego, takiego jak niedołączenie biblioteki lub błędne parametry poleceń wpisywanych w linii komend. Jak mówi reklama MasterCard: Automatyzacja procesu kompilacji — 5 tys., wysyłanie e-maila w przypadku błędów w wykonywaniu automatycznych testów — 7 tys., automatyczne skonstruowanie paczki instalacyjnej — 10 tys., wrażenie finalne — bezcenne.

Błąd systemu MultiBanku spowodował, że część przelewów zleczanych przez klientów tego banku między 8 a 11 sierpnia nie została zrealizowana. W polu odbiorcy pojawiały się bowiem dane nadawcy. To spowodowało, że część banków, do których były one kierowane, zwróciło je z powrotem. — Większość przelewów dotarła na rachunki odbiorców — przekonyuje Mateusz Żelechowski z MultiBanku. — Problem dotyczy około 1,2%, czyli 1 tys. przelewów realizowanych w poniedziałek 11 sierpnia².

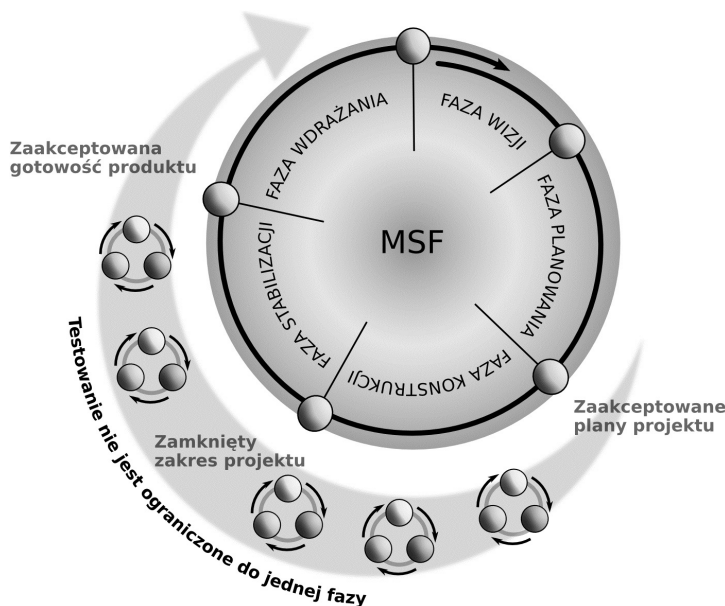
Czasami innowacyjność dostarczonego produktu może być ważniejsza od jego stabilność i dzieje się tak szczególnie w przypadku rozwiązań webowych z cienkim klientem. Wtedy naprawa błędu oznacza jedynie aktualizację plików na centralnych serwerach — koszt naprawy błędu jest stosunkowo niski. Dobrym przykładem jest tutaj Facebook, który świadomie wprowadza do użytkowania nieco „nieuczesane” rozwiązania. Posiada on rozproszony po całym świecie dedykowany zespół, którego głównym celem jest natychmiastowa minimalizacja efektów błędów, które pojawiają się w systemie.

² Monika Krześniak, „Błąd w systemie transakcyjnym MultiBanku”, Rzeczpospolita, 14-sier-2008, http://www.rp.pl/artukul/67572,176403_Blad_w_systemie_transakcyjnym_MultiBanku_.html

Należy pamiętać, że codzienna kompilacja nie zwalnia z konieczności zarządzania konfiguracją i wersjami. W praktyce praca testerów nie opiera się na codziennie generowanych wersjach. Najpierw programiści sami weryfikują przygotowane przez siebie prace i dopiero wtedy, kiedy zgłaszają, że przygotowane rozwiązanie jest gotowe do testów, następuje promocja do poziomu kandydata publicznego wydania (ang. *Release candidate*) — rysunek 36. Takie wersje nie powinny trafiać z środowiska konstrukcyjnego do środowiska testowego codziennie, ponieważ testerzy muszą mieć czas na kompletne przetestowanie wszystkich funkcjonalności, co zabiera zazwyczaj więcej niż jeden dzień. W analogiczny sposób programiści powinni mieć wystarczająco dużo czasu na naprawę wszystkich lub większości błędów wykrytych w trakcie testów poprzedniego kandydata. Oznacza to więc, że konieczne jest dobre zarządzanie konfiguracją i wersjami, aby w trakcie równoległych cyklów testów i naprawy błędów nie powstał bałagan.

Rysunek 36.

Codziennie kompilacje
a fazy MSF



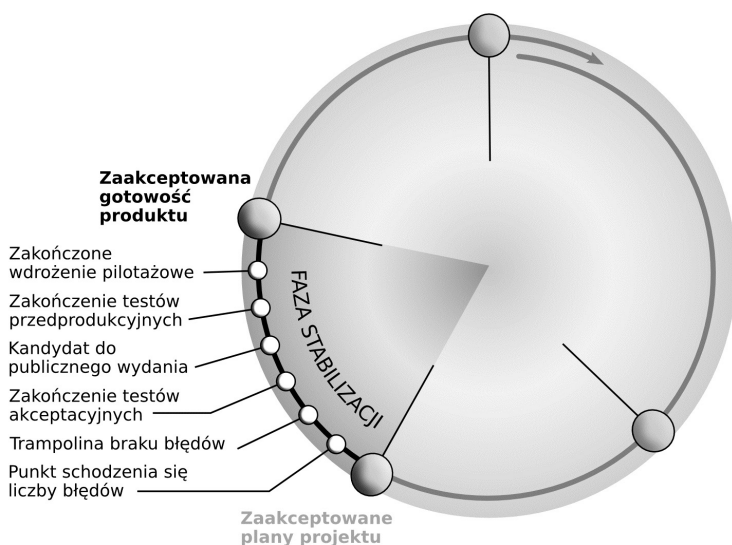
W Fazie Konstrukcji największa odpowiedzialność spoczywa na barkach Konstruktorów, którzy są wspierani przez Testerów przy testach funkcjonalnych systemu, badaniu wydajności, identyfikacji błędów, testach dokumentacji, testach planów migracyjnych, przygotowaniu raportów dotyczących postępów prac. Projektant Interfejsów Użytkownika jest odpowiedzialny za jednolity i właściwy wygląd graficzny całego rozwiązania oraz odpowiednie prezentowanie informacji i funkcjonalności. Może również wykonywać dokumentację szkoleniową i brać udział w testach, badając „używalność” rozwiązania. Kierownik Programu zarządza specyfikacją wymagań, monitoruje przebiegu projektu i ewentualnie aktualizuje plany. Kierownik Produktu odpowiada za komunikację z użytkownikami końcowymi. Wdrożeniowiec pełni rolę konsultacyjną.

Faza Stabilizacji

Po opracowaniu pełnego zakresu funkcjonalności zaczyna się okres „doszlifowywania” systemu i następuje skoncentrowanie się głównie na jego jakości (rysunek 37). Kluczowe zadania na tym etapie to testowanie rozwiązania i ciągła naprawa defektów systemu przez generowanie kolejnych kandydatów do publicznego wydania (ang. *Release Candidates*). MSF wymienia tutaj kilka ciekawych technik.

Rysunek 37.

Architektura Fazy
Stabilizacji w MSF



Sesje reanimacyjne

W ramach procesu wytwórczego pojawia się konieczność właściwego zamodelowania procesu zarządzania zgłoszeniami. W praktyce odbywa się to w ten sposób, że Kierownik Programu aranżuje spotkanie, zbierając w jednym miejscu kluczowe osoby spośród Konstruktorów oraz Testerów. Jeżeli oba zespoły są małe, można zaprosić je w całości. Następnie przeglądany jest rejestr zgłoszeń i wspólnie podejmowane są kluczowe decyzje. Taki mechanizm decyzyjny nazywa się sesjami reanimacyjnymi (ang. *Bug triage*). Angielskie słowo *triage* oznacza dosłownie „ocenę stanu zdrowia rannych”. Sesje reanimacyjne pierwszych wydań produktów faktycznie mogą przypominać kadry z serialu M.A.S.H., gdy z pola walki w Wietnamie dowożona jest helikopterami duża grupa rannych, amerykańskich żołnierzy³☺.

Decyzje podejmowane w trakcie sesji reanimacyjnych oznaczają zmiany stanów zgłoszeń w rejestrze zgłoszeń. Przykładowy diagram tych stanów oraz model prostej komunikacji zespołu Konstruktorów, Testerów i Kierownika Programu może wyglądać, tak jak na rysunku 38.

³ <http://www.regional.org.au/au/apen/2005/1/stockwellb.htm>



Rysunek 38. Przykładowy diagram stanów zgłoszeń

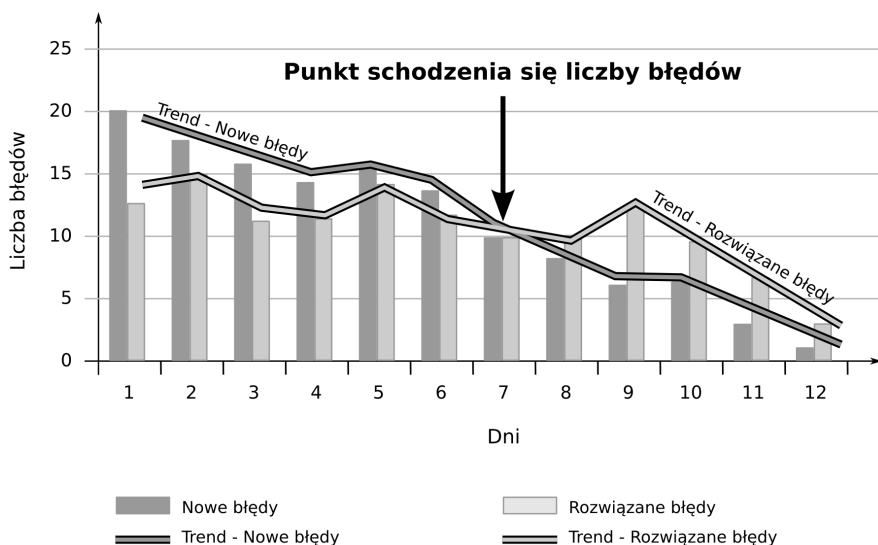
- ♦ Testerzy w sposób ciągły rejestrują zgłoszenia ze stanem „010 — Nowe” w systemie rejestracji błędów (ang. *Bug Tracking System*).
- ♦ W trakcie sesji reanimacyjnych podejmowane są decyzje o zmianie stanów zgłoszeń w obrębie stanów „010 — Nowe”, „032 — Odrzucone”, „051 — Na później” i „031 — Otwarte”.
- ♦ Programiści w sposób ciągły naprawiają zgłoszenia, które mają stan „031 — Otwarte”. Po wykonaniu naprawy modyfikują stan zgłoszenia na „080 — Zrobione”.
- ♦ Testerzy w sposób ciągły analizują zgłoszenia ze stanem „080 — Zrobione”. Po weryfikacji prace mogą zwrócić z powrotem do programistów ze stanem „031 — Otwarte”, jeżeli zostały wykonane niewłaściwie, lub zaakceptować i zmienić stan na „100 — Zamknięte”.

Monitorowanie stabilności rozwiązania

Jedyną z najistotniejszych umiejętności w kontekście Fazy Stabilizacji jest umiejętność oszacowania, jak wiele czasu zostało do jej zakończenia. Jest to szczególnie trudne w początkowych fazach projektu, kiedy szacowanie „kosztów stabilizacyjnych” jest obarczone dużym marginesem błędu. Nawet w trakcie testów często trudno odpowiedzieć, jak wiele czasu zajmie realizacja konkretnego scenariusza.

Należy okresowo wykonywać statystyki analizujące ilości błędów o poszczególnych statusach. W trakcie tej analizy można wyróżnić dwa ciekawe punkty.

- ♦ Punkt schodzenia się liczby błędów (ang. *Bug Convergence*) to pierwszy moment, w którym liczba błędów wykrytych staje się mniejsza od liczby błędów rozwiązanych. Punkt ten nie powinien być interpretowany jako połowa Fazy Stabilizacji, gdyż wiadomo, że ostatnie 10% większości zadań zabiera więcej niż 10% czasu. Jest to raczej moment, w którym zespół „dostrzega światelko w tunelu” i może bardziej miarodajnie szacować czas konieczny do zakończenia fazy. Przykład takiego punktu przedstawiono na rysunku 39.

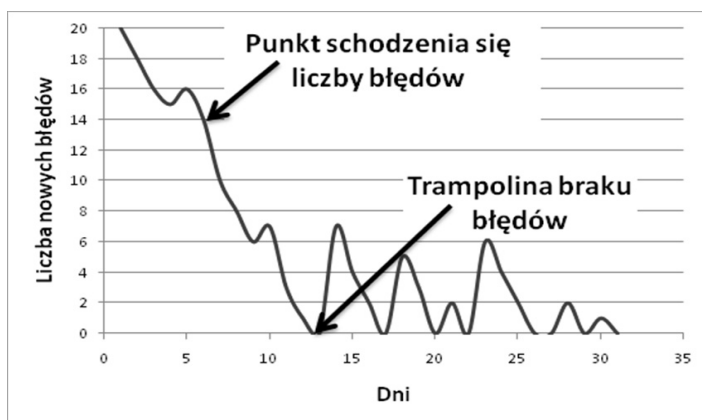


Rysunek 39. Punkt schodzenia się liczby błędów (ang. Bug Convergence)

- ♦ Trampolina braku błędów (ang. *Zero Bug Bogunce*) to pierwszy moment, w którym nie ma błędów aktywnych. Od tej chwili cały projekt zaczyna wychodzić na „ostatnią prostą”. Również i w tym przypadku nie ma reguły, która mówiłaby, że jesteśmy choćby w połowie drogi, ponieważ nie wiadomo, co przyniesie kolejna iteracja testów. Jest to jednak dobry moment, aby zacząć rozmawiać z klientem końcowym o wdrożeniu produkcyjnym. Przykład takiego punktu przedstawiono na rysunku 40.

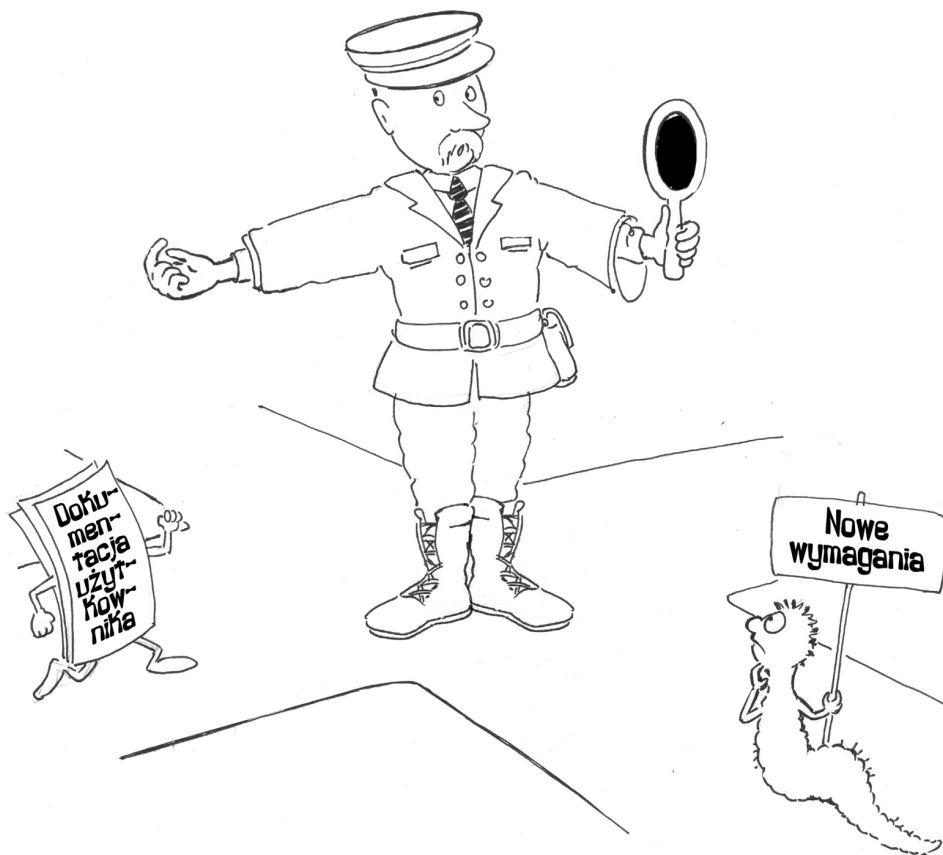
Rysunek 40.

Trampolina braku błędów
(ang. *Zero Bug Bogunce*)



Bardzo często po pierwszej prezentacji wersji demo u klientów pojawia się tzw. syndrom pełzających wymagań (ang. *scope creep*). Nagle klienci zdają sobie sprawę, czego naprawdę oczekują. O ile na początku Fazy Stabilizacji dopuszczalne są drobne zmiany funkcjonalne, o tyle w ostatniej iteracji testów powinny zostać naprawione wyłącznie istotne błędy, ponieważ zmiany rozwiązania powodują stałe ryzyko pojawienia się nowych błędów.

Wujek Dobra Rada — odcinek 7. „Twardy koleś”



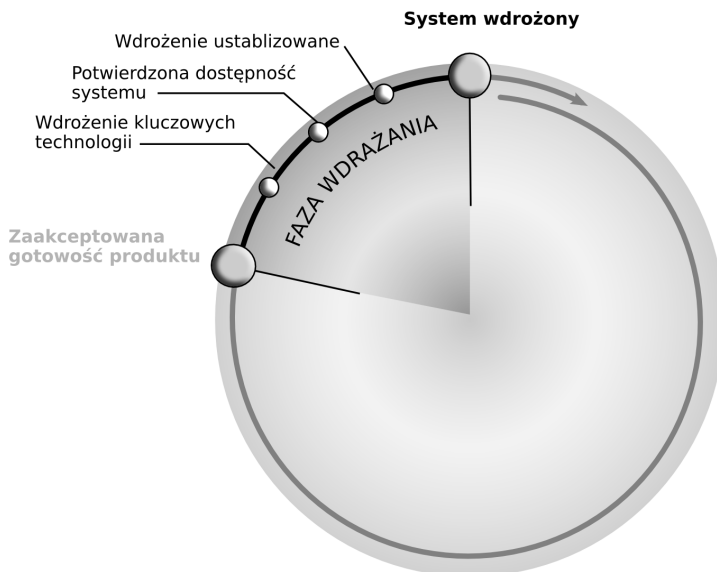
DOBRY KIEROWNIK POZWALA NA PEWNE ZMIANY PRZY PIERWSZEJ ITERACJI TESTÓW,
ALE PRZY KOLEJNYCH STOPNIOWO „TWARDNIEJE”

W trakcie Fazy Stabilizacji szczególnie kluczowa jest rola Kierownika Programu, który steruje sesją reanimacyjną i koordynuje pełzające wymagania. Kierownik Programu wraz z Kierownikiem Produktu powinni właściwie kanalizować nowe propozycje, kierując mniej konieczne do kolejnej iteracji projektu. Kierownik Produktu przygotowuje również plan uruchomienia produktu, co np. może mieć związek z kampanią marketingową. W tej fazie większość obciążenia przechodzi z Konstruktorów na Testerów, którzy walidują kryteria akceptacyjne, definiują błędy systemowe i odstępstwa od pierwotnych założeń. W tym okresie zakres funkcjonalny projektu jest zamrożony, ale Konstruktorzy mogą „polerować”, dokumentować i optymalizować rozwiązanie, o ile nie grozi to kolejnymi defektami. Projektant Interfejsów Użytkownika sprawdza, czy wykonane rozwiązanie jest zgodne ze specyfikacją, i przygotowuje materiały szkoleniowe oraz materiały dla końcowych użytkowników systemu. Wdrożeniowiec intensywnie tworzy plany wdrożenia, które mogą być testowane przy okazji instalacji kolejnych wersji oprogramowania.

Faza Wdrożenia

Faza Wdrożenia (rysunek 41) to okres przeniesienia rozwiązania do środowiska docelowego (produkcyjnego), ostatnie testy akceptacyjne po stronie klienta oraz podpisanie protokołu odbioru. Gdy planowany jest dalszy rozwój systemu, Faza Wdrożenia powinna spowodować płynne przejście do kolejnej Fazy Wizji. Kiedy planuje się długotrwałą eksploatację systemu, trzeba również pomyśleć o uruchomieniu działań utrzymaniowych zgodnych z metodyką Microsoft Operation Framework (MOF).

Rysunek 41.
*Architektura Fazy
Wdrożenia w MSF*



Przygotowując wdrożenie, warto utworzyć uszeregowaną listę weryfikacyjną (ang. *check list*), która zawierać będzie punkty analogiczne do poniższych.

- ♦ Czy posiadany sprzęt jest zgodny z wymaganym?
- ♦ Czy posiadane oprogramowanie jest zgodne z wymaganiami?
- ♦ Czy sprzęt i oprogramowanie są właściwie skonfigurowane?
- ♦ Czy instalacja finalnego rozwiązania przebiegła pomyślnie (ang. *smoke test*)?
- ♦ Czy finalna instalacja jest zaakceptowana przez klienta (ang. *acceptance test*)?
- ♦ Czy przeprowadzone zostały szkolenia? Czy sprawdzono, że użytkownicy znają system (np. egzaminy)?
- ♦ Czy wraz z rozwiązaniem dostarczona została cała konieczna dokumentacja?
- ♦ Czy całe rozwiązanie zapewnia założony poziom bezpieczeństwa, niezawodności i wydajności?
- ♦ Czy zostały przeprowadzone testy bezpieczeństwa, niezawodności i wydajności?

- ♦ Czy przeprowadzono analizę przyrostu danych w czasie i sprawdzono, jak system obsługuje większą ilość danych?
- ♦ Jakie są opcje powiększenia systemu produkcyjnego poprzez ulepszanie istniejącego sprzętu (ang. *scale up*) lub dodawanie nowego (ang. *scale out*)?
- ♦ Jaki jest czas odtworzenia systemu po awarii (testy backupu systemu)?
- ♦ W przypadku krytycznych systemów należy sprawdzić, czy awaria jednego elementu może spowodować kompletną katastrofę (ang. *No Single Point Of Failure*) — np. spalenie serwerowi lub nagła śmierć architekta?
- ♦ Czy przyjęty poziom świadczenia usług (ang. *Service Level Agreement*) jest realny?
- ♦ Czy kluczowe osoby znają procedurę obsługi zgłoszeń oraz sposobów ich eskalacji?

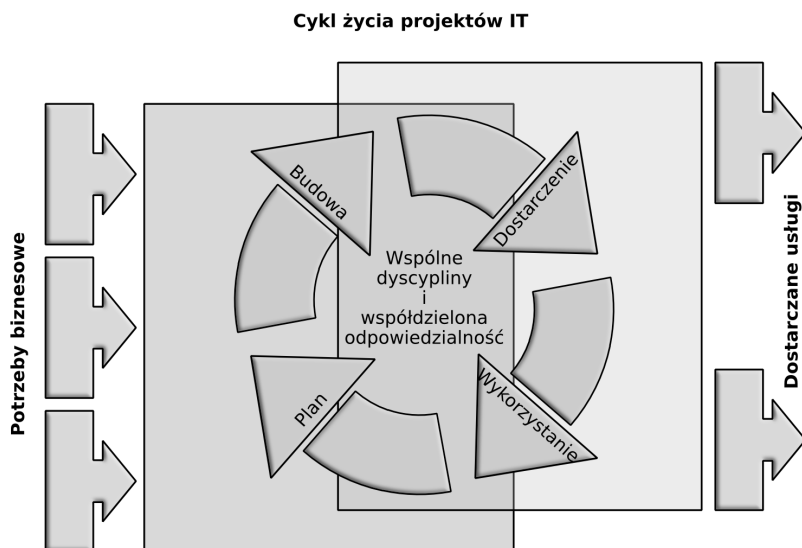
W tej fazie na pierwszą linię frontu wyrusza Wdrożeniowiec, który powinien uzyskać pełne wsparcie wszystkich pozostałych członków zespołu. Wzrasta tutaj również znaczenie Kierownika Produktu, odpowiadającego za finalne rozwiązanie przed klientem. Konstruktorzy są odpowiedzialni za rozwiązywanie wszelkich problemów, choć na tym etapie powinny to być już kwestie „kosmetyczne”. Ich precyzyjną identyfikacją i ewentualnym powtórzeniem w środowisku testowym zajmują się Testerzy, którzy mogą i powinni sugerować metody obejścia błędów poprzez właściwą konfigurację środowiska produkcyjnego. Kierownik Programu powinien głównie kontrolować, czy cały projekt nie wymyka się poza ustalony zakres oraz stale zmierza do coraz bardziej stabilnej wersji. Kierownik Programu prowadzi również „działania zamykające”.

MSF > MOF

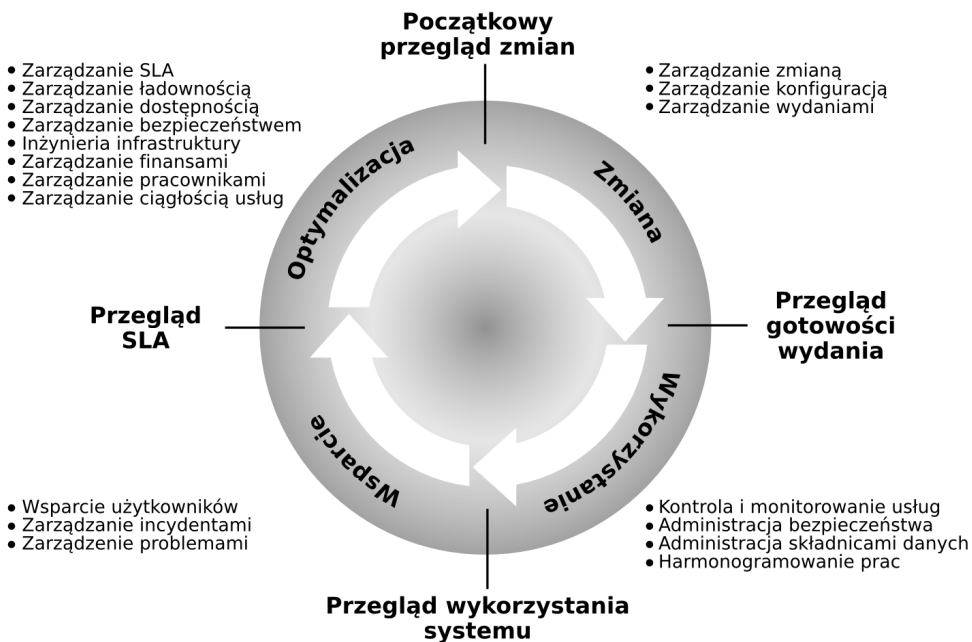
MSF to metodyka wykonania projektu, a Microsoft Operation Framework (MOF) to metodyka właściwego utrzymania projektu. Im większy system, im więcej krytycznych danych zawiera i im więcej danych przetwarza w czasie rzeczywistym, tym staranniej powinny zostać zaplanowane wszelkie zmiany oraz model migracji z wersji aktualnej do kolejnej. Wyróżnić można bieżące, codzienne utrzymanie systemu (MOF) i długofalową mapę jego rozwoju (ang. *Road map*). Taka mapa zawierać będzie kolejne duże pakiety nowych funkcjonalności, stopniowo realizowane z wykorzystaniem metodyki MSF i wprowadzane „do obiegu” przy użyciu metodyki MOF — rysunek 42.

MOF to podzbiór i modyfikacja najbardziej kluczowych praktyk IT Infrastructure Library (ITIL). MOF definiuje cztery główne usługi (rysunek 43). Oto one.

- ♦ **Zmiana** (ang. *Changing*), czyli procedura wprowadzenia nowej wersji rozwiązania, która dzieje się w tym samym czasie, co Faza Wdrożenia MSF. Usługa ta musi jednak brać pod uwagę procedurę szybkiej, programistycznej naprawy błędów krytycznych.
- ♦ **Wykorzystanie** (ang. *Operating*), czyli standardowe prace administracyjne zapewniające bieżącą operacyjność i bezpieczeństwo systemu.



Rysunek 42. *Współdziałanie MSF i MOF*



Rysunek 43. *Architektura MOF*

- ♦ **Wsparcie** (ang. *Support*), czyli procedury wsparcia użytkowników kluczowych oraz zarządzania zgłoszeniami.
- ♦ **Optymalizacja** (ang. *Optimization*), czyli udoskonalanie stanu bieżącego poprzez redukcję kosztów, poprawianie poziomu świadczonych usług, zarządzanie zmianami itp.

Trójkąt negocjacyjny

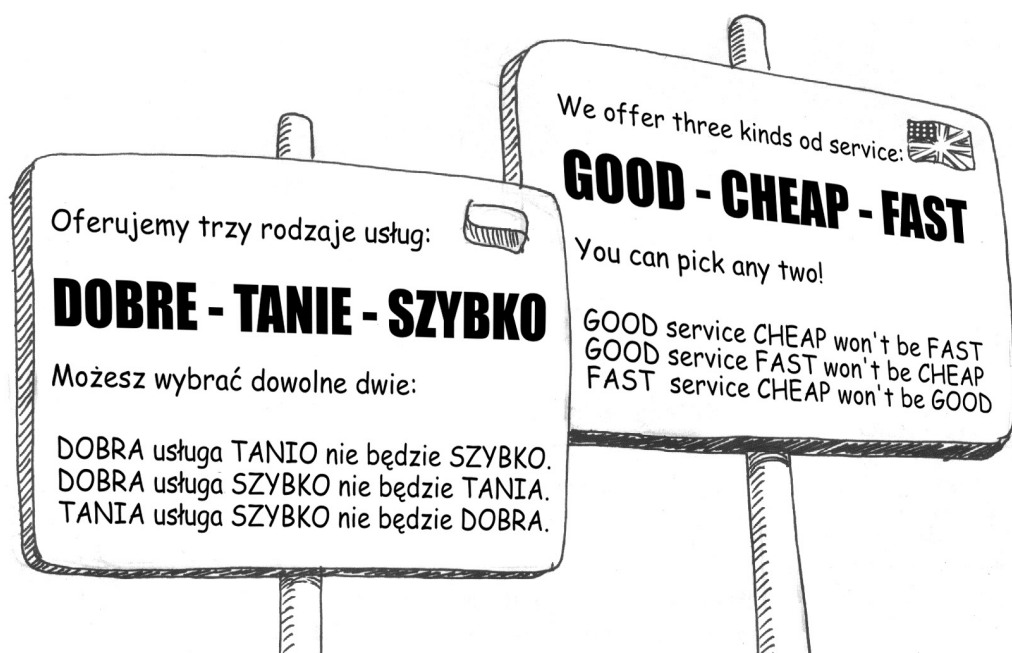
Charakterystyczną cechą MSF jest technika negocjacyjna opierająca się na „handlu” między trzema głównymi elementami: *funkcjonalnością*, *harmonogramem* i *zasobami* (ang. *trade-off*). Każdemu z tych elementów należy nadać jeden i tylko jeden z atrybutów: ustalony (ang. *fixed*), wybrany (ang. *chosen*) lub uzgodniony (ang. *adjustable*). Celem techniki jest sformułowanie zdania składającego się z trzech członów.

Przyjmując **ustaloną** liczbę **zasobów**, możemy **wybrać** *harmonogram* i **uzgodnić** zestaw *funkcjonalności*.

LUB

Przyjmując **ustalony** *harmonogram*, możemy **wybrać** *funkcjonalność* i **uzgodnić** liczbę *zasobów*.

Dobrym przykładem zastosowania tej techniki jest pewna amerykańska reklama (rysunek 44).

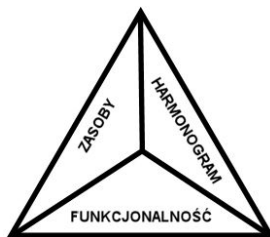


Rysunek 44. Amerykańska reklama zgodna z podejściem trade-off

Tego typu podejście skutecznie uczy, że „nie ma nic za darmo” — dodanie funkcjonalności powoduje konieczność zwiększenia zasobów i (lub) czasu. W takim przypadku trójkąt negocjacyjny powiększa się, ponieważ większy bok trójkąta o nazwie „funkcjonalność” powoduje zwiększenie boków o nazwach „harmonogram” i „zasoby” (trójkąt musi pozostać proporcjonalny) — rysunek 45.

Rysunek 45.

*Trójkąt negocjacyjny
MSF*

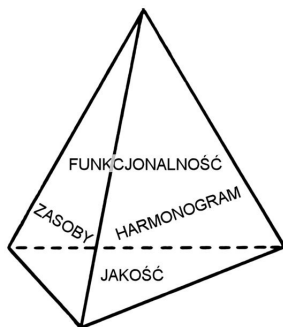
**W praktyce**

Niestety, cały mechanizm ma jeden słaby punkt, który trzeba brać pod uwagę. Można w miarę swobodnie żonglować czasem oraz funkcjonalnościami, ale zasoby to zazwyczaj ludzie. Istnieje pewien próg, powyżej którego dodawanie nowych osób nie przyspieszy projektu i nie zwiększy produktywności zespołu. 10 osób kopiących studnię nie wykopie jej 10 razy szybciej, a wręcz zaczną sobie nawzajem przeszkadzać. Dodatkowo, dodanie nowego pracownika przynosi wartość dodaną dopiero po pewnym czasie, a czasami może nie przynieść jej w ogóle. Tego typu kwestie należy mieć na uwadze, żonglując kwestiami zasobowymi.

Gdy jesteśmy postawieni pod ścianą i, mimo wyłożenia całej teorii, wymaga się od nas zapewnienia większej funkcjonalności przy tych samych zasobach i harmonogramie, można posłużyć się wycieczką w kierunku tetraedru (ang. *tetrahedron*) lub — mówiąc bardziej po ludzku — skorzystać z porównania do budowy piramidy. Jeżeli zmienimy punkt widzenia i spojrzymy na trójkąt negocjacyjny, tak jakby był piramidą widzianą z góry, możemy wyobrazić sobie jakość jako podstawę takiej piramidy (rysunek 46). W takim przypadku zwiększenie funkcjonalności powoduje dodatkowo zwiększenie wymogów jakościowych. Jeżeli nie zwiększymy nakładów na jakość, zaczniemy budować szpiczastą piramidę: „Naprawdę chcemy zbudować taką cieniutką piramidkę? Bo wydawało się, że planujemy zrobić piramidę, o której będą pamiętały pokolenia!” ☺.

Rysunek 46.

Tetraedr negocjacyjny

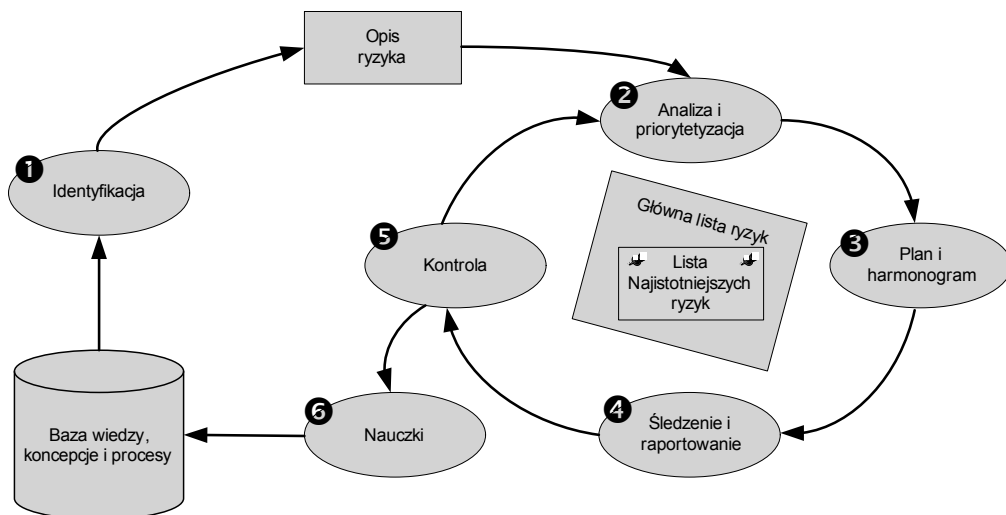


Dyscypliny zarządcze

Inną ciekawą cechą metodologii MSF jest wyróżnienie trzech dyscyplin zarządczych, które nawiązują do takich metodyk jak Prince2 czy PMBoK.

- ◆ Zarządzanie projektem w przypadku MSF jest w pełni sędowane na Kierownika Programu.
- ◆ Zarządzanie ryzykiem definiuje procedurę, za której przestrzeganie odpowiedzialny jest również Kierownik Programu. Procedura ta jest pewnym komponentem, który może i powinien zostać wykorzystany niezależnie od faz MSF.

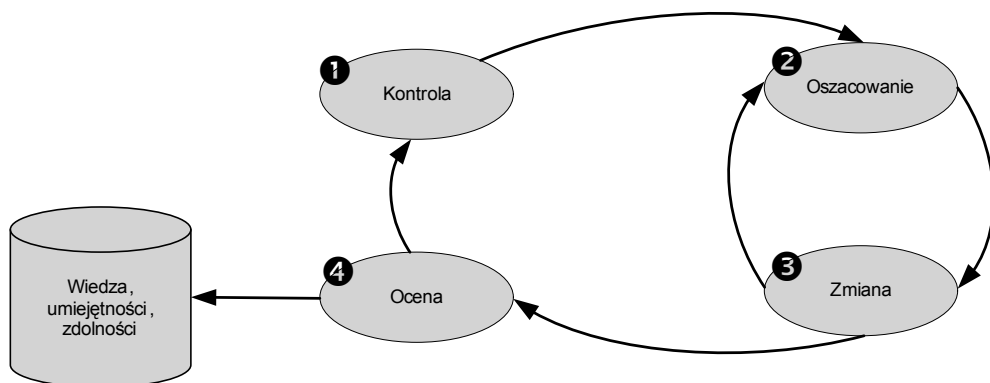
Jest to model podobny do mechanizmów zarządzania ryzykiem znanych z Prince2.



Rysunek 47. Zarządzanie ryzykiem w MSF

- ◆ Zarządzanie gotowością to mechanizm zarządzania kapitałem ludzkim, który każe poszczególnym osobom, zespołom i całej organizacji odpowiadać stale na pytanie: „Na ile posiadamy wiedzę, umiejętności i czy mamy możliwości wykonania zadań, do których się nas przydziela?”. Oprócz działań doraźnych, dyscyplina ta zawiera również elementy planowania strategicznego i ma zapewnić stałą adaptację do bieżących potrzeb, np. konieczność przekwalifikowania osób lub zespołów.

Tego typu podejście jest bardzo podobne do obszaru wiedzy „Zasoby ludzkie” w PMBoK.



Rysunek 48. Zarządzanie gotowością w MSF

Certyfikacja

Polityka certyfikacyjna firmy Microsoft wyróżnia dwie główne grupy specjalistów: programistów i administratorów. Istnieje przemyślany i skomplikowany system ścieżek certyfikacyjnych dla tych dwóch profilów. Każda osoba, która zda choć jeden certyfikat Microsoftu, uzyskuje stopień Microsoft Certification Professionalist (MCP). Jak łatwo zgadnąć, jest bardzo wiele takich osób — na kwiecień 2008 roku jest to dokładnie 8360 osób w Polsce i 2 mln 300 tys. osób na świecie⁴. Aby zdobyć jeden z wyższych stopni, należy wybrać którąś z dedykowanych ścieżek.

MSF jest głównym tematem egzaminu 70 – 301, który znajduje się zarówno na ścieżce Microsoft Certified Solution Developer (MCSO), jak i na Microsoft Certified System Engineer (MCSE). Oba certyfikaty są najwyższym stopniem certyfikacji dla programisty i administratora. System certyfikacji jest tak skonstruowany, że egzamin ten warto zdawać raczej jako ostatni z pięciu niż pierwszy. Nie wyklucza się jednak możliwości podejścia bezpośredniego do egzaminu 70 – 301 i uzyskania certyfikatu MCP.

Podsumowanie — MSF a RUP

RUP i MSF są w pewnym sensie konkurencyjnymi rozwiązaniami, ponieważ w ramach jednego zespołu projektowego nie ma sensu stosować obu metodyk. RUP jest szablonem naszkicowanym nieco bardziej ogólnie, w porównaniu z MSF, i dlatego ma większe możliwości adaptacyjne. Za to MSF daje nieco więcej praktycznych wskazówek projektowych. Ten akapit może być przedmiotem wielogodzinnej dyskusji między zwolennikami

⁴ Microsoft Polska oficjalnie udostępnia na swoich stronach aktualne „Statystyki certyfikatów” na Polskę i świat.

obu rozwiązań, szczególnie dlatego, że linia podziału pomiędzy „ciemną” (MS) i „jasną stroną mocy” (Java) jest powszechnie znana. Spoglądając jednak na cały temat dość chłodnym okiem, nie można nie zauważyć, że więcej tu podobieństw niż różnic.

Model zespołu z sześcioma specjalizacjami w MSF a dziewięć dyscyplin w RUP

Specjalizacja MSF	Dyscyplina RUP
Kierownik Produktu+Kierownik Programu	≈ Zarządzania Projektem
Konstruktor	≈ Konstruktor+Analityk (Projektant)
Tester	≈ Tester
Wdrożeniowiec	≈ Wdrożeniowiec+Zarządzanie zmianą i konfiguracją
Projektant Interfejsów Użytkownika	≈ Analityk biznesowy+Wymagania

Faktem jest, że w MSF brakuje dyscypliny Zarządzania Środowiskiem, która pojawia się w RUP

Pięć faz w MSF, a cztery fazy w RUP różnią się tylko wyróżnieniem Fazy Stabilizacji w MSF, ponieważ RUP przyjmuje, że jest ona zawarta w ramach Fazy Konstrukcji.

Trzy dyscypliny zarządcze, opisane w MSF jako odrębny fragment metodyki, znajdują swoje odzwierciedlenie w różnych fragmentach dokumentacji na temat RUP:

- ♦ (MSF) Zarządzanie projektem = dyscyplina Zarządzania Projektem (RUP)
- ♦ (MSF) Zarządzanie ryzykiem = Faza Opracowania (RUP)

Faktem jest, że w RUP brakuje „Zarządzania Gotowością” w rozumieniu MSF.

Oczywiście, każda pliszka swój ogon chwali i zarówno IBM, w porównaniu RUP z MSF, jest w stanie wykazać swoją wyższość⁵, jak i Microsoft potrafi udowodnić swoją wyższość nad RUP⁶. Nie zmienia to jednak faktu, że przy wyborze pomiędzy tymi metodykami mamy raczej do czynienia z kwestią gustu i stawiamy pomarańcze przeciw jabłkom — rysunek 49.

Rysunek 49.
RUP czy MSF?



⁵ <http://www.ibm.com/developerworks/rational/library/apr07/santos/index.html>
⁶ <http://social.msdn.microsoft.com/forums/en-US/vstsmsf/thread/2f43c39d-85fe-4034-8b7e-78232e0aa2e1>

Rozdział 5.

Przykład PMBoK i MSF — wdrożenie systemu BI

W przykładzie przedstawiony został kontrakt typu „ustalona cena” pomiędzy średnią firmą produkcyjną a średnią firmą konsultingową specjalizującą się we wdrożeniu gotowych rozwiązań klasy BI.

Klient „Bukowina” — PMBoK	Wykonawca „Mefisto” — MSF
<p>Średnia firma z branży meblowej podjęła strategiczną decyzję o wdrożeniu systemu typu BI w swoich sześćdziesięciu sześciu salonach oraz biurze centralnym. Rada nadzorcza zdawała sobie sprawę z powagi tego przedsięwzięcia i dlatego zatrudniła zewnętrznego konsultanta, który analizował potrzeby firmy w tym obszarze. Owocem jego miesięcznej pracy była wstępna koncepcja biznesowa przedsięwzięcia.</p> <p>Współwłaściciele bacznie obserwowali przebieg tej analizy i praca konsultanta zyskała w ich oczach bardzo duże uznanie nie tylko od strony merytorycznej, ale i organizacyjnej.</p> <p>W związku z tym postanowiono, że zostanie on mianowany kierownikiem projektu wdrożeniowego i uzyska duży stopień swobody w sposobie realizacji projektu. Z zadowoleniem przyjęto propozycję prowadzenia projektu w zgodzie z metodyką PMBoK.</p> <p>Zweryfikowano, że kierownik projektu Bartek posiada stosowne kompetencje.</p>	<p>Średnia firma z branży IT od wielu lat specjalizuje się we wdrożeniach istniejących rozwiązań klasy ERP, BI i innych. Firma otrzymała wiele znaczących nagród, ma ugruntowaną markę na rynku, udokumentowane wdrożenia u wielu klientów oraz certyfikowanych pracowników.</p> <p>W większości projektów wdrożeniowych wykorzystuje metodykę MSF. Ze względu na brak znajomości tej metodyki na rynku, pewne role nazywają się odmiennie od standardu:</p> <ul style="list-style-type: none">♦ Konstruktor jest nazywany konsultantem,♦ Projektant Interfejsów Użytkownika to analityk,♦ Kierownik Programu to kierownik projektu.

Cały projekt uruchamia się po stronie klienta „Bukowina”.

4 stycznia 2008 (piątek)

4.1 Dokument otwarcia

Na początku kierownik projektu uruchomił proces **4.1. Opracowanie dokumentu otwarcia**, w którym Bartek po serii rozmów ze dyrektorami firmy zanotował powody uruchomienia projektu. Oto one:

- ♦ lepsza kontrola wykonanych budżetów,
- ♦ możliwość bardziej precyzyjnego planowania budżetów,
- ♦ większa elastyczność analizy typu ad hoc,
- ♦ możliwość agregowania danych finansowych.

Planowany system BI będzie skupiał się na aspektach strategicznego, rocznego planowania i kwartalnej realizacji budżetu firmy. Dyrektorzy co roku ustalają roczne plany budżetowe, a każdy z salonów planuje własną sprzedaż na kolejny kwartał. Kontrolerzy finansowi są ogniwem spinającym te dwa światy i weryfikują zarówno plany roczne, jak i ich kwartalne wykonanie. Realizacja planów jest sprawdzana co tydzień poprzez pobranie i agregację danych z centralnego systemu ERP utrzymywanego przez dwóch administratorów.

Do dokumentu otwarcia załączono podsumowanie dotychczasowych prac koncepcyjnych.

Rada nadzorcza traktuje ten projekt jako konieczny z punktu widzenia strategicznego i główną intencją jest lepsza kontrola nad finansami firmy. Dlatego głównym celem całego przedsięwzięcia jest przygotowanie planów budżetowych na ostatni kwartał roku (od października do grudnia), biorąc pod uwagę dane zebrane do systemu z poprzedniego kwartału (od lipca do września). Celem drugoplanowym jest zwiększenie zysków firmy o 10% w ostatnim kwartale, w stosunku do zysku z ostatnich kwartałów w przeciągu pięciu ostatnich lat. Wiadome jest, że wdrożenie systemu jedynie częściowo może przyczynić się do osiągnięcia takiego wyniku.

Wdrożenie projektu musi zostać przeprowadzone siłami zewnętrznymi, ale system będzie utrzymany przez administratorów firmy Bukowina. Projekt uzyska bardzo wysoki priorytet wewnątrz firmy i kontrolerzy finansowi będą aktywnie uczestniczyć w opracowaniu i wdrożeniu systemu.

Kierownik projektu Bartek jest w pełni odpowiedzialny za wykonanie tego projektu przed radą nadzorczą. Odpowiada za:

- ♦ kontrolę realizacji poszczególnych zamówień,
- ♦ kontrolę jakościową wykonanych prac,
- ♦ zarządzanie zakresem projektu,
- ♦ przełożenie wdrożenia na wartość biznesową dla firmy,
- ♦ regularne raporty o stanie projektu.

Kierownik projektu Bartek ma duży stopień swobody w podejmowaniu decyzji, ale wskazane jest, aby informował zarząd o decyzjach kluczowych w projekcie i motywach ich podjęcia. Sugerowane jest bieżące monitorowanie kosztów projektu za pomocą techniki Zarządzania Wartością Wypracowaną (ang. *Earned Value Management* — EVM). Oczekuje się również, że w ciągu miesiąca wyznaczy co najmniej pięć łatwych do weryfikacji, ale wymiernych od strony biznesowej, kamieni milowych. W tych punktach zostaną zweryfikowane sposób i stopień realizacji zaplanowanego budżetu.

Wskazane jest wyznaczenie firmy audytorskiej, która wykona co najmniej pięć pisemnych ekspertyz, w tym minimum jedną dotyczącą weryfikacji ofert oraz cztery dotyczące realizacji projektu. Oprócz tego, dozwolone jest wykorzystanie określonej puli godzin konsultacji. Rada nadzorcza rezerwuje sobie prawo do wyłonienia jednej z trzech przedstawionych firm audytorskich, ale preselekcji dokonuje kierownik projektu Bartek. Po wybraniu firmy audytorskiej Bartek może pisemnie zakwestionować wykonaną ekspertyzę i podjąć własną decyzję.

Definitywnie cały projekt musi zostać wykonany do końca roku kalendarzowego, co oznacza, że musi funkcjonować produkcyjnie co najmniej kwartał, a idealnie byłoby, gdyby pracował dwa kwartały. Budżet firmy powinien być zarządzany właśnie za pomocą tego rozwiązania.

Kierownik projektu Bartek ma już pewne nieformalne informacje, w jakim zakresie kosztów powinien oscylować projekt.

Rejestr dokumentów jest składowany w konkretnym katalogu na laptopie kierownika projektu Bartka, ale firma udostępni dedykowane konto FTP, na którym będzie mógł utrzymywać swoją kopię bezpieczeństwa. W firmie jedynie dyrektorzy będą mieli dostęp do tego zasobu.

W przypadku tego projektu proces **10.1. Identyfikacja interesariuszy** był bardzo prosty, ponieważ po krótkiej dyskusji postanowiono, że jedynymi interesariuszami projektu będą dyrektorzy firmy, kierownik projektu i audytor zewnętrzny.

7 stycznia 2008 (poniedziałek)

Uruchomienie projektu

Kierownik projektu Bartek uzyskał zgodę na przedstawioną wizję projektu, która została dołączona do jego rocznego kontraktu. Ustalono również, że po zakończeniu tego projektu z powodzeniem (na czas, w budżecie i w zakresie) Bartek otrzyma premię w wysokości trzykrotnej pensji.

8 stycznia 2008 (wtorek)

Struktura pakietów roboczych (WBS)

Nowo mianowany kierownik projektu Bartek wiedział, że kluczową kwestią jest właściwy wybór bazowego rozwiązania i partnera wdrożeniowego. Decyzja musi zostać podjęta, zanim będzie przygotowywany szczegółowy zakres i plan całego projektu. Wskazane jest, aby przedstawione propozycje rozwiązania były weryfikowane przez stronę trzecią, aby zapewnić obiektywność osądu.

Przygotowano wstępny szkic wymagań w ramach procesu **5.1. Zbieranie wymagań**.

Zdefiniowano m.in., że w systemie będą funkcjonowały następujące role użytkowników.

- ♦ Dyrektor:
 - ♦ przegląda raporty analityczne,
 - ♦ przygotowuje strategiczne plany finansowe,
 - ♦ bada realizację strategicznych planów finansowych.
- ♦ Kontroler finansowy:
 - ♦ weryfikuje, czy zagregowane dane przekazane z systemu ERP do BI są prawidłowe, m.in. oczyszcza dane,
 - ♦ przygotowuje szablony planistyczne dla dyrektorów i użytkowników regionalnych,
 - ♦ kontroluje dane planistyczne (informacja zwrotna do dyrektorów i użytkowników regionalnych),
 - ♦ ogląda raporty historyczne oraz aktualne w kontekście wszystkich sklepów i każdego z osobna,
 - ♦ wykonuje analizy typu ad hoc,
 - ♦ wykonuje dedykowane raporty analityczne.
- ♦ Użytkownik regionalny:
 - ♦ wpisuje dane planistyczne w kontekście własnego sklepu,
 - ♦ ogląda raporty historyczne i aktualne w kontekście własnego sklepu.

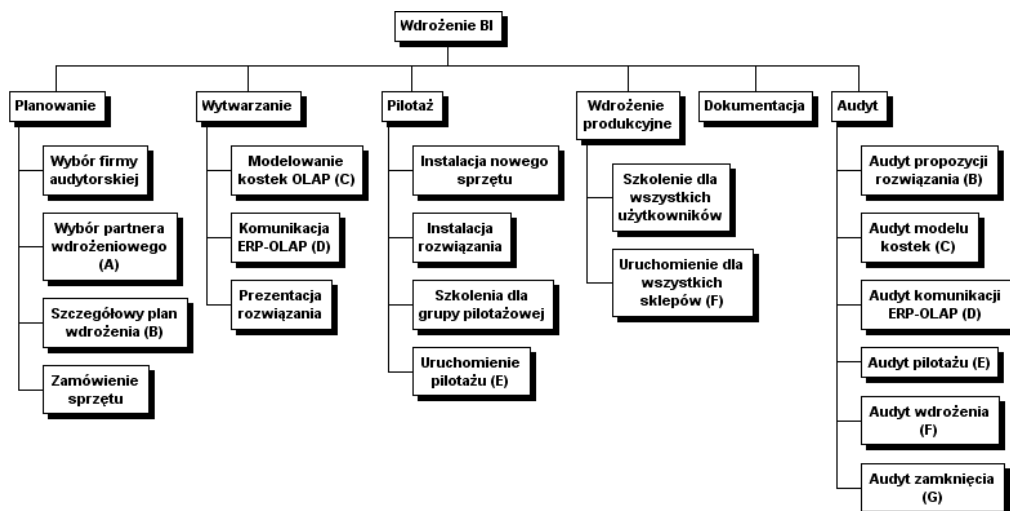
Wstępnie planuje się:

- ♦ 122 użytkowników regionalnych,
- ♦ 3 kontrolerów finansowych,
- ♦ 3 dyrektorów.

Zdefiniowano, iż kluczowymi miarami analitycznymi dla danych finansowych firmy są:

- ♦ Czas — rok, miesiąc, tydzień i dzień sprzedaży,
- ♦ Typ klienta — firma posiada już pewne analizy grupujące klientów indywidualnych i instytucjonalnych,
- ♦ Pracownik — struktura organizacyjna firmy,
- ♦ Produkt — hierarchia produktów wymaga pewnego uporządkowania,
- ♦ Usługi — na finalną wartość na fakturze składa się nie tylko cena samych mebli, ale również takie usługi dodatkowe jak transport lub montaż.

Przygotowano również szkic dokumentu o nazwie **5.3. Struktura pakietów roboczych (WBS)** — rysunek 50.



Rysunek 50. Struktura produktowa (WBS) w przykładzie wdrożenia narzędzia BI

Na podstawie tak przygotowanej struktury produktowej przygotowano plan realizacji. Zdefiniowano kamienie milowe (do każdego z nich dodana zostanie ekspertyza firmy audytorskiej). Oto one.

1. Wybór kluczowego partnera wdrożeniowego (A).
2. Zakończenie fazy planowania (B).
3. Ukończenie modelu kostek OLAP (C).
4. Ustanowienie komunikacji ERP – OLAP (D).
5. Zakończenie pilotażu (E).
6. Zakończenie wdrożenia (F).
7. Zamknięcie projektu (G).

Nie ma jeszcze możliwości oszacowania ani czasu, ani kosztów niezbędnych do osiągnięcia tych punktów. Brak zasobów projektowych jest wyraźnym, wąskim gardłem i dlatego Bartek uruchomił proces **12.1. Planowanie dostaw**.

9 stycznia (środa)

Zapytanie ofertowe o usługi audytorskie

Przygotowano dokument zaopatrzeniowy, który definiował konieczność zakupu usług i sprzętu.

1. Usługi typu „osąd eksperta” (audytorskie):
 - ♦ muszą zostać kupione „od zaraz”,
 - ♦ muszą zapewniać niezależność osądu.

2. Usługi wdrożeniowe:

- ♦ muszą zostać kupione „w trybie pilnym”,
- ♦ powinny bazować na gotowych rozwiązaniach (bogata lista referencyjna),
- ♦ wybór usługodawcy będzie wsparty przez „osąd eksperta”, który zapewni „audyt propozycji rozwiązania”,
- ♦ zostaną zrealizowane w formule „ustalona cena”.

3. Sprzęt, na którym odbędzie się wdrożenie:

- ♦ musi zostać kupiony przed rozpoczęciem testów akceptacyjnych,
- ♦ definicja sprzętu powinna wynikać z wymogów sprzętowych zdefiniowanych przez firmę wdrożeniową.

Zdefiniowano, że wszystkie umowy będą podpisywane przez jednego z członków rady nadzorczej „Bukowina”, ale będą przedkładane przez kierownika projektu. Przy kwotach większych niż 10 tys. złotych Bartek musi dołączyć jednostronicowe uzasadnienie zakupu w dowolnym, przejrzystym formacie.

W kontekście usług audytorskich zdefiniowano następujące kryteria wyboru.

- ♦ Niezależność — firma (osoba) nie może być związana z jednym konkretnym rozwiązaniem, aby zapewnić obiektywność osądu.
- ♦ Kompetencje w obszarze wdrożeń BI — firma (osoba) musi mieć udokumentowane referencje z minimum trzech wdrożeń o podobnej lub większej skali w przeciągu ostatnich pięciu lat.
- ♦ Cena.

Przygotowano krótkie zapytanie ofertowe, które zawierało:

- ♦ ogólny opis projektu wraz z opisem kamieni milowych,
- ♦ zakres usług (pięć dokumentów audytorskich przy osiągnięciu kamieni milowych oraz dodatkowa pula dwudziestu godzin konsultacji),
- ♦ formułę płatności za usługi (płatność za każdy dokument oddzielnie, końcowa płatność za wykorzystane godziny plus opcjonalna premia za powodzenie projektu),
- ♦ prośbę o referencje wraz z danymi kontaktowymi,
- ♦ prośbę o pisemną deklarację niezależności lub informację o wdrożeniu co najmniej trzech różnych typów rozwiązania BI,
- ♦ zapytanie o cenę,
- ♦ datę złożenia oferty — trzy dni robocze od momentu wysłania zapytania.

Zapytanie to zostało wysłane do pięciu ekspertów i trzech firm doradczych.

14 stycznia (poniedziałek)

Zapytanie ofertowe na usługi wdrożeniowe

Decyzja o wyborze partnera wdrożeniowego jest o wiele bardziej krytyczna niż wybór usługi audytorskiej i dlatego na zdefiniowanie tej kwestii poświęcono większą ilość czasu, włącznie z wyznaczeniem punktowych kryteriów wyboru, co ma zapewnić bardziej obiektywną selekcję. Finałem tej pracy są dokumenty dostawcze w kontekście usług wdrożeniowych. Przyjęto następujące wytyczne dotyczące kryteriów:

- ♦ cena wdrożenia 30%,
- ♦ terminowość 15%,
- ♦ zakres funkcjonalności 15%,
- ♦ kompetencje w obszarze BI 15%,
- ♦ cena wsparcia 10%,
- ♦ doświadczenie w integracji BI-ERP 10%,
- ♦ skalowalność rozwiązania 5%.

Kierownik projektu Bartek miał świadomość, że rada nadzorcza mocno podkreślała konieczność terminowej realizacji projektu. Wiedział jednak również, iż w praktyce sukces projektu będzie równie zależny od funkcjonalności i od kompetencji wybranej firmy w obszarze BI. Inną ciekawostką jest kryterium ceny, które posiada wagę tylko i aż 30%. Z jednej strony, większość przetargów tego typu zawiera kryterium ceny z wagą 100 – 70%, a z drugiej strony, w kontekście tego przetargu cena projektu nie jest kryterium najważniejszym. Rada nadzorcza jest zainteresowana zakupem dobrego rozwiązania za dobrą cenę.

Oto dodatkowe warunki konieczne, które musi spełniać usługodawca.

- ♦ Zatrudnienie od 10 do 100 osób, aby zapewnić optymalną elastyczność reakcji na zmienne potrzeby.
- ♦ Funkcjonowanie na rynku od co najmniej pięciu lat.
- ♦ Pełny rejestr osób, które będą brały udział przy realizacji projektu, wraz z wykazem ich doświadczenia i zaświadczeniami o niekaralności, ze względu na czułość danych, do których będą mieli dostęp.
- ♦ Co najmniej jedno udokumentowane wdrożenie rozwiązania BI o podobnej lub większej skali.
- ♦ Zapytanie o ofertę nie jest oficjalnym przetargiem — firma „Bukowina” ma prawo odrzucić ofertę i podać pisemny powód takiej decyzji, ale oferentowi nie przysługuje prawo do zażalenia na taką decyzję.
- ♦ Aplikacja musi zapewniać pewną bazową funkcjonalność.
 - ♦ Kontroler finansowy musi mieć możliwość samodzielnego tworzenia raportów i analiz.

- ♦ Musi istnieć możliwość zagnieżdżania wymiarów w wierszach i kolumnach do trzech poziomów (ang. *nested dimensions*).
- ♦ Musi istnieć możliwość dynamicznej analizy w górę i w dół elementów hierarchii danego wymiaru (ang. *drill up&down*).
- ♦ Musi istnieć możliwość specyfikowania i prezentacji w postaci tekstu kluczowych identyfikatorów wydajności (ang. *Key Performance Indicators* — KPI).
- ♦ Muszą zostać wprowadzone mechanizmy administracyjne (uprawnienia użytkowników).
- ♦ System musi być bezpieczny, tzn. nie będzie można się włamać do niego z zewnątrz ani z wnętrza organizacji.

Przygotowano zapytanie ofertowe, które zawierało:

- ♦ ogólny opis projektu,
- ♦ oczekiwany zakres usług (wdrożenie, szkolenie, wsparcie) — partner wdrożeniowy odpowiada za powodzenie prac wdrożeniowych, na dostarczonym sprzęcie,
- ♦ ogólną informację o docelowej liczbie użytkowników systemu,
- ♦ termin planowanego uruchomienia projektu (1 luty),
- ♦ planowany termin wdrożenia produkcyjnego (koniec drugiego kwartału, ale termin ten może zostać nieco przesunięty),
- ♦ klauzulę, że w przypadku odstępstwa od ustalonej daty wdrożenia umowa będzie zawierać kary w wysokości 0,2% wartości projektu za każdy dzień opóźnienia,
- ♦ klauzulę, że przed uruchomieniem produkcyjnym musi mieć miejsce pilotażowe uruchomienie usługi dla ograniczonej liczby użytkowników, ale ze wszystkimi kluczowymi funkcjonalnościami,
- ♦ formułę współpracy — oferent będzie zarządzał technicznym aspektem prac wdrożeniowych; po formalnym wdrożeniu systemu nastąpi okres dostosowywania go do biznesowych potrzeb firmy, w związku z tym, oprócz ceny za utrzymanie systemu, planowane jest wykonanie pakietów zleconych prac dostosowawczych,
- ♦ formułę płatności za usługi — firma „Bukowina” jest zobowiązana do przekazania informacji na temat kluczowych kosztów przedstawionego rozwiązania, takich jak:
 - ♦ koszt licencji silnika bazodanowego,
 - ♦ koszt licencji rozwiązania BI (dla poszczególnych typów użytkowników),
 - ♦ szacunkowy koszt sprzętu, który będzie musiał zakupić zleceniobiorca.
- ♦ prośbę o referencje wraz z danymi kontaktowymi,
- ♦ zapytanie o metodykę wdrożeniową,

- ♦ zapytanie o cenę wdrożenia, serwis i tabelę stawek za roboczodzień osób o konkretnych specjalizacjach, biorących udział w projekcie,
- ♦ zapytanie o funkcjonalność oferowaną dodatkowo,
- ♦ opis wdrożonego rozwiązania ERP,
- ♦ zapytanie o sposób integracji z istniejącym systemem ERP działającym w firmie „Bukowina”,
- ♦ datę złożenia oferty — 10 dni roboczych od momentu wysłania zapytania ofertowego,
- ♦ informację, że mechanizm oceny oferty bazować będzie na systemie punktowym, gdzie dopuszczalne są wartości ułamkowe do setnych po przecinku.

☒ Cena wdrożenia (maks. 30 punktów)

- ♦ Najlepsza oferta cenowa uzyskuje 30 punktów.
- ♦ Za każdy 1% różnicy od zwycięskiej ceny odejmowane jest 0,15 punktu i w ten sposób wyliczana jest siła ceny każdej z ofert (oferta 2 razy droższa dostaje 15 punktów, a oferta 3 razy droższa 0 punktów).

☒ Termin wdrożenia (maks. 15 punktów)

- ♦ Za wdrożenie produkcyjne do końca drugiego kwartału 15 punktów.
- ♦ Za każdy tydzień później 1 punkt mniej (0 punktów, jeżeli wdrożenie zakończy się w trakcie ostatniego kwartału).

☒ Punkty za funkcjonalność (maks. 15 punktów)

- ♦ Możliwość graficznego generowania raportów i analiz bez konieczności zmian w plikach konfiguracyjnych i tworzenia skryptów. Załączono sześć przykładowych raportów i za każdy z nich można otrzymać maksymalnie 0,5 punktu. Łącznie można uzyskać maksymalnie 3 punkty.
- ♦ Interfejs web z możliwością wykorzystania w trzech najbardziej powszechnych wersjach przeglądarek na rynku polskim (IE 6.0, 7.0 i FireFox). Za pełną zgodność z jedną z tych przeglądarek można uzyskać do 1 punktu. Łącznie można uzyskać maksymalnie 3 punkty.
- ♦ Mechanizmy współpracy grupowej w kontekście konkretnego raportu, gdzie wymagane są co najmniej wielowątkowe komentarze. Najbardziej rozbudowany mechanizm otrzyma 2 punkty.
- ♦ Wbudowane mechanizmy archiwizacji i odtwarzania danych. Najbardziej rozbudowany mechanizm otrzyma 2 punkty.
- ♦ Możliwość dynamicznej analizy danych transakcyjnych, przy przyjęciu za punkt wyjścia danych zagregowanych (ang. *drill trough*). Najbardziej rozbudowany mechanizm otrzyma 2 punkty.

- ♦ Panel administracyjny umożliwiający definiowanie użytkowników z wykorzystaniem kont domenowych za pomocą graficznego interfejsu użytkownika. Najbardziej rozbudowany mechanizm otrzyma 1 punkt.
- ♦ Możliwość graficznej i różnorodnej prezentacji kluczowych identyfikatorów wydajności (ang. *Key Performance Indicators* — KPI). Najbardziej rozbudowany mechanizm otrzyma 1 punkt.
- ♦ Możliwość hurtowego dodawania użytkowników. Najbardziej rozbudowany mechanizm otrzyma 0,5 punktu.
- ♦ Graficzne modelowanie mechanizmów zasilania danymi. Najbardziej rozbudowany mechanizm otrzyma 0,5 punktu.



Uwaga

Zakłada się, iż w danym kryterium maksymalną liczbę punktów może otrzymać więcej niż jeden dostawca.

☒ Kompetencje w obszarze wdrożeń BI (maks. 15 punktów)

- ♦ Za pierwsze udokumentowane wdrożenie wykonane „z należytą starannością” dla 10+ użytkowników — 5 punktów.
- ♦ Za drugie wdrożenie dla 10+ użytkowników — 3 punkty.
- ♦ Za każde następne dla 10+ użytkowników — 1punkt.
- ♦ Za każde wdrożenie dla 100+ użytkowników można uzyskać dodatkowe 2 punkty.



Uwaga

Sumarycznie liczba punktów w tej kategorii nie może przekroczyć 15. Uzyskanie tej wartości daje np. dziewięć jakichkolwiek wdrożeń lub cztery wdrożenia dla 100+ użytkowników.

☒ Cena serwisu powdrożeniowego (maks. 10 punktów)

- ♦ Najlepsza oferta za jeden rok serwisu uzyskuje 5 punktów.
- ♦ Za każdy 1% różnicy od zwycięskiej ceny odejmuje się 0,05 punktu i w ten sposób wyliczana jest siła ceny każdej z ofert (2 razy droższa oferta otrzyma 0 punktów).
- ♦ Najlepsza cena za uśredniony jeden dzień roboczy wsparcia (stosowana będzie do prac dostosowawczych) — 5 punktów.
- ♦ Za każdy 1% różnicy od zwycięskiej ceny odejmuje się 0,05 punktu i w ten sposób wyliczana jest siła ceny każdej z ofert (2 razy droższa oferta otrzyma 0 punktów).

☒ Kompetencje w integracji z istniejącym systemem ERP (maks. 10 punktów)

- ♦ Za pierwsze udokumentowane wdrożenie zaproponowanego systemu BI z systemem ERP tego samego rodzaju, który posiada „Bukowina” — 4 punkty.

- ♦ Za drugie — 3 punkty.
- ♦ Za trzecie — 2 punkty.
- ♦ Za czwarte — 1 punkt.

☒ Punkty za skalowalność rozwiązania (maks. 5 punktów)

- ♦ Wydajność prezentacji raportów

50 tys. komórek w jednym raporcie wyświetlane w 10 sekund — 2 punkty;
10 tys. — 1 punkt; 1 tys. komórek — 0 punktów.

- ♦ Wielkość bazy danych

Udokumentowane wdrożenie z bazą danych o wielkości 10 TB — 2 punkty;
6 TB — 1 punkt; 2 TB — 0 punktów.

- ♦ Czas wykonania operacji zapisu do pustej bazy BI z poziomu GUI

Udokumentowana możliwość wykonania operacji zapisu w 0,1 sekundy
— 1 punkt, 1 sekunda — 0 punktów.

Zapytanie to zostało wysłane do piętnastu firm oferujących pięć różnych rozwiązań technologicznych.

Tego samego dnia projekt zostanie uruchomiony po stronie wykonawcy.

Wykonawca „Mefisto”, 14 stycznia (poniedziałek) Startuje przygotowanie oferty

Kierownik produktu Marcin, otrzymał zapytanie ofertowe z rynku. Często brał udział w takich pracach handlowych, więc uzyskał zgodę na przygotowanie oferty od kierownika handlowego. Kierownik produktu Marcin zainicjował kilka spotkań z kluczowymi pracownikami firmy. Wraz z kierownikiem projektu Michałem, który posiadał duże doświadczenie techniczne, rozpoczęli przygotowanie wizji wdrożenia oraz propozycji finansowej.

Klient „Bukowina”, 15 stycznia (wtorek) Wybór usług audytorskiej

Na osiem wysłanych zapytań dotyczących usług audytorskich spłynęło pięć ofert — trzy od ekspertów i dwie z firm doradczych. Spośród nich kierownik projektu Bartek wybrał dwie szczególnie interesujące oferty ekspertów i dla porównania najlepszą ofertę z firmy doradczej — tabela 7. W każdym z trzech przypadków Bartek wykonał długą rozmowę telefoniczną w celu weryfikacji przedstawionej oferty. Poza tym, w przypadku obu ekspertów i firmy doradczej zweryfikował jakość świadczonych przez z nich usług, wykonując telefon do ich klientów i kontrahentów.

W trakcie dyskusji z dyrektorami wynikło, że jeśli wybrane zostanie rozwiązanie bazujące na doświadczeniach eksperta B, to właśnie on będzie najlepszym wyborem. Jeżeli jednak decyzja musi zostać podjęta dzisiaj, ekspert A jest najlepszy. W kontekście wybranego rozwiązania firma wdrożeniowa ma posiadać kluczowe doświadczenia, a ekspert

Tabela 7. Tabela wyboru usług audytorskich w przykładowym wdrożeniu narzędzia klasy BI

	Ekspert A	Ekspert B	Firma X	Uwagi
Liczba udokumentowanych wdrożeń	5	8	6	
Niezależność	Wiele różnych rozwiązań	Głównie jedno z rozwiązań — deklaracja niezależności	Wiele różnych	
Zakres doświadczenia	Różnorodne, ciekawe projekty	Głównie projekty BI	Bardzo wiele różnych projektów	
Cena sumaryczna	20300	25650	52100	
Cena za audyt trzech propozycji rozwiązania	3500	2900	5600	
Cena za audyt modelu kostek (około 6 kostek)	1800	1900	5600	B — najszerszy zakres
Cena za audyt komunikacji ERP – BI	2200	1900	4200	X — najciekawsze podejście;
Cena za audyt pilotażu	1300	1900	2800	B — największe doświadczenie
Cena za audyt raportu EVM	900	750	1900	Wstępnie planuje się trzy takie audyty
Cena za audyt systemu produkcyjnego	1800	2900	4200	X — system ankiet
Cena za dodatkowy pakiet 20 roboczodni	14000	23800	48000	

powinien zapewniać „świeżość spojrzenia” i szerokie spektrum doświadczeń. Wspólnie podjęto decyzję, że kluczowa jest obiektywność eksperta, który nie powinien preferować żadnej z platform, i dlatego zdecydowano o wyborze eksperta A.

Ekspert A został powiadomiony o decyzji i rozpoczęto przygotowanie umowy — proces wymiany uwag zaczął toczyć się w tle.

Klient „Bukowina”, 16 stycznia (środa) **Plan zarządzania projektem**

W związku z procesem **4.2. Opracowanie planu zarządzania projektem** kierownik projektu Bartek uporządkował rejestr dokumentów projektowych i przygotował własny indeks stopnia realizacji kluczowych procesów PMBoK jako arkusz Excela (tabela 8). Ze względu na charakter projektu świadomie pominięto procesy z 9. obszaru wiedzy (zasoby ludzkie).

Tabela 8. Tabela stopnia realizacji procesów PMBoK (1) w przykładowym wdrożeniu narzędzia klasy BI

Dokument		Katalog	Uwagi	Wykonanie
4.1. Dokument otwarcia	I	4. Zarządzanie\ Dokument otwarcia		100%
10.1. Rejestr interesariuszy	I	10. Komunikacja		100%
4.2. Plan zarządzania projektem	P	4. Zarządzanie	Plan w katalogu głównym	100%
5.2. Zakres projektu	P	5. Zakres projektu	Częścią tego dokumentu są zebrane wymagania (5.1)	20%
5.3. Struktura pakietów roboczych (WBS)	P	5. Zakres projektu		80%
6.5. Harmonogram	P	6. Harmonogram	W ramach harmonogramu realizowane są procesy 6.1 – 6.5	0%
7.2. Budżet projektu	P	7. Budżet	W ramach harmonogramu realizowane są procesy 7.1 – 7.2	0%
8.1. Plan jakości	P	8. Jakość		0%
10.2. Plan komunikacji	P	10. Komunikacja		0%
11.1. Plan zarządzania ryzykiem	P	11. Ryzyko		0%
11.5. Rejestr ryzyk	P	11. Ryzyko	W ramach harmonogramu realizowane są procesy 11.2 – 11.5	0%
12.1. Plan dostaw	P	12. Dostawy		100%
4.3. Kierowanie i zarządzanie realizacją projektu	R	4. Zarządzanie\ Rejestr zdarzeń		0%
		6. Harmonogram		
8.2. Przeglądy jakościowe	R	8. Jakość\Przeglądy wewnętrzne	Bieżąca kontrola jakości	0%
10.3. Okresowe raporty	R	10. Komunikacja\ Raporty		0%
12.2. Oferty od sprzedawców i wybór	R	12. Zaopatrzenie\ Oferty		15%
4.5. Rejestr zmian	K	4. Zarządzanie\ Zmiany		0%

Tabela 8. Tabela stopnia realizacji procesów PMBoK (1) w przykładowym wdrożeniu narzędzia klasy BI — ciąg dalszy

Dokument		Katalog	Uwagi	Wykonanie
5.5. Weryfikacja i kontrola zakresu	K	8. Jakość\Przeglądy	Kierownik będzie dokonywał przeglądów wraz z audytorem skoncentrowanym na jakości	0%
6.6. Kontrola harmonogramu	K	6. Harmonogram	Aktualizacja harmonogramu	0%
7.3. Kontrola kosztów	K	7. Budżet\ RaportyEVM		0%
8.3. Kontrola jakości	K	8. Jakość\Przeglądy audytowe	Kontrola zapewnienia jakości w projekcie	0%
10.5. Raportowanie wydajności	K	7. Budżet\ RaportyEVM	Z audytami eksperta zewnętrznego	0%
11.6. Monitorowanie ryzyk i nadzór nad nimi	K	11. Ryzyko		0%
12.3. Administrowanie kontraktami	K	12. Zaopatrzenie\ Rejestr zdarzeń		0%
4.6. Zamknięcie projektu	Z	4. Zarządzanie\ Zamknięcie		0%
12.4. Zamknięcie kontraktów	Z	12. Zaopatrzenie\ Zamknięcie		0%
Wykonanie kamieni milowych 0/6				

Do tego pliku dodano dodatkowy arkusz Excela, który opisywał projekt i zawierał:

- ♦ nazwę projektu,
- ♦ dane kierownika projektu,
- ♦ dane kontaktowe,
- ♦ datę rozpoczęcia i zakończenia projektu,
- ♦ uwagi ogólne do projektu,
- ♦ rejestr zmian do planu wraz z uwagami,
- ♦ rejestr dokumentów powiązanych.

Kierownik projektu Bartek będzie okresowo przeglądał ten dokument i monitorował stopień realizacji procesów przy każdym odebraniu produktu od partnera wdrożeniowego. Wszelkie zmiany w dokumencie będą ewidencjonowane.

Klient „Bukowina”, 21 stycznia (poniedziałek) Przygotowanie planów zarządczych

Kolejne trzy dni poświęcono na przygotowanie następujących dokumentów.

♦ Plan jakości

Wykorzystano tutaj proces **8.1. Planowanie jakości**

- ♦ Kierownik projektu Bartek jest odpowiedzialny za bieżącą kontrolę jakości wdrażanego rozwiązania, ale może to zadanie delegować na zewnętrznego audytora.
- ♦ Zostało już przygotowanych sześć przykładowych raportów — planowane jest przygotowanie dodatkowych list kontrolnych (ang. *checklist*).
- ♦ W fazie pilotażu kontrolerzy finansowi wykonają szereg zaplanowanych testów oraz testy ad hoc.
- ♦ Kierownik projektu Bartek może w dowolnym momencie, z jednodniowym wyprzedzeniem, poprosić o możliwość weryfikacji stanu prac wdrożeniowych i wykonać zaplanowane listy kontrolne u wykonawcy.
- ♦ Ekspert zewnętrzny może zaproponować dodatkowe mechanizmy utrzymania jakości.
- ♦ Przygotowano również jako arkusz Excela **rejestr błędów**, w którym będą notowane wszelkie zarejestrowane błędy wraz z takimi parametrami jak waga błędu.
- ♦ Analizowane będą następujące parametry względem czasu:
 - ♦ ilość otwartych błędów,
 - ♦ ilość błędów otwartych w stosunku do ilości błędów zamkniętych,
 - ♦ ilość błędów o konkretnej wadze w całym wolumenie błędów,
 - ♦ min., śr. i maks. czas reakcji względem min., śr. i maks. czasu naprawy.
- ♦ Po fazie pilotażu zostanie grupowo podjęta decyzja o uruchomieniu systemu produkcyjnego bądź konieczności naniesienia poprawek do pilotażu. Decyzja ta zostanie podjęta przez gremium składające się z kierownika projektu, przedstawiciela firmy wdrożeniowej i trzech głównych użytkowników systemu.
- ♦ Dokumentacja zostanie sprawdzona przez kontrolerów finansowych pod kątem kompletności i przejrzystości opisu.
- ♦ Efektywność szkolenia zostanie przetestowana w formie prostego egzaminu.

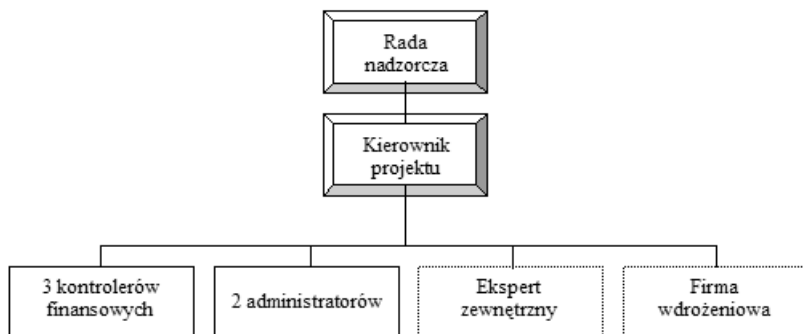
♦ Plan komunikacji

Wykorzystano tutaj proces **10.2. Planowanie komunikacji**

- ♦ Zostało jasno zdefiniowane, że rada nadzorcza jest zainteresowana monitorowaniem projektu z wykorzystaniem punktów kontrolnych

(ang. *milestones*). Przed każdym takim spotkaniem musi zostać przygotowany specjalny raport kierownika projektu oraz audyt zewnętrznego eksperta.

- ♦ Kierownik projektu Bartek musi przynajmniej raz w miesiącu kopiować własną dokumentację projektową na centralny serwer FTP klienta.
- ♦ Sugeruje się, aby Bartek na początku każdego miesiąca wykonał krótki raport ze stanu projektu, gdzie zawarte będą co najmniej informacje o pracach wykonanych w zeszłym miesiącu i plany na następny miesiąc.
- ♦ Rada nadzorcza jest zainteresowana głównie stopniem wykorzystania budżetu oraz postępami względem harmonogramu projektu i dlatego technika Zarządzania Wartością Wypracowaną (ang. *Earned Value Management* — EVM) będzie dla niej najbardziej interesująca. Od wyników analizy uzależniona jest premia dla kierownika projektu i podwykonawców.
- ♦ Rada nadzorcza w ograniczonym stopniu jest zainteresowana aspektami technicznymi przedsięwzięcia.
- ♦ Członkowie rady mają prawo do nieformalnej komunikacji z kontrolerami finansowymi i poza projektem rozliczają ich m.in. ze stopnia implementacji procesów biznesowych w systemie BI.
- ♦ Na żądanie rady nadzorczej kierownik projektu Bartek zobowiązany jest do przygotowania raportu ze stanu prac w projekcie w ciągu trzech dni.
- ♦ Struktura organizacyjna projektu jest stosunkowo prosta.



Rysunek 51. Struktura zespołu wdrożeniowego w przykładowym wdrożeniu narzędzia klasy BI

Pod koniec dnia, po serii kilku poprawek z obu stron udało się ustalić brzmienie umowy z ekspertem A i podpisano z nim umowę o współpracy.

Wykonawca „Mefisto”, 21 stycznia (poniedziałek) Zamknięcie oferty

Po całym tygodniu wyteżonej pracy kierownikowi produktu Marcinowi udało się wreszcie zamknąć kompletną wizję wdrożenia. Konieczny okazał się znaczący wkład merytoryczny kierownika projektu Michała i jednego z kluczowych wdrożeniowców, który przekazał szereg cennych uwag na temat potencjalnych oczekiwań klientów oraz bardzo ostrożnego

podejścia do rozszerzania funkcjonalności systemu, ze względu na wysokie koszty utrzymania optymalnej jakości. Rozpatrzono kilka wariantów oferty, poczynawszy od standardowego wdrożenia „prosto z pudełka”, a skończywszy na zaawansowanym oprogramowaniu istniejącego rozwiązania.

Kierownik produktu Marcin wykonał wyliczenia kosztu jednego punktu bazowego. Przy założeniu ceny wdrożenia na 2 mln zł wyliczył, że będzie on wart około 133,3 tys. W takiej sytuacji każdy tydzień opóźnienia to właśnie 133,3 tys., a pewne kluczowe, niepowtarzalne funkcjonalności, takie jak praca grupowa, archiwizacja danych czy analiza danych zagregowanych i transakcyjnych (ang. *drill through*), to 266,6 tys. Kierownik produktu Marcin wykonał kilka analiz typu co – jeśli. Finalnie postanowił oprzeć strategię sprzedaży na maksymalizacji zakresu funkcjonalności i opóźnieniu czasu wdrożenia. Konieczne było również podwyższenie prognoz kosztu utrzymania rozwiązania poprzez ustalenie wysokich poziomów kosztu rocznego serwisu i kosztu jednego dnia roboczego.

Kierownik produktu Marcin miał świadomość dużego ryzyka tego podejścia zarówno od strony technologicznej, jak i utraty pewnych punktów. Czterotygodniowe opóźnienie wdrożenia to przecież 4 punkty straty. Niestety, tylko w ten sposób można wykonać unikalne funkcjonalności, które powinny dać dodatkowe 6 punktów przewagi nad konkurencją. Dzięki stałemu monitorowaniu rynku kierownik produktu Marcin wiedział, że w danej chwili konkurenci nie posiadali zaoferowanych przez niego funkcjonalności. Decyzja ta została podyktowana również mapą drogową produktu i planami rychłego wprowadzenia nowej wersji rozwiązania, która jest spójna z przygotowaną ofertą. Założono solidne marginesy czasowe na testy wewnętrzne i możliwe opóźnienie premiery nowej wersji.

W notatce dołączonej do oferty dla celów wewnętrznych kierownik produktu Marcin odnotował, iż oferta:

- ♦ może pozytywnie wyróżnić się wśród ofert konkurencji,
- ♦ kładzie nacisk na najwyżej punktowane elementy w zapytaniu,
- ♦ przedstawia zaawansowane, ambitne wdrożenie,
- ♦ „Bukowina” to bardzo dobry klient referencyjny dla nowej wersji produktu.

Kierownik produktu Marcin przygotował kompletną ofertę, w której zawarł również ramowy plan wdrożenia, ceny, referencje dotychczasowych klientów, testy banchmar-kingowe, testy wydajnościowe, propozycję konfiguracji sprzętowej itp. Oferta została zaakceptowana przez dyrektora handlowego i przekazana klientowi.

Zakończyła się Faza Wizji projektu.

Klient „Bukowina”, 22 stycznia (wtorek) **Wybór usługodawcy wdrożenia**

Na piętnaście wysłanych zapytań otrzymano oferty z siedmiu firm na czterech platformach technologicznych. W trakcie wstępnej analizy odrzucono jedną z propozycji (G) ze względu na zdecydowanie „zbyt niską” cenę oraz małe doświadczenie wdrożeniowe

oferenta. Firma G posiadała tylko dwa poświadczone wdrożenia i nie miała doświadczenia w integracji z systemem ERP działającym w firmie „Bukowina”.

Przyjęto założenie, że kierownik projektu Bartek będzie pracował niezależnie od eksperta zewnętrznego, aby zapewnić obiektywność wyboru.

Pozostałe sześć ofert przedstawiało bardzo różne sposoby podejścia do problemu (tabela 9).

Tabela 9. Tablica porównawcza oferentów w przykładzie wdrożenia rozwiązania klasy BI

OFERENT	A	B	C	D	E	F	G
Cena wdrożenia (w tys.)	4298	3241,6	4721,5	2026,4	2131,2	2119,5	1170,8
Różnica ceny wdrożenia (%)	112	60	133	0	5	5	
Termin	30-cze	11-lip	30-cze	25-lip	30-cze	30-cze	25-lip
1 rok wsparcia (w tys.)	110	138	116	178	109	108	
Różnica ceny roku wsparcia (%)	2	28	7	65	1	0	
1 godzina wsparcia (w tys.)	2	1,8	1,7	1,9	1,9	1,8	2,1
Różnica ceny godziny wsparcia (%)	18	6	0	12	12	6	
Punktacja	74,71	73,81	71,18	83,43	78,57	78,02	
Cena wdrożenia (30)	13,18	21,00	10,05	30,00	29,22	29,31	
Terminowość (15)	15	13	15	11	15	15	
Funkcjonalność (15)	10,5	14,5	8,5	15	10	12	
6 przykładowych raportów (3)	3	3	3	3	2	2	
Interfejs web (3)	3	3	3	3	3	3	
Praca grupowa (2)	0	2	0	2	0	0	
Archiwizacja danych (2)	2	2	0	2	2	2	
Analiza danych zagregowanych i transakcyjnych (ang. <i>drill through</i>) (2)	0	2	0	2	0	2	
Panel administracyjny (1)	1	1	1	1	1	1	
KPI (1)	1	1	1	1	1	1	
Hurtowy import użytkowników (1)	0	0	0	0,5	0,5	0,5	
GUI zasilania danych (0,5)	0,5	0,5	0,5	0,5	0,5	0,5	
Kompetencje BI (15)	15	8	15	11	10	7	
Cena wsparcia (10)	9,03	8,31	9,63	6,18	9,35	9,71	
Kompetencje ERP – BI (10)	7	4	9	7	0	0	
Skalowalność rozwiązania (5)	5	5	4	3,25	5	5	
Wyświetlanie komórek (2)	2	2	1,5	1,75	2	2	
Wielkość bazy (2)	2	2	2	1	2	2	
Szybkość zapisu (1)	1	1	0,5	0,5	1	1	

Kierownik projektu Bartek zweryfikował powyższą punktację. Zwycięska oferta D zapowiadała najszerzy zakres funkcjonalności (często jako jedyna), ale najpóźniejszy termin wdrożenia i największe koszty utrzymania. Co więcej, wydajność rozwiązania D nie była najciekawsza. Propozycje druga i trzecia (E i F) oferowały z kolei wdrożenie na czas, stosunkowo dobrą wydajność i najniższe koszty utrzymania. Niestety, tylko cztery z sześciu przykładowych formularzy mogą zostać zaimplementowane. Nie są to kwestie kosmetyczne, ale istotny sygnał dotyczący ograniczeń funkcjonalnych rozwiązań E i F bazujących na tej samej platformie technologicznej.

Ostatecznie kierownik projektu Bartek zwrócił uwagę na kryterium doświadczenia. Firma D wykazywała się większym doświadczeniem w obszarze BI od E i F. Poza tym, E i F nie miały żadnego doświadczenia w kontekście integracji systemu BI z systemem ERP!

Zdefiniowano następujące ryzyka.

- ♦ Późniejsze wdrożenie rozwiązania D — możliwe jest „ręczne wykonanie budżetu” na trzeci kwartał i „przepisanie” go do systemu po wdrożeniu. Takie podejście umożliwi pełne przetestowanie systemu produkcyjnego w trzecim kwartale i użycie go w czwartym.
- ♦ Wysoki koszt wsparcia może wskazywać na konieczność dużej ingerencji wdrożeniowców w system po okresie wdrożenia (duża liczba błędów) — można zauważyć pewną tendencję, że firmy posiadające większe doświadczenie oferują wyższy koszt utrzymania rozwiązania. Postanowiono poprosić oferenta o złożenie dodatkowych wyjaśnień.
- ♦ Skalowalność rozwiązania — proponowana skalowalność jest akceptowalna. W obrębie pierwszych trzech lat nie przewiduje się, aby baza analityczna urosła do rozmiarów większych niż 2 TB. System archiwizacji zapewni również utrzymanie optymalnej wielkości bazy danych.

Postanowiono, że wszystkim oferentom, włącznie z G, zostanie zadane zapytanie o uszczegółowienie wyliczeń związanych z rocznym kosztem wsparcia — czas odpowiedzi ustalono na jeden dzień roboczy.

Klient „Bukowina”, 23 stycznia (środa) **Plan zarządzania ryzykiem**

Kierownik projektu Bartek postanowił zrealizować **11.1. Planowanie zarządzania ryzykiem**.

- ♦ Przygotowano **rejestr ryzyk** (arkusz Excel), w którym zostaną odnotowane wszystkie ryzyka projektowe.
- ♦ W obrębie tego rejestru zostały zdefiniowane takie parametry ryzyka jak waga (wpływ na projekt), obszar wystąpienia, prawdopodobieństwo wystąpienia.
- ♦ Na podstawie powyższych parametrów stworzono formułę wyliczającą siłę ryzyka.
- ♦ Rejestr ten będzie uzupełniany i monitorowany cyklicznie wraz z przedstawicielem firmy wdrożeniowej — długość cyklu zostanie wyznaczona w szczegółowym planie wdrożenia.

- ♦ Po uruchomieniu pilota do zespołu dołączy trzech kontrolerów finansowych.
- ♦ Głos decydujący dotyczący sposobu zarządzania ryzykiem ma kierownik projektu Bartek.
- ♦ Istnieje możliwość konsultacji sposobu zarządzania ryzykiem z ekspertem zewnętrznym.

Wykonawca „Mefisto”, 23 stycznia (środa) Zapytanie dodatkowe

Kierownik produktu Marcin otrzymał zapytanie dodatkowe na temat sposobu wyliczenia kosztów wsparcia i odpowiedział na nie w sposób następujący:

Firma „Mefisto” wylicza koszt rocznego serwisu jako procent wartości całego projektu. W kontekście państwa zapytania zaproponowaliśmy bardzo preferencyjne warunki i pierwszy rok wsparcia stanowi zaledwie 8,8% wartości całego projektu. Tego typu stawki proponujemy zwyczajowo jedynie stałym, strategicznym partnerom.

Proszę zwrócić uwagę, że nasze usługi serwisowe zawierają rozszerzony pakiet czynności. Oferujemy nie tylko standardową naprawę błędów, ale również usługi związane z pielęgnacją konfiguracji wdrożonego rozwiązania. W ramach tych usług planujemy dokonywanie kwartalnych przeglądów konfiguracji oraz przeprowadzenie testów wydajnościowych. Jeżeli taki kształt usługi serwisowej nie odpowiada Państwa oczekiwaniom, skłonni jesteśmy ją negocjować.

Misją firmy „Mefisto” jest dostarczanie nowoczesnych rozwiązań, które zapewnią naszym klientom przewagę konkurencyjną. W związku z tym przedstawiliśmy ofertę, która jest rozwiązaniem „pod klucz” i minimalizuje zaangażowanie Państwa zasobów w trakcie wdrożenia i eksploatacji. Bardzo proszę o informacje, czy jesteście Państwo zainteresowani inną formułą usługi serwisowej, czy też powyższe wyjaśnienia są satysfakcjonujące.

Klient „Bukowina”, 24 stycznia (czwartek) Odpowiedzi oferentów

Wszyscy oferenci, z wyjątkiem oferenta E, złożyli stosowne wyjaśnienia. Stopień szczególności i sposób wyliczenia kosztu rocznej gwarancji był bardzo różny, począwszy od bardzo ogólnikowych wyjaśnień, a skończywszy na precyzyjnych wzorach wyliczenia wartości.

Kierownik projektu Bartek uznał, że posiada już wszystkie materiały niezbędne do podjęcia decyzji. Przekazał do eksperta zewnętrznego:

- ♦ zapytanie ofertowe wraz z kryteriami i systemem punktacji ofert,
- ♦ oferty wraz wyliczeniami punktacji,
- ♦ odpowiedzi na pytanie dodatkowe.

Kierownik projektu Bartek nie przekazał żadnych swoich sugestii, przemyśleń czy analiz, aby zapewnić obiektywność osądu eksperta.

W trakcie rozmowy telefonicznej z dyrektorami ustalono, że w ciągu dwóch dni roboczych odbędzie się spotkanie. W jego trakcie zostanie podjęta jednoznaczna i ostateczna decyzja dotycząca wyboru firmy wdrożeniowej.

Klient „Bukowina”, 29 stycznia (wtorek) Decyzja o wyborze usługodawcy wdrożenia

Na początku spotkania kierownik projektu Bartek poinformował zarząd, że mając na celu obiektywność podjętej decyzji, pracował w pełni niezależnie od eksperta zewnętrznego — przekazał mu na trzy dni wszystkie dostępne materiały i dopiero dziś pozna jego stanowisko.

Następnie Bartek przedstawił krótką prezentację przedstawiającą:

- ♦ Wypracowaną wizję projektu,
- ♦ Sposób zebrania ofert,
- ♦ Zagregowane informacje na temat zebranych ofert,
- ♦ Powód wyboru oferenta D,
- ♦ Analizę zagrożeń i korzyści w przypadku oferenta D.

Prezes firmy wyraził uznanie dla pracy kierownika projektu i jego sposobu podejścia do tematu, po czym przekazał głos ekspertowi zewnętrznemu.

Ekspert wręczył każdemu z członków zebrania pisemną kopię swojego kilkustronicowego raportu oraz ustnie zaprezentował pewne kluczowe informacje.

- ♦ Brak spisanej informacji, że system punktacji jest jedynie elementem pomocniczym przy podjęciu finalnej decyzji.
- ♦ Drobne uwagi w stosunku do systemu punktacji:
 - ♦ zbyt silny nacisk na cenę wdrożenia, a zbyt mały na terminowość,
 - ♦ bardzo dobre, przykładowe sześć raportów,
 - ♦ niejasne wymogi dotyczące archiwizacji danych,
 - ♦ możliwość hurtowego dodawania użytkowników nie powinna być w ogóle wzięta pod uwagę, ponieważ ręczne dodanie wszystkich użytkowników jest pomijalnym kosztem,
 - ♦ brak oceny sposobu integracji z systemem ERP — 0,5 punktu za graficzną metodę transmisji danych to zdecydowanie zbyt mało,
 - ♦ brak opisu szczegółowych warunków mierzenia „wydajności prezentacji raportów” i „zapisu danych”, np. brak opisu, że pomiar musi być wykonywany na stacji roboczej innej niż serwer bazy danych,

- ♦ generalna pozytywna ocena sposobu wyboru partnera wdrożeniowego,
- ♦ duże kompetencje merytoryczne kierownika projektu oraz jego dobra znajomość potrzeb klienta,
- ♦ ciekawe spektrum ofert podwykonawców.
- ♦ Rankingowanie ofert podwykonawców niezależnie od przyjętego systemu punktacji.
- ♦ E — najbezpieczniejsze rozwiązanie.

Rozwiązanie na czas, niskie koszty utrzymania, brak doświadczenia w obszarze integracji z istniejącym systemem ERP nie powinien być dużym problemem, ponieważ proponowane jest rozwiązanie dość standardowe.
- ♦ D — ryzykowne, ale efektywne rozwiązanie.

Od strony funkcjonalnej zdecydowanie najciekawsze rozwiązanie, ale obarczone pewnym ryzykiem „unikalności”. Pomimo dużego doświadczenia firmy w obszarze BI oraz integracji z istniejącym systemem ERP, pojawia się ryzyko niedotrzymania terminu. Dlatego też sugeruje się wybór opcji E.
- ♦ F — wariant rozwiązania E.

Oferta F jest bardzo podobna do oferty E, ale firma posiada mniejsze kompetencje w obszarze BI.

Po tej wypowiedzi pomiędzy członkami spotkania wywiązała się burzliwa dyskusja na temat wyboru pomiędzy ofertami D a E. Finalnie zdecydowano, że to kierownik projektu Bartek jest odpowiedzialny za całość przeprowadzonych prac i w związku z tym przychylnie się do jego decyzji. Ekspert zewnętrzny podkreślił, że rozumie przyczyny takiej decyzji i podjęcie dodatkowego ryzyka związanego z rozwiązaniem D jest uzasadnione.

Rozpoczęto prace nad przygotowaniem umowy z firmą D.

Wykonawca „Mefisto”, 23 stycznia (środa) WYGRALIŚMY!!!

Otrzymano informację o wygranym przetargu, która lotem błyskawicy obiegała całe biuro.

Kierownik handlowy zabronił jednak oficjalnego angażowania jakichkolwiek pracowników do momentu podpisania umowy.

Już następnego dnia kierownik produktu Marcin wysłał pierwszy szkic umowy, który podlegał drobnym poprawkom nanoszonym przez obie strony w czasie kilku następnych dni.

Klient „Bukowina”, 4 lutego (poniedziałek) Szacowanie kosztów

Na bazie kosztów firmy D oraz zaproponowanej specyfikacji sprzętowej uruchomiono proces **7.1. Szacowanie kosztów**. Przyjęto następujące założenia budżetowe:

Wujek Dobra Rada — odcinek 8. „A co mówi o tym umowa?”



USTALONY Z KLIENTEM KOSZT I ZAKRES PROJEKTU TO POŁOWA SUKCESU,
DRUGA POŁOWA TO PODPISANIE DOBREJ UMOWY ;).

- ♦ Budżet netto (bez podatku VAT),
- ♦ Rezerwa budżetowa na poziomie około 20% wartości projektu,
- ♦ Premia dla eksperta zewnętrznego na poziomie maks. 22%,
- ♦ Utrzymanie środowiska backupowego w modelu active-passive, co daje możliwość zakupu jednego zbioru licencji,
- ♦ Aktywne wykorzystanie firmy wdrożeniowej w trzecim okresie rozliczeniowym w celu optymalizacji instalacji od strony technicznej i biznesowej,
- ♦ Zakup sprzętu na bazie specyfikacji przygotowanej przez firmę D.

Finalnie przygotowano pierwszą wersję budżetu projektu (tabela 10).

Tego samego dnia udało się ustalić wspólne brzmienie umowy z firmą wdrożeniową. Kierownik projektu Bartek wykonał telefon do jednego z dyrektorów i poinformował o tym fakcie. Ustalił z nim, że umowa zostanie podpisana w środę.

Klient „Bukowina”, 5 lutego (wtorek) Przegląd stanu projektu

W ramach procesu 10.3. **Dystrybucja informacji** przygotowano miesięczny raport ze stanu realizacji projektu (tabela 11).

Tabela 10. Plan budżetu projektu wdrożenia rozwiązania klasy BI

	Kierownik projektu	Premia kierownika	Ekspertyzy eksperta	Roboczodni eksperta	Premie eksperta	Licencje typu DB/BI	Praca wdrożeniowa	Miesięczne wsparcie	Godziny wsparcia	Zakup sprzętu	Licencje typu OS/Office	Rezerwa budżetowa	SUMA
Styczeń	10		3,5										13,5
Luty	10			2,8								35	47,8
Marzec	10			1,4						110	15	80	318,4
Kwiecień	10		2,7	2,1		102						35	49,8
Maj	10		3,1	2,1								37	52,2
Czerwiec	10		2,2	2,1	2							37	53,3
Lipiec	10		1,8	1,4	4	488	1436					180	2122
Sierpień	10			0				14,82	9,5			5	39,32
Wrzesień	10			0				14,82	3,8			4	32,62
Październik	10			1,4				14,82	7,6			5	38,82
Listopad	10			0				14,82	3,8			4	32,62
Grudzień	10	30		0,7				14,82	5,7			10	71,22
SUMA	120	30	13,3	14	6	590	1436	74,1	30,4	110	15	432	2871

Wybór firmy wdrożeniowej
Przygotowanie planu wdrożenia
Zakup sprzętu i licencji
Zamodelowane kostki
Komunikacja ERP – BI
Pilotaż
Wdrożenie
4. okres rozliczeniowy
Zamknięcie projektu

Tabela 11. Tabela stopnia realizacji procesów PMBoK (2) w przykładowym wdrożeniu narzędzia klasy BI

Dokument		Katalog	Uwagi	Wykonanie
4.1. Dokument otwarcia	I	4. Zarządzanie\ Dokument otwarcia		100%
10.1. Rejestr interesariuszy	I	10. Komunikacja		100%
4.2. Plan zarządzania projektem	P	4. Zarządzanie	Plan w katalogu głównym	0%
5.2. Zakres projektu	P	5. Zakres projektu	Częścią tego dokumentu są zebrane wymagania (5.1)	20%
5.3. Struktura pakietów roboczych (WBS)	P	5. Zakres projektu		80%
6.5. Harmonogram	P	6. Harmonogram	W ramach harmonogramu realizowane są procesy 6.1 – 6.5	0%
7.2. Budżet projektu	P	7. Budżet	W ramach harmonogramu realizowane są procesy 7.1 – 7.2	100%
8.1. Plan jakości	P	8. Jakość		100%
10.2. Plan komunikacji	P	10. Komunikacja		100%
11.1. Plan zarządzania ryzykiem	P	11. Ryzyko		100%
11.5. Rejestr ryzyk	P	11. Ryzyko	W ramach harmonogramu realizowane są procesy 11.2 – 11.5	0%
12.1. Plan dostaw	P	12. Dostawy		100%
4.3. Kierowanie i zarządzanie realizacją projektu	R	4. Zarządzanie\ Rejestr zdarzeń		0%
		6. Harmonogram		
8.2. Przeglądy jakościowe	R	8. Jakość\Przeglądy wewnętrzne	Bieżąca kontrola jakości	0%
10.3. Okresowe raporty	R	10. Komunikacja\ Raporty		0%
12.2. Oferty od sprzedawców i wybór	R	12. Zaopatrzenie\ Oferty		80%
4.5. Rejestr zmian	K	4. Zarządzanie\ Zmiany		0%
5.5. Weryfikacja i kontrola zakresu	K	8. Jakość\Przeglądy	Kierownik będzie dokonywał przeglądów wraz z audytorem skoncentrowanym na jakości	0%

Tabela 11. Tabela stopnia realizacji procesów PMBoK (2) w przykładowym wdrożeniu narzędzia klasy BI — ciąg dalszy

Dokument		Katalog	Uwagi	Wykonanie
6.6. Kontrola harmonogramu	K	6. Harmonogram	Aktualizacja harmonogramu	0%
7.3. Kontrola kosztów	K	7. Budżet\ RaportyEVM		0%
8.3. Kontrola jakości	K	8. Jakość\Przeglądy audytowe	Kontrola zapewnienia jakości w projekcie	0%
10.5. Raportowanie wydajności	K	7. Budżet\ RaportyEVM	Z audytami eksperta zewnętrznego	0%
11.6. Monitorowanie ryzyk i nadzór nad nimi	K	11. Ryzyko		0%
12.3. Administrowanie kontraktami	K	12. Zaopatrzenie\ Rejestr zdarzeń		0%
4.6. Zamknięcie projektu	Z	4. Zarządzanie\ Zamknięcie		0%
12.4. Zamknięcie kontraktów	Z	12. Zaopatrzenie\ Zamknięcie		0%
Wykonanie kamieni milowych 1/6				

Klient „Bukowina”, 6 lutego (środa) Prezentacja stanu projektu

W ramach procesu **10.4. Zarządzanie oczekiwaniami interesariuszy** odbyło się spotkanie prezentujące stan projektu. Przedstawiono miesięczny raport ze stanu projektu, w którym odnotowano kilkudniową zwłokę, ale udowodniono, że nie zagraża ona celom projektowym w obecnym kształcie.

Podpisano umowę z firmą wdrożeniową, która natychmiast przystąpiła do prac projektowych z kierownikiem projektu Bartkiem.

Wykonawca „Mefisto”, 6 lutego (środa) Podpisano umowę

Otrzymano podpisaną umowę i kierownik produktu Marcin oficjalnie przekazał pałeczkę nowemu kierownikowi projektu Michałowi, który niezwłocznie poczynił przygotowania do nowego wdrożenia — jeszcze tego samego dnia wysłał klientowi zaproszenie na spotkanie.

Rozpoczęto fazę planowania (MSF).

Klient „Bukowina”, 11 lutego (poniedziałek) **Złożono zamówienie na sprzęt**

W trakcie prac planistycznych zdefiniowano, że wąskim gardłem będzie czas oczekiwania na platformę sprzętową. W związku z tym, wspólnie z firmą wdrożeniową postanowiono niezwłocznie wyspecyfikować i zamówić tenże sprzęt.

Okazało się, że dział IT ma jednego, preferowanego dostawcę, u którego już wcześniej składano takie zamówienia. Sprawdzone konkurencyjność oferowanych przez niego cen oraz czasy dostaw i złożono zamówienie.

Klient „Bukowina”, 21 lutego (czwartek) **Sprzęt dostarczono**

Sprzęt został dostarczony i rozpoczęto jego instalację w serwerowni firmy ”Bukowina”.

Klient „Bukowina”, 27 lutego (środa) **Gotowa platforma sprzętowa**

Na sprzęcie zainstalowano system operacyjny oraz wszelkie oprogramowanie wymagane przez wewnętrzne polityki bezpieczeństwa. Zainstalowano i przetestowano również oprogramowanie typu VPN, które umożliwiło zdalny dostęp do serwerów firmie wdrożeniowej.

Klient „Bukowina” — PMBoK	Wykonawca „Mefisto” — MSF
29 lutego 2008 (piątek) <p>Przy dużym współudziale partnera wdrożeniowego i wykorzystaniu dwóch roboczości eksperta w celu konsultacji zbiorczych ustalono ostatecznie:</p> <ul style="list-style-type: none"> ♦ 5.2. Zakres projektu, ♦ 5.3. Strukturę pakietów roboczych, ♦ 6.2. – 6.4. Listę czynności do wykonania i czas ich trwania, ♦ 6.5. Harmonogram projektu, ♦ 7.1 – 7.2. Budżet projektu, ♦ 11.2 – 11.4. Rejestr ryzyk, ♦ 11.5. Sposób reakcji na ryzyka, ♦ 4.6. Sposób wprowadzania zmian do projektu (zintegrowane zarządzanie zmianami). <p>Ustalono, że po odbiorze produktu firma wdrożeniowa będzie miała zablokowany dostęp do systemu z zewnątrz. Dostęp ten będzie odblokowywany na żądanie, gdy zajdzie taka konieczność. Obie firmy mają podpisaną klauzulę o zachowaniu poufności pod groźbą wysokiej kary umownej.</p>	<p>Główną część prac w tej fazie wykonali analityk i kierownik projektu Michał przy wsparciu wdrożeniowca oraz kierownika produktu Marcina. Przygotowano szczegółowy plan wdrożenia, w którym:</p> <ul style="list-style-type: none"> ♦ zweryfikowano bazowe założenia kierownika produktu Marcina, ♦ spisano listę wszystkich funkcjonalności planowanego rozwiązania, czyli projekt koncepcyjny, ♦ ustalono hierarchię produktów oferowanych przez firmę „Bukowina”, ♦ ustalono plan zaangażowania pracowników firmy „Bukowina”, ♦ określono czas i zakres testów wewnętrznych, ♦ wyznaczono szczegółowy czas i zakres pilotażu, ♦ opracowano harmonogram projektu, ♦ omówiono listę ryzyk i sposób zarządzania nimi.

Ryzyka

Wspólnie spisano rejestr ryzyk (tabela 12).

Skala oceny prawdopodobieństwa wystąpienia ryzyka od 0 do 1:

- 0 — na pewno się nie wydarzy,
- 0,2 — bardzo nieprawdopodobne,
- 0,4 — raczej nieprawdopodobne,
- 0,5 — prawdopodobne,
- 0,6 — raczej się wydarzy,
- 0,8 — bardzo możliwe, że się wydarzy,
- 1 — na pewno się wydarzy.

Wpływ na cel projektu od 0 do 1:

- 0 — nie ma bezpośredniego wpływu na cel projektu,
- 0,2 — ma minimalny bezpośredni wpływ na cel projektu,
- 0,4 — ma pewien bezpośredni wpływ na cel projektu,
- 0,5 — ma bezpośredni wpływ na cel projektu,
- 0,6 — ma istotny, bezpośredni wpływ na cel projektu,
- 0,8 — ma bardzo istotny, bezpośredni wpływ na cel projektu,
- 1 — ma fundamentalny, bezpośredni wpływ na cel projektu.

Wyróżniono dwa główne typy ryzyk — techniczne i organizacyjne.

Harmonogram

Obopólnie ustalono kluczowe terminy wykonania poszczególnych działań, zsynchronizowanych z kamieniami milowymi kierownika projektu Bartka („Bukowina”):

- ♦ (B) 3 marca — uruchomienie projektu i rozpoczęcie instalacji rozwiązania,
- ♦ 15 kwietnia — oczekiwana jest wewnętrzna premiera nowej wersji produktu i uruchomienie wewnętrznych testów w firmie wdrożeniowej „Mefisto”,
- ♦ (C) 28 kwietnia — zamknięcie wdrożenia całego rozwiązania na starej wersji produktu i pełne zamodelowanie kostek OLAP, przygotowanie raportów i formularzy do ręcznego wpisywania danych,
- ♦ (D) 19 maja — zamknięcie integracji z istniejącym systemem ERP,
- ♦ 2 czerwca — system zaktualizowany do najnowszej wersji produktu, rozpoczęcie testów funkcjonalnych i wydajnościowych,

Tabela 12. Przykładowy rejestr ryzyk w projekcie wdrożenia narzędzia typu BI

Rodzaj ryzyka	Typ	Właściciel	Prawdopodobieństwo wystąpienia (0 – 1)	Wpływ na koszt (0 – 1)	Wpływ na czas (0 – 1)	Wpływ na zakres (0 – 1)	Wpływ na jakość (0 – 1)	Siła ryzyka	Plan
1 Opóźni się data premiery nowej wersji produktu	tech.	PM „Mefisto”	0,5	0,1	1	1	0,8	1,45	Praca na starej wersji i aktualizacja wersji wdrożenia (ang. <i>upgrade</i>) tuż przed pilotażem
2 Nowa wersja produktu będzie zawierała pewną liczbę błędów	tech.	PM „Mefisto”	0,7	0,6	0,6	0,4	1	1,82	Jak najszybsze testy po stronie firmy wdrożeniowej
3 Proces aktualizacji wersji wygeneruje specyficzne błędy	tech.	PM „Mefisto”	0,3	0,4	0,4	0	0	0,24	Testy scenariusza aktualizacji po stronie firmy wdrożeniowej
4 Niskie zaangażowanie użytkowników i poprawki w ostatniej chwili	org.	PM „Bukowina”	0,2	0,4	0,8	0,7	1	0,58	Ankieta zapotrzebowań, szkolenia, badanie zadowolenia
5 Niewystarczająca łatwość używania dla użytkowników	org.	PM „Bukowina”	0,7	0,5	0,6	0,7	0,8	1,82	Ankieta zapotrzebowań, szkolenia, badanie zadowolenia
6 Niższa niż planowana wydajność rozwiązania	tech.	PM „Mefisto”	0,2	1	0	0,2	0,5	0,34	Testy wydajnościowe
7 Syndrom „brudnych danych”	tech.	PM „Bukowina”	0,9	0,2	0,6	0,4	1	1,98	Testy i analiza danych przez pracowników firmy Bukowina

- ♦ 16 czerwca — zamknięcie testów wewnętrznych i uruchomienie pilotażu, start szkoleń serii I,
- ♦ (E) 30 czerwca — zakończenie pilotażu i rozpoczęcie wdrożenia całego rozwiązania, start szkoleń serii II,
- ♦ 14 lipca — zamknięcie fazy wdrożenia i uruchomienie testów akceptacyjnych,
- ♦ (F) 25 lipca — zamknięcie fazy testów akceptacyjnych i wdrożenia, odbiór wraz dokumentacją,
- ♦ (G) koniec roku — zamknięcie projektu.

Bieżący stan wykonania projektu z punktu widzenia PMBoK wygląda tak, jak przedstawiono w tabeli 13.

Tabela 13. Tabela stopnia realizacji procesów PMBoK (3) w przykładowym wdrożeniu narzędzia klasy BI

Dokument		Katalog	Uwagi	Wykonanie
4.1. Dokument otwarcia	I	4. Zarządzanie\ Dokument otwarcia		100%
10.1. Rejestr interesariuszy	I	10. Komunikacja		100%
4.2. Plan zarządzania projektem	P	4. Zarządzanie	Plan w katalogu głównym	100%
5.2. Zakres projektu	P	5. Zakres projektu	Częścią tego dokumentu są zebrane wymagania (5.1)	100%
5.3. Struktura pakietów roboczych (WBS)	P	5. Zakres projektu		100%
6.5. Harmonogram	P	6. Harmonogram	W ramach harmonogramu realizowane są procesy 6.1 – 6.5	100%
7.2. Budżet projektu	P	7. Budżet	W ramach harmonogramu realizowane są procesy 7.1 – 7.2	100%
8.1. Plan jakości	P	8. Jakość		100%
10.2. Plan komunikacji	P	10. Komunikacja		100%
11.1. Plan zarządzania ryzykiem	P	11. Ryzyko		100%
11.5. Rejestr ryzyk	P	11. Ryzyko	W ramach harmonogramu realizowane są procesy 11.2 – 11.5	100%
12.1. Plan dostaw	P	12. Dostawy		100%
4.3. Kierowanie i zarządzanie realizacją projektu	R	4. Zarządzanie\ Rejestr zdarzeń		0%
		6. Harmonogram		

Tabela 13. Tabela stopnia realizacji procesów PMBoK (3) w przykładowym wdrożeniu narzędzia klasy BI — ciąg dalszy

Dokument		Katalog	Uwagi	Wykonanie
8.2. Przeglądy jakościowe	R	8. Jakość\Przeglądy wewnętrzne	Bieżąca kontrola jakości	0%
10.3. Okresowe raporty	R	10. Komunikacja\Raporty		0%
12.2. Oferty od sprzedawców i wybór	R	12. Zaopatrzenie\Oferty		100%
4.5. Rejestr zmian	K	4. Zarządzanie\Zmiany		0%
5.5. Weryfikacja i kontrola zakresu	K	8. Jakość\Przeglądy	Kierownik będzie dokonywał przeglądów wraz z audytorem skoncentrowanym na jakości	0%
6.6. Kontrola harmonogramu	K	6. Harmonogram	Aktualizacja harmonogramu	0%
7.3. Kontrola kosztów	K	7. Budżet\RaportyEVM		0%
8.3. Kontrola jakości	K	8. Jakość\Przeglądy audytowe	Kontrola zapewnienia jakości w projekcie	0%
10.5. Raportowanie wydajności	K	7. Budżet\RaportyEVM	Z audytami eksperta zewnętrznego	0%
11.6. Monitorowanie ryzyk i nadzór nad nimi	K	11. Ryzyko		0%
12.3. Administrowanie kontraktami	K	12. Zaopatrzenie\Rejestr zdarzeń		0%
4.6. Zamknięcie projektu	Z	4. Zarządzanie\Zamknięcie		0%
12.4. Zamknięcie kontraktów	Z	12. Zaopatrzenie\Zamknięcie		0%
Wykonanie kamieni milowych 2/6				

Klient „Bukowina”, 10 marca (poniedziałek)

Tydzień po uruchomieniu projektu kierownik projektu Bartek udał się na dwa tygodnie urlopu.

Wykonawca „Mefisto”, 20 marca (czwartek)

Wykonano pierwszy, całościowy model wielowymiarowej bazy danych i wypełniono go przykładowymi danymi. Do weryfikacji przedstawiono fizyczny model kluczowych struktur analitycznych, takich jak typ klienta, hierarchię produktów i usług.

Klient „Bukowina”, 25 marca (wtorek)

Kierownik projektu Bartek wraz z niezależnym ekspertem spędzili cały dzień w siedzibie firmy wdrożeniowej, monitorując postęp projektu. Są to działania związane m.in. z realizacją procesów:

- ♦ 5.4. Weryfikacja zakresu,
- ♦ 5.5. Kontrola zakresu,
- ♦ 8.2. Zapewnienie jakości,
- ♦ 12.3. Administrowanie dostawami.

Prace te zostały zebrane w zbiorczy raport o stanie, który został umieszczony na serwerze FTP, a rada nadzorcza została o tym poinformowana e-mailem, zgodnie z procesem 10.2. Dystrybucja informacji.

Wykonawca „Mefisto”, 2 kwietnia (środa)

Przygotowano wszystkie kluczowe miary KPI i dwadzieścia cztery przykładowe raporty oraz sześć formularzy do wpisywania danych.

Wykonawca „Mefisto”, 11 kwietnia (piątek)

Rozpoczęto wewnętrzne testy wdrożenia.

Wykonawca „Mefisto”, 22 kwietnia (wtorek)

Po trzech seriach poprawek dział testów zaakceptował wdrożenie od strony jakościowej i odnotowano trampolinę braku błędów (ang. *Zero Bug Bounce*). Rozpoczęto prace integracyjne z istniejącym systemem ERP. Kierownik projektu Michał poinformował o tym drogą e-mailową kierownika projektu Bartka.

Klient „Bukowina”, 22 kwietnia (wtorek)

Bartek na dzień następny zapowiedział wizytę w siedzibie firmy wdrożeniowej.

Klient „Bukowina”, 23 kwietnia (środa)

Bartek wraz z niezależnym ekspertem spędzili cały dzień w siedzibie firmy wdrożeniowej. Zweryfikowali zakres projektu względem pierwotnych planów. Zwrócili uwagę na pewne zmiany modelu w stosunku do planów; zmiany te idą we właściwym kierunku, ale brakuje dokumentacji na temat podjętych decyzji. Sporządzono protokół różnic i kierownik projektu Bartek oficjalnie zatwierdził zmiany planów. Oprócz procesów, które zostały wykonane miesiąc wcześniej, dokonano również wspólnego przeglądu rejestru ryzyk (11.6. Monitorowanie ryzyk i nadzór nad nimi).

Klient „Bukowina”, 25 kwietnia (piątek)

Kierownik projektu Bartek wraz z audytorem wykonali szereg działań związanych z procesami PMBoK. Oto one.

- ♦ **8.3. Kontrola jakości.** W trakcie wizyty w firmie wdrożeniowej niezależny ekspert przeprowadził dłuższą rozmowę z testerem firmy wdrożeniowej na temat funkcjonowania zasad zapewnienia bieżącej jakości. Dokonano wspólnego przeglądu scenariuszy testowych oraz powtórzono wykonanie jednego z nich. W przygotowanym raporcie znalazły się zbiorcze informacje na temat liczby wykonanych testów, usuniętych błędów, pracy poświęconej na zapewnienie jakości itp. Zauważono brak wykonania scenariuszy wydajnościowych.
- ♦ **8.2. Zapewnienie jakości.** Niezależny ekspert otrzymał kopię całego modelu bazy danych i aplikację wypełniającą model danymi. Wykorzystując te pakiety oraz swoje narzędzia, wykonał szereg testów wydajnościowych. Sporządzony raport zawierał pewne sugestie zmian, ale finalnie stwierdzono, iż zaproponowany model powinien odpowiadać zapotrzebowaniom firmy „Bukowina”.
- ♦ **6.6. Kontrola harmonogramu.** Odnotowano cztery dni przyspieszenia realizacji kamienia milowego C, czyli ukończenie modelu kostek OLAP.
- ♦ **4.4. Monitorowanie i nadzór nad pracami projektu.** Wykonano ogólne porównanie postępu prac z planami pod kątem zakresu i harmonogramu.
- ♦ **7.3. Nadzór nad kosztami.** W trakcie wykonywania prac pojawiły się pewne dodatkowe zapotrzebowania, nieprzewidziane w planie, takie jak dodatkowe licencje na aplikacje VPN. Zakupiony sprzęt okazał się nieco droższy. Mimo to, rezerwy budżetowe wciąż pozostają nienaruszone. Wydano 320 tys. w stosunku do planowanych 430 tys.
- ♦ **10.5. Raportowanie wydajności.** Na podstawie uzyskanych informacji opracowano raport zbiorczy, który prezentował postęp prac względem czasu, zakresu i kosztów.
- ♦ **4.3. Zarządzanie projektem.** Zaktualizowano rejestr dokumentów projektowych.
- ♦ **10.3. Dystrybucja informacji.** Skopiowano dane projektowe na FTP.
- ♦ **10.4. Zarządzanie oczekiwaniami interesariuszy.** Na poniedziałek zaplanowane zostało spotkanie z dyrektorami, które ma na celu prezentację stanu prac projektu.

Klient „Bukowina”, 5 maja (poniedziałek)

W trakcie krótkiego spotkania kierownik projektu Bartek zaprezentował przygotowany raport, który zawierał informacje o szybszym postępie projektu i mniejszym wykorzystaniu budżetu. Bartek poinformował również o realizacji kolejnego kluczowego kamienia milowego C, czyli o ukończeniu modelu kostek OLAP.

Wykonawca „Mefisto”, 5 maja (poniedziałek)

Ponad 2 tygodnie po terminie pojawiła się oficjalna, nowa wersja produktu. Firma „Mefisto” niezwłocznie zaczęła testy wewnętrzne i poinformowano o tym fakcie kierownika projektu Bartka („Bukowina”).

Klient „Bukowina”, 6 maja (wtorek)

Podniesiono parametry wpływu ryzyka numer 1, „Opóźni się data premiery nowej wersji produktu”.

Wykonawca „Mefisto”, 8 maja (czwartek)

Zakończono prace nad integracją z systemem ERP za pomocą narzędzia typu ETL (ang. *Extract, Transform and Load*). Odnotowano, że instalacja systemu ERP jest o wiele bardziej specyficzna, niż podejrzewano. Udało się jednak efektywnie zintegrować oba rozwiązania. Przyjęto, że co tydzień, w piątek wieczorem, będzie miała miejsce agregacja oraz wysłanie danych z bazy relacyjnej do bazy wielowymiarowej.

Wykonawca „Mefisto”, 19 maja (poniedziałek)

Testy nowej wersji produktu przeciągają się i obecna wersja ma raczej status kandydata, a nie wersji finalnej. Zgłoszono kwestie naprawy kluczowych problemów, które odnaleziono w trakcie testów wewnętrznych, i naciskano na ich realizację. Na 2 czerwca planowana jest publikacja kolejnej propozycji nowej wersji produktu BI. Kierownik projektu Michał wraz z kierownikiem produktu Marcinem podjęli decyzję o „przećwiczeniu” ścieżki upgrade’u. Ma to na celu szczegółowe zaplanowanie przejścia ze starej do nowej wersji produktu w kontekście klienta „Bukowina”.

Mimo dużego spiętrzenia prac testowych wykonano na czas testy integracji z systemem ERP oraz testy wydajnościowe. Niestety, w rejestrze wciąż widnieją pewne drobne, nierozwiązane błędy. Po dwóch seriach poprawek zaakceptowano przygotowane pakiety do transformacji i przekazania danych (ETL). Równocześnie zaznaczono, że wykonane testy nie zawierały walidacji merytorycznej samych danych, a jedynie techniczną poprawność wykonanych operacji.

Klient „Bukowina”, 19 maja (poniedziałek)

Bartek wraz z niezależnym ekspertem spędzili cały dzień w siedzibie firmy wdrożeniowej. Zweryfikowali zakres projektu względem pierwotnych planów. Ponownie zwrócono uwagę na pewne zmiany względem planów. Zagrożono, że kolejne takie działania bez zgody kierownika projektu Bartka będą miały konsekwencje finansowe. Ponownie sporządzono protokół różnic i udokumentowano akceptację. Dokonano również wspólnego przeglądu rejestru ryzyk (**11.6. Monitorowanie ryzyk i nadzór nad nimi**).

Klient „Bukowina”, 20 maja (wtorek)

Tak jak miesiąc wcześniej (25 kwietnia) kierownik projektu Bartek wykonał szereg działań związanych z procesami PMBoK. Oto one.

- ♦ 6.6. Kontrola harmonogramu — odnotowano realizację na czas kamienia milowego D, czyli ustanowienia komunikacji ERP – OLAP.
- ♦ 4.4. Monitorowanie i nadzór nad pracami projektu — wykonano ogólne porównanie postępu prac z planami pod kątem zakresu i harmonogramu.
- ♦ 8.3. Kontrola jakości — przeprowadzono audyt sposobu pracy testerów firmy wdrożeniowej „Mefisto”. Niezależny ekspert przygotował i przekazał audyt integracji z systemem ERP.
- ♦ 8.2. Zapewnienie jakości — na tym etapie zapewnienie jakości leży po stronie firmy wdrożeniowej „Mefisto”.
- ♦ 10.2. Raportowanie postępu prac — na podstawie uzyskanych informacji napisano raport zbiorczy, który prezentował postęp prac względem czasu, zakresu i kosztów.
- ♦ 4.3. Zarządzanie projektem — zaktualizowano rejestr dokumentów projektowych.
- ♦ 7.3. Nadzór nad kosztami — wzięto pod uwagę konieczność zakupu dodatkowej licencji do administrowania i „tuningowania” bazy danych.

Wykonano również analizę przy użyciu techniki Zarządzania Wartością Wypracowaną (EVM) — rysunek 52.

Raport EVM za maj

ANALIZA WYNIKÓW TECHNIKI EVM WYKONANA PRZEZ EKSPERTA (MAJ)

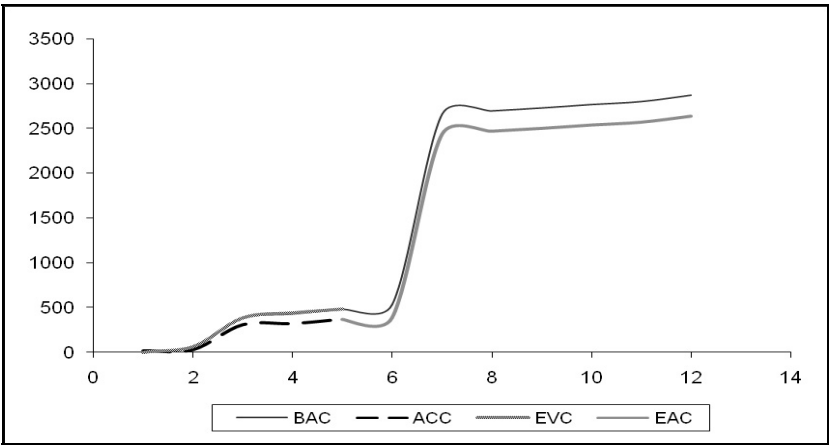
Analiza wskaźników

- ♦ CPIC wskazuje na wysoką efektywność pracy (+30%), ale jest ona głównie wynikiem niewykorzystania rezerw budżetowych, czyli kosztów mniejszych niż planowane.
- ♦ ETC wskazuje na sumę kosztów do poniesienia, aby zamknąć projekt — jest ona nieco niższa od przewidywań.
- ♦ EAC wskazuje na nową, sumaryczną wartość projektu — jest ona nieco niższa od przewidywań.

Pomimo tych prognoz, sugerowane jest pozostanie przy oryginalnym planie budżetu i zbieranie owoców „nadmiernych zapasów” na końcu projektu.

Trendy realizacji harmonogramu

Dość niepokojący jest fakt, że w kwietniu projekt miał cztery dni przyspieszenia, a już w maju był „na czas”. Istnieje więc szansa, że kolejny raport wykaże opóźnienia. Rezultaty z dwóch kamieni milowych to jednak za mało, aby wyciągać pewne wnioski. Patrząc na projekt w szerszym kontekście, można stwierdzić, że przebiega we właściwy sposób.



Miesiące	Planowane koszty częstkowe (PV)	Skumulowany koszt zaplanowany (BAC)	Skumulowany koszt rzeczywisty (ACC)	Skumulowane wartości wypracowane (EVC)	Skumulowany przewidywany koszt (EAC)
Styczeń	13,5	13,5	13,5	13,5	
Luty	47,8	61,3	26,3	61,3	
Marzec	318,4	379,7	306,7	384	
Kwiecień	49,8	429,5	319,9	439,48	
Maj	52,2	481,7	369,8	481,7	369,8
Czerwiec	53,3	535			387,73
Lipiec	2121,6	2656,6			2430,37
Sierpień	39,32	2695,92			2468,122
Wrzesień	32,62	2728,54			2499,604
Październik	38,82	2767,36			2536,806
Listopad	32,62	2799,98			2568,288
Grudzień	71,22	2871,2			2635,63
			CPIC	1,30	
			ETC	2265,83	
			EAC	2635,63	

Rysunek 52. Przykład techniki EVM (1) w projekcie wdrożenia rozwiązania typu BI

Prognozy wykonania budżetu

Wykonany raport EVM wykazał, że rezerwy budżetowe nie są w pełni wykorzystywane. Wykonano pewne, nowe prognozy wykonania budżetu i stwierdzono, że 10% rezerwy budżetowe powinny w zupełności wystarczyć zamiast pierwotnych 20%. Sugeruje się jednak pozostawienie budżetu oraz sposobu jego realizacji w obecnym kształcie.

Wnioski generalne

W styczniu i lutym projekt był realizowany zgodnie z planem, bo był to czas planowania — nie są to więc dane w pełni miarodajne. Dane z kolejnych miesięcy są jednak wiarygodne i wskazują na właściwy przebieg prac.

Obszar	Status	Komentarz
Jakość	Zielony	Pozytywny audyt procedur jakościowych u wykonawcy.
Zakres	Żółty	Wykonawca dwukrotnie nie udokumentował wprowadzonych zmian do systemu.
Budżet	Zielony	Wykonanie budżetu poniżej planów.
Harmonogram	Zielony	Projekt jest realizowany na czas.

Rada nadzorcza została poinformowana o stanie prac telefonicznie — dalsze szczegóły znajdują się w dokumentacji umieszczonej na serwerze FTP.

Wykonawca „Mefisto”, 5 czerwca (czwartek)

Cztery dni po czasie pojawiła się propozycja nowej wersji produktu. Dział testów rozpoczął kolejną iterację testów wewnętrznych.

Wykonawca „Mefisto”, 16 czerwca (poniedziałek)

Pomimo pewnych drobnych błędów, dział testów zaakceptował propozycję nowej wersji produktu i potwierdził, że może być wdrażana u klienta „Bukowina”.

Kierownik projektu Michał uruchomił w trybie pilnym aktualizację systemu u klienta.

Klient „Bukowina”, 16 czerwca (poniedziałek)

Kierownik projektu Bartek odnotował opóźnienie w stosunku do planu bieżącego, ale przyjął wyjaśnienia wykonawcy. Docenił również otwartość w komunikacji problemu¹.

Wykonawca „Mefisto”, 18 czerwca (środa)

Dzięki wcześniejszemu „przećwiczeniu” upgrade’u, prace zostały wykonane sprawnie. Po ogólnych testach wdrożenia podjęto decyzję o natychmiastowym uruchomieniu pilotażu rozwiązania. Oprócz pracowników w centrali firmy, do pilotażu postanowiono zaangażować sześć regionalnych salonów sprzedaży. Ma to na celu przetestowanie komunikacji między użytkownikami terenowymi a użytkownikami w biurze centralnym.

Odnotowano wykonanie jednego z kluczowych, pośrednich kamieni milowych — kompletność pilotażu (MSF) — i rozpoczęto dwudniowe szkolenia u klienta.

¹ W przeciwnym przypadku miałyby do czynienia z tzw. „zamiataniem pod dywan” (ang. *schedule chicken*).

Klient „Bukowina”, 19 czerwca (czwartek)

Dwanaście licencji na oprogramowanie zakupiono już wcześniej. W trakcie wdrożenia pilotażowego konieczne było dokupienie jednego serwera komunikacyjnego oraz lepszego łącza między centralą a jednostkami terenowymi.

Odnotowano trzydniowe opóźnienie w realizacji wstępnego harmonogramu. Postanowiono jednak zachować dwutygodniowy plan pilotażowy.

Klient „Bukowina”, 2 lipca (środa)

Po dwóch tygodniach zakończono okres pilotażu rozwiązania, w którym uczestniczyli wszyscy kontrolerzy finansowi oraz sześć z sześćdziesięciu sześciu sklepów w sześciu różnych województwach.

- ♦ **4.6. Zintegrowane zarządzanie zmianami.** Ostatniego dnia kierownik projektu Bartek przeprowadził rozmowy telefoniczne i osobiste ze wszystkimi użytkownikami systemu na bazie przygotowanej listy pytań. Przygotowany został zbiorczy raport. Oto jego zawartość.
- ♦ Wykryto szereg nieprawidłowych danych w systemie ERP, czyli tzw. syndrom „brudnych danych”. Zaproponowano również modyfikację niektórych kluczowych identyfikatorów wydajności (KPI). Poza tym, ustalono rutynowy mechanizm kontroli tych kwestii.
- ♦ Szczegóły na temat znalezionych błędów, z których cztery były krytyczne, a aż trzy z nich dotyczyły błędnych mechanizmów pobierania danych z systemu ERP.
- ♦ Propozycje kilkunastu zmian w obszarze interfejsu oraz funkcjonalności, z czego dwie muszą zostać wprowadzone, ponieważ były wymieniane przez około 70% osób.
- ♦ Zdefiniowano potrzebę utworzenia dodatkowych raportów i formularzy do wprowadzania danych.
- ♦ Odnotowano generalną, pozytywną informację zwrotną od użytkowników systemu, którzy bardzo chętnie korzystali z mechanizmów pracy grupowej, m.in. komentarzy.

Wraz z niezależnym ekspertem ponownie wykonano tę samą, rutynową grupę procesów PMBoK, co 20 maja i 25 kwietnia.

Po spotkaniu z dyrektorami (proces **10.4. Zarządzanie oczekiwaniami** interesariuszy) oficjalnie zaakceptowano pilotaż rozwiązania, czyli osiągnięcie kamienia milowego E, oraz uzyskano zezwolenie na produkcyjne uruchomienie systemu.

Na 16 lipca zaplanowano zebranie ze wszystkimi użytkownikami systemu.

Wykonawca „Mefisto”, 2 lipca (środa)

Otrzymano raport z pilotażu rozwiązania:

- ♦ Zmodyfikowano kluczowe identyfikatory wydajności (KPI);
- ♦ Zaakceptowano większość opisanych błędów. Jeden z krytycznych błędów nie może być naprawiony, ale znaleziono dla niego obejście;
- ♦ Postanowiono spośród kilkunastu zmian wprowadzić jedynie tę część zmian, która wiąże się z minimalnym ryzykiem wystąpienia nowych błędów;
- ♦ Postanowiono zamodelować wszystkie zaproponowane raporty i formularze.

Wykonawca „Mefisto”, 15 lipca (wtorek)

Konsultanci poprawili błędy, wprowadzili zmiany i opracowali nowe raporty.

Testerzy przeprowadzili skrócony audyt jakościowy i po drobnych poprawkach zaakceptowali wdrożenie.

Odnutowano zamknięcie fazy stabilizacji (MSF).

Równolegle u klienta „Bukowina” wzbogacono system o serię „samouczków”. Wdrożeniowiec zweryfikował również dostępność rozwiązania poprzez interfejs webowy we wszystkich sześćdziesięciu sześciu sklepach firmowych.

Wykonano kopię bezpieczeństwa całego systemu i kierownik projektu Michał przekazał kierownikowi projektu Bartkowi („Bukowina”) protokół odbioru.

Klient „Bukowina”, 15 lipca (wtorek)

Bartek odnotował jeden dzień opóźnienia, ale ze względu na wysokie zaangażowanie firmy „Mefisto” postanowił ograniczyć czas poświęcony na weryfikację wdrożenia. Odebranie produktu planowane jest na 30 lipca.

Kierownik projektu Bartek ogłosił wśród użytkowników wewnętrznych pięciodniowy okres na zgłaszanie wszelkich uwag do systemu.

Klient „Bukowina”, 16 lipca (środa)

Na grupowym spotkaniu pojawiła się przynajmniej jedna osoba z każdego sklepu regionalnego, wszyscy kontrolerzy finansowi, jeden z dyrektorów i przedstawiciele firmy „Mefisto” — wdrożeniowiec i kierownik projektu Michał. Całodniowe szkolenie składało się z serii wykładów przedstawicieli firmy „Mefisto” oraz wystąpienia kierownika projektu Bartka, który tłumaczył biznesowe mechanizmy działania systemu.

Wykonawca „Mefisto”, 16 lipca (środa)

Przeprowadzono całodniowe szkolenie dla klienta oraz przekazanie wykonanej dokumentacji projektowej — instrukcje użytkownika i administratora.

Klient „Bukowina”, 22 lipca (wtorek)

♦ 4.6. Zintegrowane zarządzanie zmianami

Kierownik projektu Bartek zebrał od użytkowników listę drobnych usterek i do firmy wdrożeniowej przesłał roboczy dokument.

Wykonawca „Mefisto”, 23 lipca (środa)

Firma wdrożeniowa zaakceptowała raport z listą usterek i jeszcze tego samego dnia wykonała żądane poprawki.

Klient „Bukowina”, 23 lipca (środa)

Kierownik projektu Bartek wraz z niezależnym ekspertem ponownie wykonali tę samą grupę procesów, co 20 maja i 25 kwietnia. W ramach procesu **7.3. Nadzór nad kosztami** kierownik projektu Bartek m.in. uaktualnił stopień realizacji budżetu (tabela 14.).

Na podstawie tych informacji niezależny ekspert opracował kolejną analizę EVM.

Raport EVM za lipiec

ANALIZA WYNIKÓW TECHNIKI EVM WYKONANA PRZEZ EKSPERTA (LIPIEC)

Analiza wskaźników

CPIC spadł z 1,3 do 1,16, ale i tak jest stosunkowo wysoki, biorąc pod uwagę, że projekt zmierza ku końcowi. CPIC wskazuje na faktyczną, wysoką efektywność pracy firmy wdrożeniowej.

Trendy realizacji harmonogramu

Koszty prac są poniżej przewidywanych, a wykonana praca powyżej przewidywań. Brak konieczności wykonywania jakichkolwiek korekt.

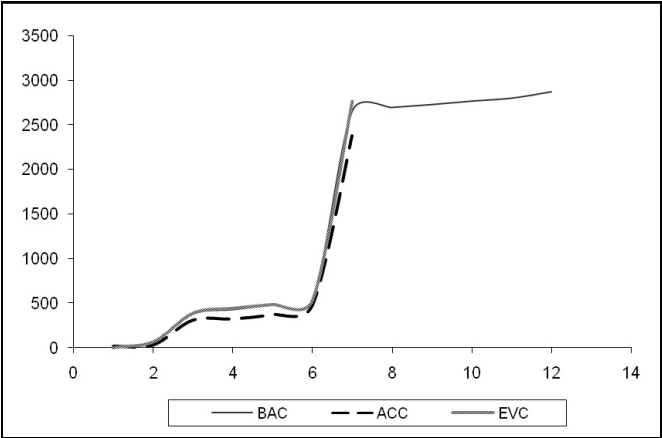
Wnioski generalne

Jedynie obszar wykonania harmonogramu budzi pewien niepokój u zewnętrznego eksperta, ale wszystkie pozostałe aspekty realizacji projektu wskazują na pozytywny przebieg spraw.

Obszar	Status	Komentarz
Jakość	Zielony	Procedury jakościowe zgodne z założeniami, niska liczba błędów.
Zakres	Zielony	Dobra znajomość zakresu przeprowadzanych prac po obu stronach i udokumentowane zmiany planu bazowego.
Budżet	Zielony	Wykonanie budżetu poniżej planów.
Harmonogram	Zielony	Dwa dni wyprzedzenia względem planu.

Tabela 14. Realizacja budżetu projektu wdrożenia rozwiązania rozwiązania klasy BI na lipiec

		SUMA	Opis	SUMA	Opis
	Rezerwa budżetowa	0		13,5	Wybór firmy wdrożeniowej
		35		47,8	Przygotowanie planu wdrożenia
		80		318	Zakup sprzętu i licencji
		35		49,8	Zamodelowane kostki
		37		52,2	Komunikacja ERP – BI
		37		53,3	Pilotaż
		180		2122	Wdrożenie
	SUMA	13,5			
	Lepsze łącze	12,8			
	Zakup licencji VPN		15		
	Licencje typu OS/Office		12		
	Zakup sprzętu		140		
	Godziny wsparcia				
	Miesięczne wsparcie				
	Praca wdrożeniowa				
	Licencje typu DB/BI		102		
	Premie eksperta				
	Roboczodni eksperta		1,4		
	Ekspertyzy eksperta	3,5	1,8		
	Premia kierownika		3,1		
	Kierownik projektu	10	2,2		
Styczeń		10			
Luty		10			
Marzec		10			
Kwiecień		10			
Maj		10			
Czerwiec		10			
Lipiec		10			



Miesiące	Planowane koszty cząstkowe (PV)	Skumulowany koszt planowany (BAC)	Skumulowany koszt rzeczywisty (ACC)	Skumulowana wartość uzyskana (EVC)
Styczeń	13,5	13,5	13,5	13,5
Luty	47,8	61,3	26,3	61,3
Marzec	318,4	379,7	306,7	384
Kwiecień	49,8	429,5	319,9	439,48
Maj	52,2	481,7	369,8	481,7
Czerwiec	53,3	535	470,4	527,005
Lipiec	2121,6	2656,6	2379,6	2762,68
Sierpień	39,32	2695,92		
Wrzesień	32,62	2728,54		
Październik	38,82	2767,36		
Listopad	32,62	2799,98		
Grudzień	71,22	2871,2		
			CPIC	1,16

Rysunek 53. Przykład techniki EVM (2) w projekcie wdrożenia rozwiązania typu BI

Kierownik projektu Bartek uaktualnił swój indeks stopnia realizacji kluczowych procesów PMBoK (tabela 15).

Klient „Bukowina”, 24 lipca (czwartek)

W trakcie spotkania z kierownikiem projektu Bartkiem i zewnętrznym ekspertem dyrektorki bardzo pozytywnie odnieśli się do całości wykonanych prac.

Tabela 15. Tabela stopnia realizacji procesów PMBoK (4) w przykładowym wdrożeniu narzędzia klasy BI

Dokument		Katalog	Uwagi	Wykonanie
4.1. Dokument otwarcia	I	4. Zarządzanie\ Dokument otwarcia		100%
10.1. Rejestr interesariuszy	I	10. Komunikacja		100%
4.2. Plan zarządzania projektem	P	4. Zarządzanie	Plan w katalogu głównym	100%
5.2. Zakres projektu	P	5. Zakres projektu	Częścią tego dokumentu są zebrane wymagania (5.1)	100%
5.3. Struktura pakietów roboczych (WBS)	P	5. Zakres projektu		100%
6.5. Harmonogram	P	6. Harmonogram	W ramach harmonogramu realizowane są procesy 6.1 – 6.5	100%
7.2. Budżet projektu	P	7. Budżet	W ramach harmonogramu realizowane są procesy 7.1 – 7.2	100%
8.1. Plan jakości	P	8. Jakość		100%
10.2. Plan komunikacji	P	10. Komunikacja		100%
11.1. Plan zarządzania ryzykiem	P	11. Ryzyko		100%
11.5. Rejestr ryzyk	P	11. Ryzyko	W ramach harmonogramu realizowane są procesy 11.2 – 11.5	100%
12.1. Plan dostaw	P	12. Dostawy		100%
4.3. Kierowanie i zarządzanie realizacją projektu	R	4. Zarządzanie\ Rejestr zdarzeń		90%
		6. Harmonogram		
8.2. Przeglądy jakościowe	R	8. Jakość\Przeglądy wewnętrzne	Bieżąca kontrola jakości	90%
10.3. Okresowe raporty	R	10. Komunikacja\ Raporty		90%
12.2. Oferty od sprzedawców i wybór	R	12. Zaopatrzenie\ Oferty		100%
4.5. Rejestr zmian	K	4. Zarządzanie\ Zmiany		90%
5.5. Weryfikacja i kontrola zakresu	K	8. Jakość\Przeglądy	Kierownik będzie dokonywał przeglądów wraz z audytorem skoncentrowanym na jakości	90%

Tabela 15. Tabela stopnia realizacji procesów PMBoK (4) w przykładowym wdrożeniu narzędzia klasy BI — ciąg dalszy

Dokument		Katalog	Uwagi	Wykonanie
6.6. Kontrola harmonogramu	K	6. Harmonogram	Aktualizacja harmonogramu	90%
7.3. Kontrola kosztów	K	7. Budżet\Raporty EVM		90%
8.3. Kontrola jakości	K	8. Jakość\Przeglądy audytowe	Kontrola zapewnienia jakości w projekcie	90%
10.5. Raportowanie wydajności	K	7. Budżet\Raporty EVM	Z audytami eksperta zewnętrznego	90%
11.6. Monitorowanie ryzyk i nadzór nad nimi	K	11. Ryzyko		90%
12.3. Administrowanie kontraktami	K	12. Zaopatrzenie\Rejestr zdarzeń		90%
4.6. Zamknięcie projektu	Z	4. Zarządzanie\Zamknięcie		0%
12.4. Zamknięcie kontraktów	Z	12. Zaopatrzenie\Zamknięcie		0%
Wykonanie kamieni milowych 6/6				

Kierownik projektu Bartek otrzymał podpis pod protokołem odbioru firmy wdrożeniowej, który jest podstawą do dokonania płatności — osiągnięto kamień milowy F, czyli zamknięcie fazy testów akceptacyjnych i wdrożenia. Rozpoczęto produkcyjną eksploatację systemu oraz postanowiono od przyszłego poniedziałku próbnie zablokować dostęp do systemu firmie wdrożeniowej.

Wykonawca „Mefisto”, 25 lipca (piątek)

Otrzymano podpisany protokół odbioru. Odnotowano zamknięcie Fazy Wdrożenia (MSF) oraz zakończono działania projektowe. Uruchomiono mechanizmy serwisowania wdrożonej aplikacji.

Klient „Bukowina” samodzielnie kontynuował wykorzystanie systemu... 12 sierpnia (wtorek)

Wykryto i zgłoszono kilka drobnych błędów, ale konsultant firmy wdrożeniowej wezwany przez kierownika projektu Bartka po dwudniowej wizycie znalazł sposób na „obejście” problemu.

1 września (poniedziałek)

Kierownik projektu Bartek, chcąc podsumować pierwszy miesiąc wykorzystania systemu, rozesłał do wszystkich użytkowników krótką ankietę.

8 września (poniedziałek)

Kierownik projektu Bartek wykonał tę samą, rutynową grupę procesów PMBoK, co 20 maja i 25 kwietnia. Dodatkowo podsumował otrzymane ankiety i przekazał dyrektorom generalną informację zwrotną na temat systemu, który zapewniał użytkownikom głównie:

- ♦ lepszą wizualizację prognoz i swobodniejszy dostęp do zagregowanych danych finansowych,
- ♦ większy wpływ na podejmowane decyzje,
- ♦ bieżącą kontrolę realizacji planów oraz większą i aktualniejszą wiedzę po stronie biura centralnego,
- ♦ dodatkowe mechanizmy komunikacji grupowej.

15 września (poniedziałek)

Kierownik projektu Bartek zebrał wszystkich kontrolerów finansowych w celu przygotowania sposobu realizacji planów budżetowych na ostatni kwartał roku.

W ramach procesu **4.6. Zintegrowane zarządzanie zmianami** zdefiniowano konieczność przygotowania dodatkowych formularzy i raportów — ustalono, że zabierze to trzy roboczodni. Od tego czasu założono tygodniowy okres ustalania planów z poszczególnymi sklepami.

17 września (środa)

Okazało się, że nowo wykryty błąd uniemożliwia wykonanie jednego z kluczowych formularzy. Problem został natychmiast przekazany przez kierownika projektu Bartka do firmy wdrożeniowej „Mefisto” w trybie priorytetowym. Otrzymano potwierdzenie, że zgłoszenie zostało zarejestrowane jako kluczowe.

22 września (poniedziałek)

Otrzymano komunikat o naprawionym błędzie — konsultant firmy „Mefisto” przekazał informację o konieczności wykonania dodatkowego formularza. Został on zaimplementowany przez jednego z kontrolerów finansowych w systemie produkcyjnym.

Jeszcze tego samego dnia jeden z kontrolerów finansowych rozesłał do wszystkich sklepów informację o planach budżetowych na ostatni kwartał.

29 września (poniedziałek)

W czternastu z sześćdziesięciu sześciu sklepów wciąż nie udało się ustalić wspólnego planu budżetowego. Kierownik projektu Bartek zaprosił kontrolerów finansowych, a następnie telefonicznie przeprowadził konferencje z każdym ze sklepów. W czasie dwóch dni udało się zamknąć plany budżetowe dla całej firmy i dla każdego kierownika sklepu z osobna. Plan zakładał o 18% wyższe zyski w stosunku do analogicznych wyników za

ten sam kwartał w przeciągu pięciu ostatnich. Każdy ze sklepów, który zrealizuje swój plan, otrzyma premię. Wielkość tej premii jest uzależniona od realizacji uzgodnionych planów i zawartego w nich przyrostu zysków.

5 stycznia 2009 (poniedziałek)

Na wniosek kierownika projektu zewnętrzny ekspert zaczął wykonanie audytowego raportu zamknięcia całego wdrożenia.

Kierownik projektu Bartek wykonał następujące czynności.

- ♦ 4.3. Zarządzanie projektem.
- ♦ 4.4. Monitorowanie i nadzór nad pracami projektu — podsumowanie.
- ♦ 5.4. Weryfikacja zakresu.
- ♦ 5.5. Kontrola zakresu.

Bartek podsumował cały okres wykorzystania systemu wewnątrz firmy oraz projekt wdrożeniowy. Stwierdził, że:

- ♦ zebrano kluczowe opinie użytkowników,
- ♦ zaprezentowano graf dziennej liczby operacji na systemie,
- ♦ zaprezentowano liczbę naprawionych błędów i zmian,
- ♦ zaraportowano stopień wykorzystania budżetu,
- ♦ załączono raport audytora.
- ♦ 6.6. Kontrola harmonogramu — zamknięto harmonogram.
- ♦ 7.3. Nadzór nad kosztami — zaktualizowano wykonanie budżetu. Wykryto, że wykorzystano zaledwie 28% zaplanowanych rezerw, ale było to spowodowane głównie przez to, iż rezerwy obejmowały również ceny licencji, które nie uległy zmianie (tabela 16).
- ♦ 8.3. Kontrola jakości — wykonano ostatni raport na temat liczby wykrytych błędów i odnotowano sporadyczną komunikację z firmą wdrożeniową w listopadzie i grudniu. System BI działa stabilnie.
- ♦ 11.6. Monitorowanie ryzyk i nadzór nad nimi — zamknięto rejestr ryzyk projektowych.
- ♦ 12.3. Administrowanie kontraktami — zamknięto roczny kontrakt z niezależnym ekspertem i wykonano finalne rozliczenia. Zamknięto również kontrakt na wdrożenie systemu BI i rozpoczęto rozmowy na temat ramowej umowy na wsparcie.
- ♦ 10.3. Dystrybucja informacji — wykonano raport zamknięcia.
- ♦ 4.7. Zamknięcie projektu — zarchiwizowano całą dokumentację projektową.

Tabela 16. Końcowa realizacja budżetu projektu wdrożenia rozwiązania klasy BI

	Kierownik projektu	Premia kierownika	Ekspertyzy eksperta	Roboczoźni eksperta	Premie eksperta	Licencje typu DB/BI	Praca wdrożeniowa	Miesięczne wsparcie	Godziny wsparcia	Zakup sprzętu	Licencje typu OS/Office	Zakup licencji VPN	Lepsze łącze	SUMA	Rezerwa budżetowa	SUMA	Opis
Styczeń	10		3,5											13,5	0	13,5	Wybór firmy wdrożeniowej
Luty	10			2,8										12,8	35	47,8	Przygotowanie planu wdrożenia
Marzec	10			1,4		102				140	12	15		280	80	318	Zakup sprzętu i licencji
Kwiecień	10		1,8	1,4										13,2	35	49,8	Zamodelowane koszty OLAP
Maj	10		3,1	2,8		34								49,9	37	52,2	Komunikacja ERP – BI
Czerwiec	10		2,2	4,2	2	48				24	10		0,2	101	37	53,3	Pilotaż
Lipiec	10		1,8	2,8	4	440	1436			14			0,2	1909	180	2122	Wdrożenie
Sierpień	10			0				14,8	3,8				0,2	28,8	5	39,3	
Wrzesień	10			0				14,8	13,3				0,2	38,3	4	32,6	
Październik	10			2,8				14,8	5,7				0,2	33,5	5	38,8	IV okres rozliczeniowy
Listopad	10			0				14,8	0				0,2	25	4	32,6	
Grudzień	10	30		0,7				14,8	0				0,2	55,7	10	71,2	Zamknięcie projektu
SUMA	120	30	12,4	18,9	6	624	1436	74,1	22,8	178	22	15	1,4	2561	432	2871	

Kierownik projektu Bartek już od trzech miesięcy koncentruje się na optymalizacji procesów biznesowych i marginalnie zajmuje się kwestiami utrzymania systemu, który zaczyna „żyć własnym życiem”. Przykładowo bez ingerencji partnera wdrożeniowego wykonywane są dodatkowe raporty i formularze.

12 stycznia 2009 (poniedziałek)

W ramach procesu **10.4. Zarządzanie oczekiwaniami interesariuszy** kierownik projektu Bartek wraz z zewnętrznym ekspertem zaprezentowali raport zamknięcia. Udało się uzyskać 16% wzrost sprzedaży za czwarty kwartał w stosunku do analogicznego okresu w latach ubiegłych. Wzrost ten jest jedynie w części zasługą nowego systemu, który jednak umożliwił wprowadzenie nowych mechanizmów kontrolnych i motywacyjnych dla poszczególnych sklepów. Kierownik projektu Bartek zaprezentował swój eksperymentalny mechanizm premiowy i zaproponował reformę procesów księgowo-finansowych zaplanowaną na rok 2009. Uzyskał absolutorium, zgodę na zamknięcie projektu, premię za jego wykonanie oraz awans na dyrektora finansowego.

16 stycznia 2009 (piątek)

Dyrektor Bartek zorganizował wyjazdowe spotkanie ze wszystkimi kierownikami sklepów oraz kontrolerami finansowymi inaugurujące nowy rok finansowy. Na spotkaniu zaprezentował retrospekcję tego, co udało się osiągnąć w roku ubiegłym. Oficjalnie podziękował za wsparcie oraz wkład w powodzenie projektu. W drugiej połowie dnia zaprezentował nowy pomysł reformy procesów księgowo-finansowych, który wywołał burzliwą dyskusję wszystkich zgromadzonych ☺.

Część III

Metodyki adaptacyjne a praktyka

Metodyki adaptacyjne mają w ostatnim okresie „dobrą prasę” i swoją popularność zawdzięczają głównie temu, że przywróciły IT „ludzką twarz”. Metodyki te nie zajmują się kwestiami związanymi ze sposobem podejmowania decyzji strategicznych, np. „uruchomić projekt, czy nie”. Dlatego też dobrze dopełniają bardziej formalne rozwiązania, takie jak Prince2 czy PMBoK.

Duża część projektów informatycznych wiąże się nierozzerwalnie z innowacyjnością, która wymaga dużego stopnia swobody przy podejmowaniu kluczowych decyzji taktycznych i stałej, bieżącej współpracy z klientem. Dodatkowo ambitne projekty mają o wiele większą szansę realizacji, jeżeli ludzie biorący w nich udział najzwyczajniej w świecie lubią swoją pracę i czerpią z niej przyjemność. Metodyki adaptacyjne koncentrują się właśnie na zmotywowaniu uczestników projektu.

Oto kluczowe cechy metodyk adaptacyjnych.

- ◆ Przedkładanie „miękkich” technik zarządczych ponad „twarde” procesy i narzędzia — oznacza to m.in. koordynacyjny tryb zarządzania projektem zamiast jednego, centralnego punktu decyzyjnego. Największą wagę mają techniki sprawdzone w praktyce, nawet jeśli czasami sprawiają wrażenie „partyzanckich”, a nie formalne procesy i obowiązkowe procedury. Istnieje pewne ryzyko anarchizacji projektu (ang. *cowboy coding*), ale metodyki adaptacyjne stawiają na „samoorganizujące się zespoły”, które proaktywnie zmierzają do wspólnie wyznaczonego celu. Gdy projekt jest źle realizowany, decydenci szybciej zorientują się w sytuacji, ponieważ każda iteracja ma dostarczać konkretną wartość biznesową. Na koniec iteracji, w trakcie prezentacji jej efektów, stworzona aplikacja działa lub nie, a zakres wykonanych prac jest widoczny.

- ♦ Przykładanie większego znaczenia do działającego oprogramowania niż do wyczerpującej dokumentacji — bardzo ciekawą techniką jest takie tworzenie interfejsu użytkownika, aby opisywał się on samodzielnie poprzez właściwe systemy interaktywnej pomocy. Przeciętny użytkownik powinien szybko sam nauczyć się sposobu obsługi oprogramowania bez korzystania z dodatkowej, papierowej dokumentacji. Podobną technikę można zastosować do kodu, który powinien być zrozumiały dla każdego programisty po zapoznaniu się z wygenerowaną automatycznie dokumentacją wykorzystującą stworzone opisy klas i metod. Istnieje tutaj pewne ryzyko, że po pewnym czasie okaże się, iż pewna część wiedzy na temat systemu została tylko w głowach kluczowych twórców projektu lub wręcz o niej zapomniano. Dlatego też bardziej skomplikowane systemy informatyczne powinny zostać dostarczone wraz z pewną dokumentacją. Jej zakres w przypadku metodyk adaptacyjnych jest jednak ograniczony do wymogów odbiorcy i kwestii związanych z utrzymaniem systemu.
- ♦ Dawanie pierwszeństwa współpracy z użytkownikiem przed negocjacją kontraktu — oznacza to częstą prezentację wyników poszczególnych iteracji projektu; widząc działający system, interesariusze projektu przedstawiają pełniejszą informację zwrotną. Dzięki temu łatwiejsze będzie również wspólne uzgadnianie dalszych kierunków rozwoju aplikacji nawet wtedy, kiedy nie do końca zgadzają się one z zapisami kontraktu. Istnieje tutaj pewne ryzyko, że projekt wymknie się spod kontroli i zupełnie przestanie odpowiadać zapisom w umowie dotyczącym zakresu oraz budżetu. Dlatego też dialog z klientem końcowym powinien — mimo wszystko — uwzględniać ramowe zapisy w kontrakcie, a przy większych zmianach jego renegotiacja może okazać się niezbędna. Metodyki adaptacyjne są jednak unikalne w kwestii nieformalnej komunikacji z klientem końcowym i wspólnego planowania działań.
- ♦ Preferowanie odpowiedzi na zmianę nad podążaniem za planem bazowym — szczegółowe plany dotyczą głównie kolejnej iteracji, a nie całego projektu. Dlatego też stosunkowo łatwo można wprowadzić zmiany do kolejnej iteracji. Metodyki adaptacyjne nie przewidują głębokiej analizy skutków wprowadzonej zmiany i dlatego istnieje zwiększone ryzyko, że będzie ona bardziej kosztowna niż planowano. Istnieje też możliwość sytuacji odwrotnej. O ile obie strony, czyli wykonawca i odbiorca ufają sobie wzajemnie, adaptacyjny tryb realizacji projektu znacząco zmniejsza „koszty komunikacyjne”.

Zastosowanie metodyki adaptacyjnej nie będzie panaceum na wszelkie zło. Każda metoda jest przecież *tylko* lub *aż* tak dobra jak ludzie, którzy ją stosują. W niesprzyjających warunkach metodyki, takie jak XP lub Scrum, mogą przybrać „pokracczną” postać.

Jednym z przejawów rosnącej popularności metodyk adaptacyjnych jest to, że wielu gracze, tacy jak IBM czy Microsoft, udostępniają adaptacyjne wersje swoich metodyk, czyli odpowiednio *Agile Unified Process* (odmiana RUP) i *MSF4 for Agile*.

Rozdział 6.

eXtreme Programming

Programowanie ekstremalne jest jednym z pierwszych zbiorów dobrych praktyk, związanych z adaptacyjnym sposobem wytwarzania oprogramowania. Znakiem rozpoznawczym tego podejścia są takie elementy jak programowanie parami, programowanie kierowane testami (ang. *Test Driven Development*) czy stałe przebudowywanie kodu (ang. *refactoring*).

Szczypta historii

Za twórcę XP uznawany jest Kent Beck (rysunek 54), który na początku lat 90. ubiegłego wieku wraz z Wardem Cunnighamem prowadził prace nad prostym i efektywnym sposobem tworzenia oprogramowania. W 1996 roku Kent, korzystając z tej metodyki, rozpoczął projekt dla DaimlerChrysler i zdefiniował m.in. cztery główne wartości metodyki: komunikację, prostotę, zwrotną pętlę informacji (ang. *feedback*) i śmiałość. W ostatnim okresie XP zostało również wzbogacone o pewne techniki znane ze Scrum, takie jak codzienne spotkania organizacyjne „na stojaka” (ang. *Stand-up meeting*). XP jest metodyką wciąż ewoluującą, która nie doczekała się jeszcze w pełni precyzyjnej specyfikacji, pozwalającej na łatwą i kompleksową implementację. Wynika to z tego, że XP świadomie unika podejścia procesowego i jest raczej zbiorem dobrych praktyk. Ich zastosowanie jest uzależnione od kontekstu konkretnej organizacji.

Rysunek 54.
Kent Beck



Role

XP wymienia wprost jedynie dwie role. Oto one.

- ♦ Klienci odpowiedzialni za zdefiniowanie życzeń i weryfikację poprawności otrzymanych produktów.
- ♦ Programiści odpowiedzialni za właściwe zaprojektowanie, wykonanie i przetestowanie rozwiązań spełniających życzenia klientów.

Proces

XP składa się z czterech kluczowych obszarów, takich jak planowanie (PL), projektowanie (PR), kodowanie (KO) i testowanie (TE). XP to głównie opis praktyk, które angażują jeden lub kilka z wymienionych powyżej obszarów. Poniżej przedstawiono dwanaście kluczowych praktyk, specyficznych dla XP. Oto one.

- 1. (PL) Planowanie gry.** Programiści i klienci współpracują, aby jak najszybciej uzyskać maksymalną wartość biznesową. Zazwyczaj odbywa się to w następujący sposób:

- ♦ Klienci przychodzą z listą żądanych funkcjonalności w postaci „opowieści użytkownika” (ang. *user stories*). Mogą one wyglądać np. tak:
 - ♦ zakup online miesięcznego zezwolenia na parkowanie,
 - ♦ zakup może odbyć się za pomocą karty kredytowej,
 - ♦ zakup może odbyć się przy użyciu systemu PayPal.

Każda z tych opowieści jest opisana na osobnej karcie, która może przybrać dowolną formę, poczynając od zwykłego tekturowego kartonika, a kończąc na zaawansowanych, elektronicznych formularzach.

- ♦ Programiści dokonują zgrubnego oszacowania kosztu wykonania oraz wstępnie analizują, ile elementów i które z nich mogą być wykonane w danej iteracji.
- ♦ Klienci podejmują decyzje, które opowieści zaimplementować i w jakiej kolejności oraz jak długie będą poszczególne iteracje.

- 2. (PL, PR, KO, TE) Małe wydania** (ang. *Releases*). Należy jak najwcześniej i najczęściej tworzyć wydania do przekazania klientom w celu weryfikacji. Klienci, widząc realizację rozwiązania, mogą lepiej sterować kierunkiem rozwoju produktu. W zależności od wielkości projektu, jeden cykl powinien trwać od jednego tygodnia do trzech miesięcy. Przekazanie wydania do klienta powinno być jak najbardziej bezobsługowe.

3. (PL, PR, KO, TE) **Metafora systemu** to łatwa do przekazania opowieść o tym, jak działa system. Powinna angażować wzorce projektowe i klasy, ale musi być zrozumiała zarówno dla programistów, jak i klientów¹. Jedną z największych bolączek w trakcie realizacji projektu jest komunikacja pomiędzy klientami i programistami, którzy mają tendencję do posługiwania się technicznym slangiem². Którą wersję wydarzeń szybciej zrozumie pani Jadwiga z księgowości? Taki techniczny opis systemu?

System interpretuje tekst jako komendy i przekazuje je do wykonania serwerowi aplikacyjnemu. Rezultatem jest zbudowanie obiektów, inicjacja komunikacji z bazą danych i tworzenie dedykowanych, przefiltrowanych paczek z danymi. Na ich podstawie nasze usługi tworzą raport i generują e-maile. Wszystko dzieje się na bazie pewnej standardowej architektury obiektów i usług (ang. *framework*). Jeżeli jest to pierwsze zamówienie danego dnia, konieczny jest dłuższy czas inicjacji całego systemu. Końcowi użytkownicy dostają gotowy raport na skrzynkę e-mailową wyspecyfikowaną w komendzie wejściowej.

Czy raczej pani Jadwiga wolalaby usłyszeć taką metaforę?

System ten jest jak pizza na zamówienie złożone za pomocą SMS-a. Jeżeli chcesz ją dostać, najpierw musisz wybrać składniki, jakie mają się na niej znaleźć, zaznaczyć ich symbole i skomponować w tekście SMS-a własne zamówienie. Wtedy krasnoludki (framework) zaczynają robić ciasto, a potem kładą te dodatki, które zostały zamówione. Jeżeli jest to pierwsze zamówienie danego dnia, trzeba trochę poczekać, aż rozgrzeją się piec. Pizza jest dowożona do domu na adres wymieniony w SMS-ie.

4. (PR) **Prosty projekt**. Projekt powinien w najprostszy możliwy sposób wykonać powierzone zadanie. Wymagania zmieniają się jutro, więc należy wykonać tylko to, co zostało wyspecyfikowane dzisiaj. Architektura rozwiązania tworzona jest przyrostowo. Praktyka ta jest również znana poza XP jako zasada YAGNI (ang. *You Are Not Gonna Need It*), którą można przetłumaczyć na eNBeTeP — Nie Będziesz Tego Potrzebował³. Oto przykład.

— *Ustaliliśmy, że będziemy musieli zapisywać domyślną wartość tego komponentu pomiędzy sesjami. Lepiej byłoby napisać ogólny obiekt, który umożliwi w przyszłości zapisanie wszystkich parametrów sesji.*

— *Zastosuj teraz YAGNI. Napisz jedną linijkę kodu! Jeśli będziemy potrzebowali zapisywać więcej niż jedną wartość, wtedy zastanowimy się nad napisaniem ogólnego obiektu.*

¹ K. Beck, *Extreme Programming Explained*, s. 179. Wersja angielska: The system metaphor is a story that everyone — customers, programmers, and managers — can tell about how the system works.

² O. Shahine, 5. *Make hard problem look easy. Don't make easy problems look hard* — <http://www.shahine.com/omar/TipsForWorkingAtMS.aspx>

³ L. Lindstrom; C. Zannier; E. Hakan (2004). *Extreme Programming and Agile Methods — XP/Agile Universe 2004: 4th Conference on Extreme Programming and Agile Methods, Calgary, Canada, August 15 – 18*. Berlin: Springer. pp. 121. ISBN 3-540-22839-X.

5. (TE) **Stale testowanie. Testy jednostkowe** powinny zostać oprogramowane za pomocą takich narzędzi jak JUnit lub NUnit. Sugerowane jest zastosowanie programowania kierowanego testami (ang. *Test Driven Development*), które zakłada wykonanie testów przed opracowaniem funkcjonalności. Każdy test jednostkowy powinien sprawdzać pojedynczą klasę lub mały zbiór klas. Oprócz tego, klienci specyfikują **testy akceptacyjne**, które badają cały system pod kątem zgodności z opowieściami użytkownika. W idealnym układzie testy akceptacyjne są również zautomatyzowane, ale często zdarza się, że nie jest to możliwe lub opłacalne.

Ciekawy eksperyment został przeprowadzony przez Jamesa W. Newkirka i Alexeia A. Vorontsova, którzy w swojej książce *Test-Driven Development in Microsoft.NET* opisują realizację projektu zgodnego z TDD. Perypetie autora zamieszczone w tej książce dobrze obrazują sformułowanie „łatwo powiedzieć” ☺.

6. (KO) **Przebudowywanie** (ang. *Refactoring*). Należy stale weryfikować kod i usuwać lub unifikować powtarzające się fragmenty. Dzięki automatycznym testom jednostkowym ewentualne błędy integracyjne są szybko wykrywane w trakcie codziennego budowania aplikacji.



W praktyce

Niestety, w praktyce często brakuje czasu na spokojne „wyczyszczenie kodu” przed przekazaniem do klienta. Zdarza się, że wygrywa reguła „jak działa, nie dotykaj”, co w dłuższej perspektywie może spowodować problemy u klienta, który rozwija rozwiązanie.

7. (KO) **Programowanie parami**. Kod produkcyjny jest pisany przez dwóch programistów siedzących przed jednym komputerem, co powoduje, że jest przeglądany w trakcie pisania. Cały zespół projektowy lub przynajmniej programiści znajdują się w jednym, fizycznym miejscu. Każda kluczowa decyzja musi zostać podjęta przez obu programistów.



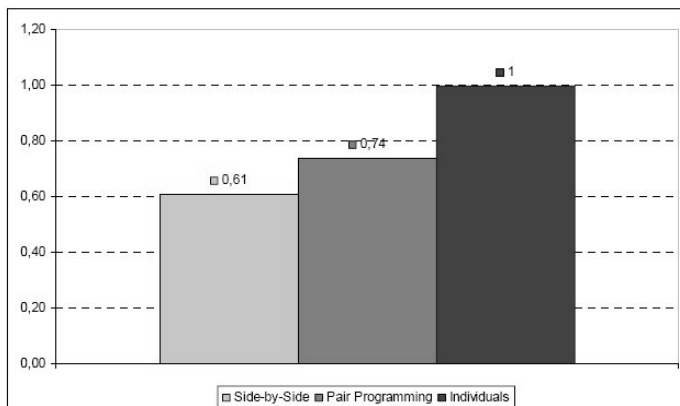
W praktyce

Co ciekawe, Scrum nie definiuje już takiego wymagania. Jeszcze ciekawsze są wyniki badań zespołu Michała Jasińskiego z Politechniki Poznańskiej, który wykazał, że programowanie „ramię w ramię” jest bardziej wydajne niż programowanie parami (rysunek 55).

8. (KO) **Kolektywne posiadanie kodu**. Żadna pojedyncza osoba „nie posiada” konkretnego modułu systemu na wyłączność od strony technologicznej. Każdy programista może pracować nad dowolną częścią systemu, bazując na dotychczas powstałym oprogramowaniu. Powoduje to konieczność opracowania standardów

Rysunek 55.

Badanie wydajności
prac różnych
wariantów
współpracy par
programistycznych⁴



Średni względny czas pracy dla indywidualistów, par XP i par SbS.

kodowania, przestrzegania ich oraz silną motywację do uniwersalizacji kodu — każdy z członków zespołu wie, że pozostałe osoby mogą za chwilę przerabiać jego pracę, a on sam może modyfikować pracę innego kolegi z zespołu.



W praktyce

Jedynym z ciekawszych rozwiązań jest wzajemna weryfikacja napisanego kodu i warto na koniec lub początek każdego tygodnia zarezerwować programistom pewien czas na takie czynności. Pary weryfikujące swoje rozwiązania powinny się zmieniać.

W ten sposób dość naturalnie cały zespół poznaje szczegóły techniczne całego rozwiązania i każdy z członków zespołu może szybko zacząć rozbudowywać jego dowolny moduł. Dodatkowo mechanizm ten, poprzez serię spontanicznych dyskusji, doprowadza do podobnego stylu pisania kodu przez każdego programistę.

9. (KO, TE) **Stala integracja.** Wszystkie zmiany są integrowane w ramach codziennych kompilacji (ang. *daily build*). Dowodem na powodzenie integracji jest m.in. wykonanie z powodzeniem testów jednostkowych i automatycznych testów akceptacyjnych. Warto monitorować stopień pokrycia kodu tymi testami.

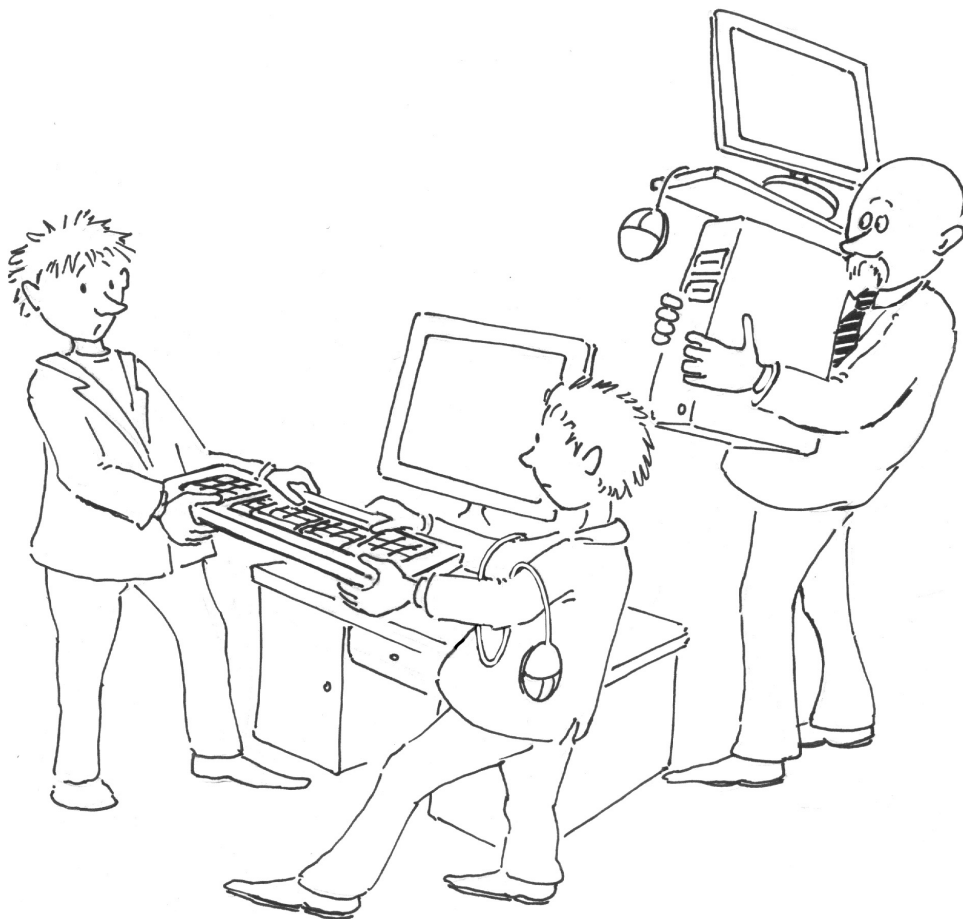


W praktyce

Istnieją firmy, w których błąd w trakcie codziennych kompilacji powoduje włączenie głośnej syreny alarmowej w całym biurze. Publicznie mówi się o osobie, która spowodowała taki błąd poprzez wprowadzenie do repozytorium wadliwego kodu. Błędy

⁴ M. Jasiński, „Programowanie parami: Razem czy osobno”, dokument dostępny na stronie http://www.inmost.org.pl/articles/Programowanie_Parami_-_razem_czy_osobno

Wujek Dobra Rada — odcinek 9. „Ekstremalny wysiłek”



ZAMIAST EKSTREMALNYCH PAR...
CO DWA KOMPUTERY „RAMIĘ W RAMIĘ”, TO NIE JEDEN

są rzeczą ludzką, ale obowiązkiem każdego programisty jest weryfikacja, czy po jego zmianach całość rozwiązania kompiluje się prawidłowo. Oprócz publicznej informacji, taka osoba może również stać się odpowiedzialna za codzienne budowanie kodu do momentu, kiedy nie pojawi się kolejny „nieszczęśnik”. Taka metoda jest dość rygorystyczna, ale skutecznie motywuje każdego z osobna i wszystkich razem do kilkukrotnej weryfikacji kodu przed umieszczeniem go w centralnym repozytorium.

- 10. (KO) Czterdziestogodzinny tydzień pracy.** Programiści oddelegowani do projektu nie pracują w nadgodzinach. Wyjątkowo dopuszczalny jest jeden tydzień „kryzysowy”, ale dwa takie tygodnie pod rząd są już sygnałem, że „coś idzie nie tak”. Długofalowo praca w nadgodzinach przynosi zgubne efekty i powoduje

więcej dodatkowych, kosztownych błędów niż korzyści. Z tego też powodu należy w taki sposób realizować projekt, aby nie było konieczności pracy ponad standardowe czterdziestu godzin tygodniowo.



W praktyce

Zdarza się, że XP jest wyśmiewana jako metoda ekstremalna szczególnie wtedy, gdy źle stosowana powoduje sześćdziesięciogodzinny tryb pracy i marsze śmierci (ang. *death march*). Tego typu sytuacje nie są jednak dowodem na słabość XP, ale raczej wskazują na złe wdrożenie metodyki lub inny problem związany z organizacją prac projektowych.

-
- 11. (PL, PR, KO, TE) Stały dostęp do klienta.** Zespół programistów ma stały, personalny dostęp do klienta, czyli osoby, która będzie używać systemu. W przypadku oprogramowania przeznaczonego dla wielu klientów taką osobą jest menadżer produktu. Tego typu podejście zmniejsza koszty komunikacyjne oraz powinno redukować ewentualne nieporozumienia.



W praktyce

Metoda XP nie powinna być interpretowana jako zupełnie swobodny tryb realizacji projektu pozbawiony jakiejkolwiek kontroli zakresu. Pomimo swobodnego trybu komunikacji, warto pewne kluczowe ustalenia zapisać w jakiegokolwiek formie pisemnej, nawet w postaci e-maila lub krótkiej notatki ze spotkania. Bardzo często po pewnym czasie ustne ustalenia mogą ulec zapomnieniu lub wypaczeniu i powodować niepotrzebne dyskusje. Forma pisemna pozwala na bardziej merytoryczne podejście do zagadnienia.

-
- 12. (KO) Standardy kodowania.** Wszyscy kodują na bazie tego samego standardu. W idealnej sytuacji nie ma możliwości odróżnienia, kto z zespołu wykonał konkretną część kodu.



W praktyce

Stworzenie lub przyjęcie standardów kodowania nie jest zadaniem trudnym. Sedno problemu tkwi w mechanizmach kontroli, czy standardy te są stosowane.

Z jednej strony, można wspomagać się takimi narzędziami jak fxCop, PMD lub FindBugs. Mogą one generować raporty po zakończeniu codziennych kompilacji, ale wszystkie zalecenia należy od razu implementować. Za każdym razem powinien zostać przeprowadzony krótki rachunek zysków i kosztów, którymi mogą być nowe, niespodziewane błędy.

Z drugiej strony, współdzielenie całego rozwiązania przez wszystkich członków zespołu powoduje, że z biegiem czasu każdy programista przyjmuje podobny styl kodowania. Wystarczy upewnić się, że styl ten jest zgodny z przyjętymi standardami. Jeżeli nie jest, równie dobrze może to wskazywać na pewien problem w zespole, jak i na fakt, że standardy nie przystają do rzeczywistości projektowej.

Wdrożenie



W praktyce

Każdy klient będzie zadowolony z możliwości elastycznej, bieżącej zmiany planów. Niestety, bardzo często osoby niezaznajomione z XP nie zdają sobie sprawy z „drugiej strony” medalu. Szczególną uwagę należy zwrócić na poniższe fakty.

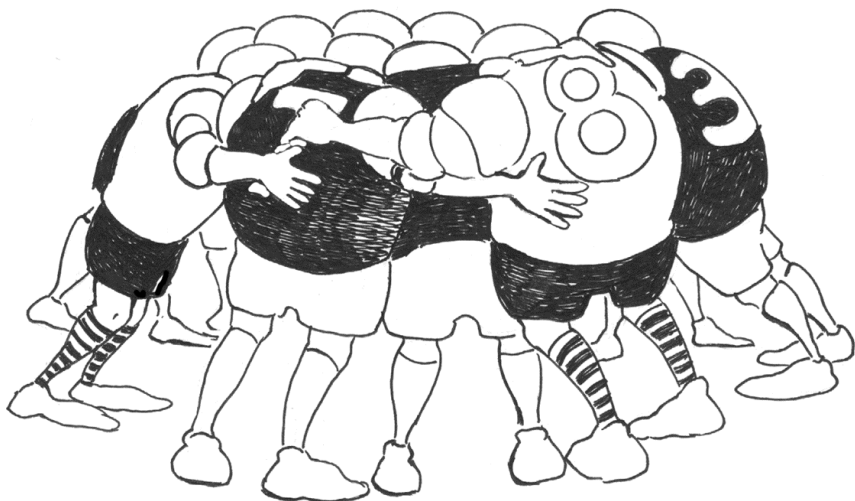
- ♦ W projektach o stałej cenie i relacji klient-wykonawca sprawą kluczową jest kontrola zakresu projektu. W przypadku XP zakres ten może ewoluować i zmieniać się w czasie. Celem nadrzędnym metodyki jest końcowa satysfakcja klienta, ale powinna ona zostać osiągnięta za zakontraktowaną cenę. XP zakłada, że w relacji klient-wykonawca negocjacje prowadzone są w duchu obopólnego zrozumienia (strategia wygranej dla obu stron). Jednak czasami może się zdarzyć, że klient lub wykonawca będzie wykorzystywał sytuacje dyskusyjne na swoją korzyść w sposób nadmierny. Tego typu kwestie leżą poza zakresem XP.
- ♦ (5) Należy pamiętać o tym, że automatyzacja testów jednostkowych powoduje wydłużanie czasu programowania od 1,5 do 2 razy. W innych metodykach czas podobny lub większy jest poświęcany na końcową fazę testów. W każdym projekcie należy przeanalizować, w jakim zakresie opłacalne jest wykonanie testów automatycznych.
- ♦ (6) Refaktoring to dodatkowa czynność, która również wydłuża czas programowania. Długofalowo koszt utrzymania nieprzebudowanego kodu jest jednak o wiele wyższy niż koszt natychmiastowego usunięcia jego powtarzalnych fragmentów. W trakcie planowania iteracji warto więc uwzględnić pewien czas na tego typu czynności.
- ♦ (7) Ze względu na powszechnie występujący syndrom „krótkiej kołdry” i brak wolnych ludzi, programowanie parami to dla wielu firm spory „luksus”. W praktyce można z powodzeniem wykorzystać podejście „ramię w ramię” i kluczową kwestią jest zebranie całego zespołu wytwórczego w jednym miejscu. Nie zmienia to faktu, że przy okazji bardziej skomplikowanych prac lub modułów programiści w sposób naturalny i spontaniczny mogą pracować wspólnie przed jednym komputerem.

- ♦ (11) Zdarza się, że stały dostęp do klienta końcowego jest trudny do wyegzekwowania. W przypadku klienta o niskim zaangażowaniu możemy mieć do czynienia z sytuacją, w której im więcej szczegółowych pytań zadają programiści, tym mniej jest osób, które mają czas na nie odpowiadać. Wtedy wykonawca może zdecydować o wykonaniu pierwszej wersji prototypowej. Po prezentacji często pojawia się grupa osób chętnych do weryfikacji wykonanych prac 😊.
-

Rozdział 7.

Scrum

Scrum to grupa rugbyistów kotłujących się przy piłce, kiedy wznawiana jest gra.



Rysunek 56. *Scrum na boisku*

W kontekście IT Scrum to kompletna metodyka adaptacyjna, która w szybkim tempie zyskuje popularność na świecie i w Polsce. Dzieje się tak, ponieważ — z jednej strony — Scrum daje konkretny i spójny przepis na prowadzenie projektów informatycznych, a z drugiej strony — pozostawia dużą swobodę dotyczącą sposobu implementacji w konkretnej organizacji. Dobra implementacja metodyki Scrum znacznie zwiększa wydajność zespołów wykonawczych i daje dużą elastyczność przy sposobie realizacji projektu. Należy jednak zaznaczyć, że metodyka ta dotyczy wyłącznie fazy wytwarzania i w żaden sposób nie opisuje prac związanych z przygotowaniem wizji projektu, podejmowaniem decyzji strategicznych czy zarządzaniem kontraktem.

Scrum, w porównaniu do XP, jest rozwiązaniem komplementarnym, ponieważ koncentruje się na procesie, a nie na dobrych praktykach. Wiele organizacji implementuje więc Scrum oraz wybrane praktyki XP.

Szczypta historii

Scrum jest bardzo młodą metodyką. Jego historia zaczyna się od artykułu *The New Product Development Game*, jaki ukazał się w Harvard Business Review na początku 1986. Jego autorami byli dwaj Japończycy, Takeuchi i Nonaka. Artykuł ten opisuje nowe podejście do realizacji nowych produktów w branży motoryzacyjnej, fotograficznej, komputerowej i drukarskiej. Japończycy opisują zasady funkcjonowania jednego, małego, interdyscyplinarnego zespołu, który realizuje swoje zadania w ramach wielu zachodzących na siebie faz. W artykule tym po raz pierwszy pojawia się odniesienie do gry w rugby, gdzie cały zespół „próbuje zdobyć teren, przekazując piłkę do tyłu i do przodu”.

O ile dwaj Japończycy uznawani są za inicjatorów tej metodyki, to jej ojcami można by nazwać Jeffa Sutherlanda i Kena Schwabera, którzy jako pierwsi zaczęli używać nazwy Scrum. W 1995 roku na konferencji OOPSLA w Austin zaprezentowali pierwszą kompletną wizję tej metodyki.

Ken Schwaber jest obecnie prezydentem organizacji Scrum Alliance, która zajmuje się popularyzacją tej metodyki na świecie. Oprócz publikacji artykułów i organizowania konferencji, organizacja ta zajmuje się również przeprowadzaniem certyfikowanych kursów Scrum.

Scrum jest obecnie używany nie tylko w małych organizacjach, ale również w dużych korporacjach, takich jak Fuji-Xerox, Canon, Honda, NEC, Epson, Brother, 3M, Xerox, Hewlett-Packard, IBM, Borland i Microsoft. W Polsce podejście to stosują m.in. ABG Spin i Google. Ciekawe podejście zastosowano w firmie Nokia Siemens, która do swojego polskiego oddziału zaprosiła Craiga Larmana¹ w celu przeprowadzenia serii szkoleń wewnętrznych. Warto również zauważyć, że w ostatnich latach Scrum Alliance zaczął organizować w Polsce otwarte, odpłatne szkolenia.

Role

Scrum definiuje w projekcie następujące role.

- ♦ Właściciel produktu — klient lub sponsor projektu, który jest odpowiedzialny za kwestie biznesowe.
- ♦ ScrumMaster — odpowiednik sędziego na boisku, czyli osoba odpowiedzialna za kontrolę przebiegu procesu.

¹ Craig Larman jest m.in. autorem książki *Agile and Iterative Development: A Manager's Guide* http://www.craiglarman.com/wiki/index.php?title=Main_Page

- ♦ Członek zespołu — wchodzi w skład małego, interdyscyplinarnego zespołu wykonawczego (5 – 9 osób).
- ♦ Interesanci — osoby odpowiedzialne za podejmowanie decyzji strategicznych. Za ich taktyczną implementację odpowiedzialny jest właściciel produktu. Interesanci mogą obserwować codzienne spotkania, ale nie powinni brać w nich aktywnego udziału. Dobrze obrazuje to rysunek 57.



Rysunek 57. Bajka Scrum o kurczaku i śwince

Morał z tej bajki jest taki, że należy jasno rozgraniczyć świnki, którym przydzielana jest praca, od kurczaków, które są zaangażowane, ale nie pracują bezpośrednio nad projektem. Podczas codziennych spotkań świnki są jedynymi uczestnikami, którzy mają prawo czynnego udziału.

Proces

Cały proces bazuje na szybkich i intensywnych cyklach nazwanych **sprintami**, które mogą trwać od dwóch do czterech tygodni. Ramy te nie są sztywne i w pewnych okolicznościach (np. prace utrzymaniowe) sprint może trwać jeden tydzień. Zdarzają się też sytuacje, w których sprint trwa dłużej, i tak w przypadku nowego produktu wytworzenie pierwszego prototypu może zająć dwa miesiące. Jeden sprint może następować bezpośrednio po drugim, ale nie musi. Jeżeli zespół zidentyfikował większą grupę dużych zadań i nie wie nic na temat sposobu ich realizacji, możliwa jest kilkudniowa przerwa przed rozpoczęciem planowania kolejnej iteracji. Kluczową kwestią jest, że w okresie sprinta wszyscy członkowie zespołu koncentrują się na realizacji dobrze określonej listy zadań.

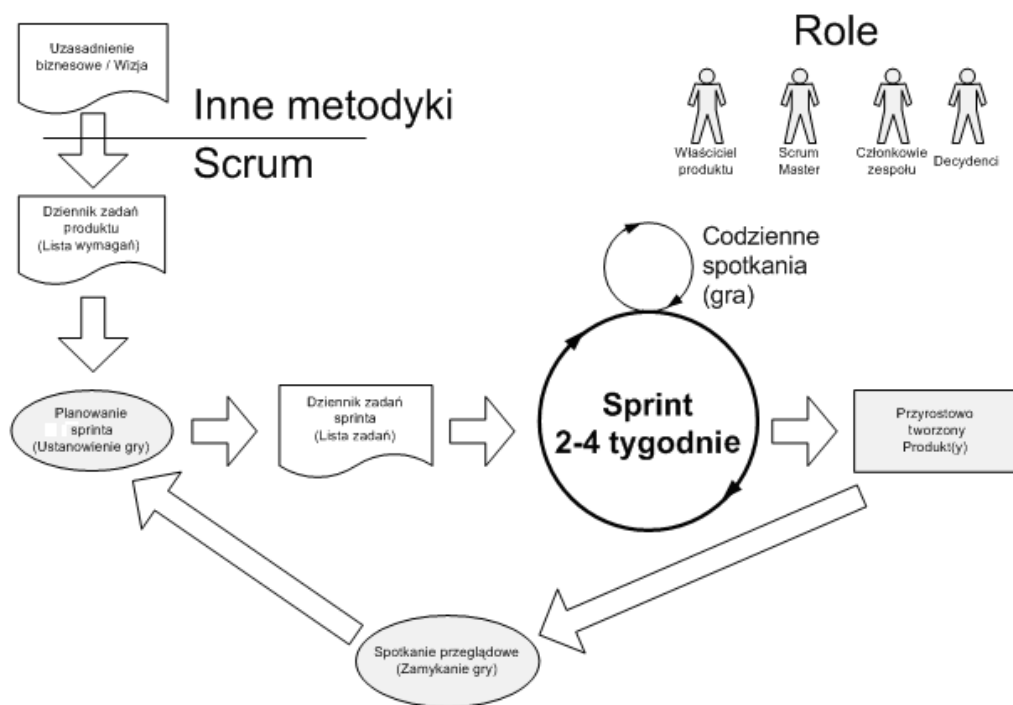
Kluczowym mechanizmem do rejestracji i rozliczania zadań jest **dziennik zadań** (ang. *back log*), a dokładniej dwa takie dzienniki.

- ♦ **Dziennik zadań produktu** (ang. *Product backlog*), który definiuje wysokopoziomowe, biznesowe funkcje aplikacji, jakie powinny zostać wykonane w trakcie całego projektu. Jest to podstawowa platforma dyskusyjna pomiędzy zespołem wykonawczym a właścicielem produktu — tabela 17 na stronie 210.

- ♦ **Dziennik zadań sprinta** (ang. *Sprint backlog*), który opisuje szczegółowe zadania techniczne w ramach danego sprinta. Optymalna długość trwania pojedynczego zadania w tym dzienniku to od czterech do dwunastu godzin, dla dwutygodniowego sprinta — tabela 18 na stronie 212.

Dzienniki te mogą przybrać bardzo różną postać, począwszy od arkusza Excela, na korkowej tablicy z karteczkami skończywszy. Zadania w tych dziennikach mogą być opisane przez różne parametry, takie jak wykonawca, priorytet biznesowy lub planowana i faktyczna długość trwania.

Całościowy obraz procesu wytwórczego Scrum przedstawiono na rysunku 58.



Rysunek 58. Architektura metodyki Scrum

Przed rozpoczęciem procesu wytwórczego powinniśmy mieć ustaloną wizję projektu. Byłoby najlepiej, gdyby powstał już jakiś dokument analityczny opisujący tę wizję. Jeżeli z pewnych powodów brakuje takiej wizji lub nie ma odpowiedzi na kluczowe pytania projektowe, istnieje możliwość uruchomienia sprinta zerowego. Jego celem jest ustalenie biznesowego zakresu projektu i technicznej wizji realizacji projektu. W ramach tego sprinta można przykładowo zbadać wykonalność najbardziej wrażliwych funkcjonalności.

Kiedy posiadamy już ogólną, biznesową wizję przyszłego produktu, najlepiej spisać ją w postaci „Dziennika zadań produktu”. Jeśli wykonany został jakiś dokument analityczny, warto powiązać zadania w dzienniku z funkcjonalnościami wymienionymi w dokumencie analitycznym (np. poprzez numery rozdziałów lub identyfikatory funkcjonalności

wymienione w dokumencie). Innym ciekawym rozwiązaniem jest powiązanie elementów z „Dziennika zadań produktu” z „Opowieściami użytkownika” stworzonymi w sposób zalecany przez XP.

Jeżeli mamy już ogólną, biznesową wizję projektu, jesteśmy gotowi do uruchomienia całej mechaniki Scrum, która składa się z trzech głównych etapów.

♦ **Ustanowienie gry** (ang. *pregame*) zajmuje jeden, pełny dzień.

- ♦ Planowanie — pierwsza połowa dnia (cztery godziny) jest poświęcona na zdefiniowanie, które funkcjonalności z „Dziennika zadań produktu” zostaną zrealizowane w kolejnym sprincie. Innymi słowy, wynikiem tego spotkania powinno być ustalenie biznesowego zakresu „Dziennika zadań sprinta”. Oznacza to, że zespół wykonawczy musi zapoznać się z bieżącym „Dziennikiem zadań produktu”. Następnie właściciel produktu przedstawia priorytety poszczególnych zadań. Zadania o najwyższym priorytecie powinny zostać wykonane najpierw. Zespół wykonawczy deklaruje wstępnie, które z zadań może wykonać w kolejnym sprincie, ustalając kolejność zadań od najważniejszych do najmniej ważnych. Końcowym efektem tego spotkania powinno być jasne określenie celu sprinta.
- ♦ Architektura — druga połowa dnia (cztery godziny) jest poświęcona na kwestie techniczne i krótką dyskusję na temat technicznego wykonania każdej ze zdefiniowanych funkcjonalności. To spotkanie odbywa się już wewnątrz zespołu wykonawczego, bez udziału właściciela produktu. W ramach tego spotkania weryfikowane są czasy wykonania poszczególnych zadań i zakres sprinta może ulec pewnym, dodatkowym zmianom.

Pierwszy dzień kończy się *automatycznie* uruchomieniem sprinta.

- ♦ **Gra** (ang. *game*) to okres wykonania projektu. W ramach Scrum należy zaplanować *codzienne* 10 – 30 minutowe spotkania prowadzone przez ScrumMastera. Scrum nie definiuje, w jakiej części dnia powinny się odbywać, ale warto z góry założyć pewną stałą, konkretną godzinę. Spotkania należy przeprowadzić na stojąco (ang. *Stand-up meeting*), aby dodatkowo zmotywować uczestników do szybkiego ich przeprowadzenia. W trakcie każdy z członków zespołu ma odpowiedzieć wyłącznie na trzy pytania:

- ♦ Co zostało wykonane od ostatniego spotkania?
- ♦ Co będzie wykonane do następnego spotkania?
- ♦ Czy istnieje cokolwiek, co może zagrażać wykonaniu planów?

Gdy w trakcie spotkania pojawia się nowy wątek lub niskopoziomowa dyskusja, rolę ScrumMastera jest interweniować i przenieść tę rozmowę na inny termin, w innym składzie.

Codzienne spotkania służą również weryfikacji stopnia realizacji projektu. Scrum posiada tutaj bardzo oryginalną technikę **diagramu spalania** (ang. *burndown chart*). Polega ona na tym, że po zaplanowaniu sprinta wyliczana jest liczba godzin (dni) pracy. Następnie przeprowadza się krótką symulację, jak z dnia na dzień liczba tych godzin (dni) będzie ulegać zmniejszeniu. W trakcie realizacji

projektu codziennie weryfikuje się, jak rzeczywistość ma się do pierwotnych planów. Przykład wykresu spalania został zaprezentowany na rysunku 60. na stronie 227.

W większych projektach można pokusić się o wykonanie diagramu spalania dla całego projektu. W takim przypadku „Dziennik zadań produktu” musi zawierać szacunki wykonania wszystkich funkcjonalności biznesowych i należy przeprowadzić symulację ich realizacji w poszczególnych sprintach.

Najlepszą techniką jest tutaj szczegółowe oszacowanie pierwszego sprinta, a dopiero potem zaplanowanie ogólnie, które funkcje będą zrealizowane w kolejnych sprintach. Po tak wykonanych szacunkach można już narysować przewidywany przebieg spalania całego projektu. Następnie, po zakończeniu każdego ze sprintów należy monitorować, jaka część wstępnie zaplanowanej pracy została faktycznie wykonana, jak dużo pojawiło się nowych zadań i czy końcowy termin realizacji projektu jest nadal realny.

W trakcie realizacji sprinta cele i wizja projektu mogą ulec zmianom, ale one nie powinny mieć wpływu na prace bieżące. Dopiero w trakcie kolejnej fazy planowania można nanieść zmiany w „Dzienniku zadań produktu” i zaplanować kolejny sprint. W „sytuacji wyjątkowej” można skrócić czas trwania sprinta przy założeniu redukcji funkcjonalności.

Niektóre zespoły Scrum, szacując czas wykonania zadania, bazują na punktach abstrakcyjnych, a nie rzeczywistych czasach wykonania. Jeżeli np. szacujemy, że wykonanie funkcjonalności ekranu logowania zajmie nam 1, ile zajmie wykonanie formatki do wprowadzania danych? Technika ta bazuje na wartościach z ciągu Fibonacciego i tak poszczególne zadania mogą przyjąć wartości: 1, 2, 3, 5, 8, 13, 21, 34, 55. Technika zakłada, że ten sposób szacowania bardziej przystaje do sposobu działania ludzkiego umysłu, który właśnie tak wykonuje szacowanie. Przykładowo zadania o wielkości 12 i 15 mają w praktyce identyczną wielkość (są również abstrakcyjne). Dodatkowo ludzki umysł, wykonując szacunki, aktywnie wykorzystuje technikę porównywania poszczególnych wartości. Technika ta motywuje również do dzielenia dużych zadań na mniejsze. Kluczem do jej efektywnego wykorzystania jest zrozumienie, że punkty abstrakcyjne w żaden sposób nie odpowiadają rzeczywistemu czasowi wykonania. To, jak wiele zadań zostanie zaplanowanych w ramach sprinta, zależy tylko i wyłącznie od zespołu.

- ♦ **Zamykanie gry** (ang. *postgame*) to zamknięcie sprinta, które zajmuje jeden pełny dzień.
- ♦ **Spotkanie przeglądowe** (ang. *Review meeting*) — pierwsza połowa dnia (cztery godziny) — jest poświęcone na prezentację wykonanych prac właścicielowi produktu oraz interesantom. Wykonanie poszczególnych zadań w Scrum oznacza ich gotowość do prezentacji, czyli pełnego przetestowania. Oznacza to również, że po zakończeniu ostatniego dnia sprinta wykonany produkt powinien być „prezentowalny” bez dodatkowych czynności. W trakcie planowania sprinta należy więc wziąć tę kwestię pod uwagę i zaplanować zadania związane z integracją końcowego rozwiązania.

- ♦ Spotkanie retrospekcyjne (ang. *Postmortem meeting*) — druga połowa dnia (cztery godziny) — jest poświęcone na wewnętrzną dyskusję w zespole na temat tego, jak przebiegał ostatni sprint. W trakcie spotkania należy szczególnie przeanalizować, co się udało, co się nie udało, czego się nauczyliśmy, co nas martwi. Spotkanie to ma na celu doskonalenie sposobu wykorzystania metodyki Scrum oraz uświadomienie członkom zespołu wykonawczego, że to oni sami są właścicielem całego procesu. Jeżeli w organizacji istnieją inne zespoły realizujące projekty według Scrum, warto zaprosić ich przedstawiciela na takie spotkanie, ponieważ dzięki temu organizacja może naturalnie budować wewnętrzne standardy.

Wymienione powyżej czasy trwania poszczególnych etapów są przykładowe i w zależności od sytuacji mogą ulec pewnemu zwiększeniu lub zmniejszeniu. Ustanowienie i zamykanie gry nie powinno jednak trwać dłużej niż jeden dzień.

W trakcie wykorzystania metodyki Scrum należy zwrócić uwagę na kilka specyficznych kwestii praktycznych.

- ♦ Scrum ma na stałe ustaloną długość i bez względu na przebieg wypadków wersja rozwiązania musi powstać do ustalonej daty nawet wtedy, jeśli powoduje to redukcję wstępnie planowanej funkcjonalności. Data realizacji sprinta jest pojmowana śmiertelnie poważnie (ang. *dead line*).
- ♦ Nie jest to powiedziane wprost, ale Scrum zakłada zarzucenie klasycznych narzędzi planistycznych, takich jak MS Project i wykresy Gantta. W takim przypadku należy manualnie uwzględnić takie kwestie jak urlopy pracowników lub święta narodowe.
- ♦ Warto zastanowić się nad mechanizmami doskonalenia szacunków wykonania zadań w kontekście całego zespołu i poszczególnych pracowników. Scrum proponuje tutaj ciekawą technikę szybkości zespołu (ang. *velocity*), która polega na monitorowaniu, jak wiele pracy zespół jest w stanie zrealizować w trakcie jednego sprinta. Wartość ta jest zapamiętywana po każdym ze sprintów, co pozwala lepiej planować kolejne sprinty.
- ♦ Wykonując szacunki, nieświadomie posługujemy się pojęciem dnia idealnego. W praktyce na realizację zadań projektowych pracownicy są w stanie poświęcić średnio pięć godzin z ośmiu. Pozostały czas to kawa, e-maile i różne zadania administracyjne. Warto tę kwestię wziąć pod uwagę, szczególnie przy pierwszych szacunkach.
- ♦ Zespołowi, który korzysta ze Scrum po raz pierwszy, warto najpierw przedstawić, na czym cała metodyka polega. Średnio konieczne są 3 – 4 sprinty, zanim Scrum zostanie w pełni wdrożony i zespół zacznie realizować zadania w sposób przewidywalny.
- ♦ Scrum nie zajmuje się zarządzaniem ryzykiem, ale przykład zawarty w rozdziale „Przykład — Scrum BlogSerwis” przypisuje pewne ryzyko do każdego z zadań i na tej podstawie modyfikowany jest finalny szacunek kosztu realizacji.

- ♦ Zadania, które zawiera Scrum, to nie tylko zadania związane wprost z realizacją funkcjonalności, ale wszystkie zadania związane z realizacją celu sprinta. Oznacza to, że planując sprint, nie można zapomnieć o testowaniu, przygotowaniu dokumentacji czy wykonaniu instalatorów, ponieważ tylko wtedy możliwe jest dostarczenie funkcjonalności biznesowej.
- ♦ „Adaptacyjność” metodyki nie zwalnia z konieczności dużej samodyscypliny. O ile pierwszy sprint będzie wiązał się z dużą ekscytacją w całym zespole, o tyle w kolejnych łatwo „zapomnieć” o pewnych technikach, takich jak codzienne spotkania. Z jednej strony, królewską zasadą są praktyczne praktyki (ang. *rule of thumb*), ale z drugiej strony, mamy grzech zaniechania, który może spowodować chaotyczną realizację projektu. ScrumMaster jest odpowiedzialny za właściwe wyważanie, które praktyki muszą być stosowane, a które mogą zostać zaniechane.
- ♦ W organizacji, która aktywnie wykorzystuje Scrum przy więcej niż jednym projekcie, warto zastanowić się nad wdrożeniem techniki Scruma Scrumów (ang. *Scrum of Scrums*). Polega ona na tym, że kluczowe osoby odpowiedzialne za realizację poszczególnych sprintów (ScrumMaster lub Właściciel produktu) spotykają się codziennie na własnych spotkaniach i omawiają na wyższym poziomie ogólności:
 - ♦ Co zostało wykonane w projekcie od ostatniego spotkania?
 - ♦ Co będzie wykonane w projekcie do następnego spotkania?
 - ♦ Czy istnieje cokolwiek, co może zagrażać wykonaniu planów?

Takie podejście służy zwiększeniu transparentności sposobu realizacji projektu oraz pełnego współdzielenia wiedzy w obrębie organizacji.

Podsumowanie

W praktyce główną cechą charakterystyczną metodyki Scrum jest duży stopień nieformalności realizowanych zadań i koncentracja na efektach. Z jednej strony, oznaczać to może dużą produktywność dobrze prowadzonego zespołu, ale z drugiej strony, oznacza również dużą odpowiedzialność. Błędne działania i decyzje ScrumMastera mogą powodować chaotyczną realizację projektu, co w efekcie może spowodować przekroczenie terminów lub wytworzenie produktu o złej jakości. W związku z tym, warunkiem kluczowym do dobrego działania metodyki jest stałe podtrzymanie wysokiej motywacji całego zespołu oraz bieżąca komunikacja z klientami końcowymi.

Jak wszystko, również i metodyki adaptacyjne można zastosować dobrze lub źle. Dobrze użyta metodyka może dać nadzwyczajne efekty, ale bardzo często mogą one ulec wypaczeniu. Oto, na co należy zwrócić szczególną uwagę.

- ♦ Jakość — metodyki adaptacyjne mówią o automatyzacji testów, ale konieczne jest również zapewnienie właściwej weryfikacji wykonanej funkcjonalności biznesowej (testy akceptacyjne).

- ♦ Dokumentacja — często słyszy się, że metodyki adaptacyjne oznaczają brak konieczności tworzenia standardowej dokumentacji, ponieważ aplikacja „opisuje się sama”. Jeżeli użytkownicy kluczowi zrezygnują z wykonania dokumentacji użytkownika, administratora i dokumentacji technicznej, faktycznie można przyjąć takie założenie. Należy jednak przedyskutować takie decyzje w trakcie spotkania planistycznego, aby uniknąć niemiłych niespodzianek.
- ♦ Kompletność — wykonane zadanie nie jest zakończeniem prac wykonawczych, jest nim w pełni działająca funkcjonalność. Zespół wykonawczy jest odpowiedzialny za wykonywane zadania i jego celem jest dostarczenie produkcyjnej wersji systemu, a nie półproduktu.
- ♦ Zarządzanie ryzykiem — Scrum nie definiuje wprost mechanizmów zarządzania ryzykiem, z wyjątkiem pytania: „Co może pójść źle?”. Nie zawsze jest to działanie wystarczające i warto zastanowić się nad mechanizmami oceny ryzyka poszczególnych zadań lub modułów.

O ile XP jest bardzo ciekawym zbiorem dobrych praktyk, to Scrum jest w pełni dojrzałym modelem wytwarzania produktów bazującym na dobrze opisanym procesie. Dlatego właśnie najbardziej popularnym podejściem jest wdrożenie Scrum z wykorzystaniem wybranych praktyk XP.

Scrum i XP na pewno nie zrobią wszystkiego, ale spowodują skrócenie łańcucha decyzyjnego.

Nie rozwiążą wszystkich problemów, ale ułatwią ich rozwiązywanie dzięki większemu zaangażowaniu ludzi.

Nie zapewnią spełnienia wszystkich oczekiwań klientów, ale spowodują, że będą bardziej wymagający ☺.

Rozmowa z...

Bartosz Kiepuszewski jest certyfikowanym praktykiem Scrum (CSP), starszym konsultantem Cutter Consortium i doradcą zarządu Business Management Software, polskiej firmy IT specjalizującej się w budowaniu zaawansowanych rozwiązań informatycznych dla sektora finansowego. Dr Kiepuszewski specjalizuje się w adaptacyjnym zarządzaniu projektami informatycznymi oraz budowie i zarządzaniu architekturą korporacyjnych systemów IT. Pracę doktorską obronił na Queensland University of Technology, gdzie pracował nad teoretycznymi aspektami modelowania procesów biznesowych w systemach workflow. Przed rozpoczęciem współpracy z Business Management Software pracował przez pięć lat w Australii, najpierw w Distributed Systems Technology Centre, potem w Mincom LTD jako starszy architekt systemowy. Po powrocie do Polski w 2001 roku podjął pracę w Infovide S.A., gdzie zarządzał zespołem tworzącym aplikacje w architekturze J2EE, a następnie współzarządzał zespołem konsultingowym doradzającym klientom w obszarze strategicznego zarządzania IT.

Wujek Dobra Rada — odcinek 10. „Lekko, łatwo i przyjemnie?”

METODYKI ADAPTACYJNE TO NIE PANACEUM NA WSZELKIE ZŁO,
ALE ODPOWIEDZIALNY ZESPÓŁ WYKONAWCZY MOŻE ZROBIĆ WIĘCEJ ZA MNIEJ!

Adam Koszłajda (AKO): Bartku, jesteś osobą, która już od dłuższego czasu propaguje temat Scrum na polskim rynku. Dlaczego akurat ta metodyka?

Bartosz Kiepuszewski (BKI): Osobiście nie określiłbym się jako zwolennik jednej, konkretnej metodyki — w każdym projekcie używam elementów Scrum, FDD, XP, APM.

AKO: Dlaczego więc styl adaptacyjny, a nie tradycyjny?

BKI: Pracowaliśmy nad nowym, innowacyjnym produktem, który miał być szybko wdrożony na rynek, mimo dużej niepewności zarówno w obszarze funkcjonalności, jak i technologii wykonania.

- AKO:** W takich przypadkach rzeczywiście bezcenna może okazać się wysoka motywacja całego zespołu, który sam się organizuje. Wiele osób preferuje jednak tradycyjne metody zarządzania projektami, ponieważ bardzo obawia się porzucenia swojej kierowniczej roli na rzecz demokratyzacji podejmowanych decyzji. Gdzie kota nie ma, tam myszy harcują.
- BKI:** W metodyce Scrum rolę kota pełni poniekąd ScrumMaster, a w doświadczonym zespole myszy zaczynają pilnować się nawzajem. To się ładnie nazywa „poczuciem odpowiedzialności”, które trzeba w zespole pielęgnować.
- AKO:** I doskonalić.
- BKI:** I wzmacniać. Ważne jest zrozumienie tego, że w metodykach adaptacyjnych tradycyjny formalizm jest zastąpiony przez dużą dyscyplinę, a ta nie bierze się znikąd. Co więcej, może wygasać. Na to trzeba bardzo uważać.
- AKO:** Często pojawia się jednak wymaganie wykorzystania konkretnej metodyki po stronie dostawcy, takiej jak Prince2.
- BKI:** Pytanie jednak, co to ma oznaczać. Jeżeli zamawiający ma wdrożone Prince2 w sposób poprawny, zgodny z metodyką, to ja, jako dostawca, w zasadzie nie muszę tego wiedzieć. Przecież w Prince2 nic nie mówi się na temat procesu wytwórczego u dostawcy lub głównego wykonawcy.
- AKO:** Fakt... Prince2 to bardziej ramy zarządcze. Zawiera specjalną grupę procesów „Zarządzanie Wytwarzaniem Produktów” dedykowaną do przekazywania i odbierania konkretnych pakietów prac, które mogą być wykonywane w dowolny sposób. W metodykach adaptacyjnych widzę jednak dość poważną lukę, która uniemożliwia weryfikowanie dokładności pierwotnych planów i efektywności realizacji projektu.
- BKI:** Pamiętajmy, że zmorą typowego projektu informatycznego jest to, iż postęp prac dochodzi do 80 – 90%, a potem „magicznie” utyka. Podejście Scrum w założeniach ma wyeliminować ten problem i proponuje diagramy spalania (ang. *burndown chart*), które w moim odczuciu są bardziej wiarygodne. Z punktu widzenia tej metodyki mierzenie dokładności planów nie jest specjalnie ciekawe, ponieważ plany są jedynie spekulacją. W praktyce liczy się bardziej to, kto szybciej i taniej od konkurencji zbuduje bardziej innowacyjny produkt lub usługę. A co do efektywności, jej mierzenie w IT przy często zmieniającej się technologii jest rzeczywiście bardzo złożonym zagadnieniem.
- AKO:** Generalnie większość źródeł dowodzi, że właściwie zastosowane metodyki adaptacyjne powodują szybsze wygenerowanie aplikacji...
- BKI:** To, o czym mówisz, to faktyczne badania i dane z rynku. Rzeczywista skuteczność metodok adaptacyjnych jest empirycznie stwierdzonym faktem popartym badaniami ☺. Warto zaznaczyć, że wielu często ugruntowanych stwierdzeń na temat zarządzania nie poparto żadnymi badaniami... Czy widziałeś np. badania na temat skuteczności Prince2?

AKO: W Prince2 i w innych tego typu metodykach głównym celem nie jest efektywność, ale udokumentowanie każdego ważnego punktu decyzyjnego. Wydaje mi się wręcz, że o ile metodyki adaptacyjne stawiają na szybkie wytworzenie „czegoś”, o tyle Prince2 stawia na formalną poprawność.

BKI: Czyli powtarzalność i audytowalność, tak? To ma niewiele wspólnego ze skutecznością.

AKO: Ciekawe, jak wyglądałby audyt projektu prowadzonego w Scrum?

BKI: Oczywiście, że w idealnym świecie chciałoby się mieć i skuteczność, i powtarzalność, i pełną audytowalność. Problem polega na tym, że przy ograniczonych zasobach czegoś się zazwyczaj pozbywamy. Niestety, w wielu projektach zostaje formalizm i audytowalność, a skuteczność schodzi na dalszy plan. Takie projekty regularnie się opóźniają i nie wytwarzają wysokiej jakości produktów o dużej wartości dla biznesu. No ale najpierw zdefiniujmy pojęcie audytu.

AKO: Wydaje mi się, że przy audytach kluczową sprawą jest możliwość weryfikacji podejmowanych decyzji, sposobu i pobudek ich podejmowania oraz możliwość realnej weryfikacji postępu prowadzonych prac. Zazwyczaj nic nie idzie zgodnie z planem i audyt powinien odpowiedzieć, czy finalnie można było uzyskać dużo więcej, czy rezultat prac jest satysfakcjonujący. Jeżeli można było osiągnąć dużo więcej, to w jaki sposób, po podjęciu jakich decyzji i kiedy?

BKI: Scrum nie wypowiada się na temat tego, czy decyzje powinny być formalnie notowane, ale nie widzę przeszkód, aby je formalnie zapisywać. A co do realnej weryfikacji postępu prac, uważam, że w tym tkwi największa zaleta każdej metodyki adaptacyjnej, ponieważ postęp jest dużo realniej mierzony, a efektywność pracy zespołu jest widoczna już po pierwszej iteracji... Powtórzę za Kenem Schwabermem, że Scrum świetnie się sprawdza zarówno w doświadczonych zespołach, jak i w zespołach składających się z bardzo niedoświadczonych nowicjuszy, którzy nie potrafią nic sensownie zrobić. I w jednym, i w drugim przypadku już po pierwszej iteracji będziemy w stanie ocenić wydajność zespołu i jakość prac. Właśnie ta transparentność działań jest w metodyce Scrum kluczowa. Jeżeli jednak zgodność formalna jest ważniejsza od efektywności pracy, rzeczywiście adaptacyjne metodyki nie są najlepszym wyborem :).

AKO: Założmy jednak, że mamy do czynienia ze średnio inteligentnym zespołem programistów, który zafascynował się metodyką Scrum. Od czego mieliby zacząć?

BKI: Uważam, że najpierw zespół powinien wdrożyć narzędzia i praktyki inżynierskie umożliwiające minimalizację kosztu zmian, takie jak test-driven development, continuous integration itp.

AKO: Czyli przejrzeć prace Joela Spolsky'ego?

BKI: Na przykład. A jeszcze lepiej praktyki zdefiniowane w XP. A potem jest już łatwo... Zamiast tradycyjnie planować, analizować i projektować cały system, pracujemy nad jego fragmentem. Jak ten fragment systemu skończymy, zabieramy

się za kolejny. Do tego — oczywiście — musimy wdrożyć praktyki zarządcze mające na celu usprawnienie komunikacji i stworzenie właściwych warunków do innowacyjnej oraz skutecznej pracy. Zatem, pozbywając się formalizmu, budujemy zdyscyplinowany i samoorganizujący się zespół. Jak o tym zapomnimy, możemy — niestety — z formalnego procesu wpaść w chaos i anarchię. Scrum jest w tym sensie metodyką zarządczą, a nie wytwórczą.

AKO: Ok... Zalecasz więc de facto wprowadzenie najpierw pewnych konkretnych technik, których Scrum wprost nie proponuje.

BKI: Wiele zespołów używa Scrum + XP.

AKO: Z programowaniem w parach?

BKI: Tak. Badania pokazują, że programowanie w parach jest bardzo skuteczne. Pytanie: „Wprowadzić programowanie w parach, czy nie?” jest jednak pytaniem źle zadany. Powinniśmy się zastanowić, jak zapewnić dobry przepływ informacji między członkami zespołu, sprawdzenie kodu innej osoby przez drugą, zapewnienie tego, że w przypadku choroby czy nieobecności któregoś z programistów prace nad danym modułem nie stają, kontrolę utrzymywania zgodności kodu z przyjętą architekturą i standardami itp. XP na to wszystko proponuje programowanie w parach. Możemy stosować inne techniki — szybko jednak dojdziemy do wniosku, że większość z nich jest droższa we wdrożeniu i mniej skuteczna.

AKO: Do mnie bardziej przemawia technika „ramię w ramię”, czyli dwóch programistów siedzących obok siebie przy własnych komputerach. Ciekawe badanie zostało wykonane przez zespół doktora Nawrockiego z Politechniki Poznańskiej.

BKI: Nie znam tego badania, ale jest dużo badań wykonanych w USA, które pokazują wysoką skuteczność programowania w parach. Ale — oczywiście — nie ma problemu ze stosowaniem rozmaitych nowych technik. Taka jest zresztą idea adaptacyjności — nasz proces wytwórczy też powinien być adaptowalny, a niekoniecznie powtarzalny.

AKO: Technika „ramię w ramię” jest bardzo podobna do programowania w parach, z tą minimalną różnicą, że programiści posiadają własne maszyny. Dzięki temu są „zsynchronizowani”, ale zawsze mają też możliwość wykonywania oddzielnie swojej pracy i się nie nudzą. Słyszałem plotkę o przypadku, kiedy jeden z programistów spał, podczas gdy drugi pisał kod ;).

BKI: To świadczy o kompletnym braku odpowiedzialności obu programistów :). Ale, tak jak mówiłem wcześniej, poczucie odpowiedzialności to fundament, na którym możemy budować kolejne aspekty metodyk adaptacyjnych.

AKO: No przecież XP nie zabrania spania :D.

BKI: Spania? Zabrania!

AKO: Ja tego punktu nie znalazłem.

- BKI:** Programowanie w parach oznacza przede wszystkim, że żadna decyzja programistyczna nie jest podejmowana samodzielnie. Spanie trochę to wyklucza :). Pair Programming to nie jest po prostu posadzenie dwóch osób przy jednym komputerze :).
- AKO:** Wracając do głównego wątku... Jakie są inne kluczowe elementy wdrożenia metodyki Scrum?
- BKI:** Zazwyczaj wdrażanie w organizacji zaczyna się od krótkiego szkolenia wszystkich zaangażowanych. Następnie warto się zastanowić, kto powinien pełnić rolę ScrumMastera i ProductOwnera — szczególnie tradycyjny kierownik projektu nie zawsze jest najlepszym kandydatem do roli ScrumMastera. Dobrze, jeżeli rolę tę obejmuje osoba, której zadaniem jest krzewienie kultury scrumowej w firmie.
- AKO:** Czyli zakładasz, że kierownik projektu z podejścia tradycyjnego nie może „zreformować się” do ScrumMastera?
- BKI:** Chodzi o to, że zgodnie z metodyką ScrumMaster nie angażuje się w prace zespołu, a już na pewno nie podejmuje decyzji projektowych — zupełnie odwrotnie do kierownika projektu, od którego oczekuje się przede wszystkim podejmowania wielu decyzji w projekcie. Jeśli z kolei ProductOwnerem nie może być przedstawiciel klienta, co jest sytuacją idealną, może nim zostać główny analityk pełniący rolę „proxy” klienta, czyli reprezentujący klienta wobec zespołu. Tradycyjnemu kierownikowi projektu zostaje nadzór nad kontraktem. Znam firmy, w których po wdrożeniu metodyki Scrum „znikli” kierownicy projektów, bo rodziło to wątpliwości co do ich roli, a zamiast tego zostali powołani „Contract Managers”.
- AKO:** A jakie są inne błędy najczęściej popełniane przy wykorzystaniu metodyki Scrum?
- BKI:** Lista błędów jest długa :).
- ♦ Brak zaangażowania klienta (robimy oprogramowanie sami bez jego wiedzy).
Efekt: nie ma feedbacku po każdej iteracji i nie ma szans na adaptację produktu, więc korzyści z wdrożenia metodyki są niewielkie.
 - ♦ Brak zdyscyplinowanego stosowania technik umożliwiających współdziałanie (codienne spotkania, programowanie w parach itp.) przy jednoczesnym pozbyciu się tradycyjnych, formalnych mechanizmów (specyfikacje, sign-offy, formalne przeglądy).
Efekt: dramatyczny spadek jakości z jednoczesną utratą wszelkiej dokumentacji.
 - ♦ Brak dyscypliny w utrzymaniu stałego czasu trwania iteracji (mamy skończyć iterację w piątek, ale zostało niewiele, więc skończymy w następnym wtorek, może środę).
Efekt: spada wiarygodność raportowania postępu prac, zaburza się rytm pracy zespołu, rozluźnia dyscyplina.

- ♦ Przy kierowniku projektu, który „reformuje się” do ScrumMastera, może zaistnieć kontynuacja mikrozarządzania przez kierownika projektu, czyli zlecanie pojedynczych zadań ze ścisłą kontrolą terminu ich wykonania.

Efekt: zrujnowanie koncepcji samoorganizacji i samodyscypliny, „dylbertyzacja”, wyrobienie fatalnego nawyku „skoro kierownik projektu kazał, zrobię to, mimo iż sędzę, że jest to bez sensu”.

Mógłbym tak długo ;).

AKO: Kilka razy pojawiło się tutaj słowo dyscyplina. Wiele osób postrzega metodyki adaptacyjne jako coś, co zwolni ich z obowiązku pisania dokumentacji i da możliwość szybkiej reakcji na zmieniające się otoczenie biznesowe, ale nie dostrzega, że metodyka — oprócz tych korzyści — narzuca też pewne obowiązki. Być może właśnie dlatego często zdarza się, że po początkowej euforii na końcu mamy do czynienia z powrotem do starych praktyk... Mimo to, wydaje mi się, że można wprowadzić metodyki adaptacyjne „tylnymi drzwiami”. Firma może niejako oddolnie ulec zmianie, jeśli tylko mamy do czynienia z grupą osób, która zmierza w tym właśnie kierunku. Co o tym myślisz?

BKI: Byłbym ostrożny w takim podejściu — jeżeli zespół chce pracować adaptacyjnie, a np. biuro projektów wymaga od kierownika projektu składania szczegółowych Ganttów, liczenia Earned Value itd., niestety — szybko pojawi się tendencja budowania WBS zamiast podziału funkcjonalności na iteracje i kontrolowania stanu wykonania zadań, a nie realizacji poszczególnych elementów funkcjonalnych. Co więcej, tradycyjny kontroling nie lubi, jak zmieniają się plany, więc naturalne będzie blokowanie wszelkiej elastyczności zakresu. No i ostatecznie brak wsparcia „z góry” może szybko doprowadzić do frustracji i wygaszenia zapału zespołu. Dobrym przykładem jest tu ISO bądź CMMI; jeżeli firma jest na trzecim poziomie CMMI i jest wymagane, aby w każdym projekcie informatycznym najpierw powstała szczegółowa specyfikacja zakresu zgodna z szablonem, zawierająca z góry zdefiniowane artefakty (np. przypadki użycia zgodne z przyjętym szablonem), a dopiero zatwierdzenie tej specyfikacji umożliwia rozpoczęcie prac projektowych (nie mówiąc o programowaniu), wdrożenie metodyki Scrum się nie uda.

AKO: Sugerujesz więc, że w dużych firmach Scrum raczej nie ma racji bytu jako filozofia działania całej organizacji? Może tam być wykorzystywany głównie do „operacji specjalnych”?

BKI: Nie, sugeruję, że warto zadbać o wsparcie „z góry”, ale próba oddolnego wprowadzenia Scrum w firmie, która ma np. ISO i CMMI, może się nie udać. ISO nie jest jednak problemem samo w sobie. Jak napiszemy w ISO, że używamy Scrum, to super. Gorzej, jak jest napisane w ISO, że ma być proces kaskadowy (a najczęściej tak właśnie jest). Podobnie jest zresztą z CMMI. Znam firmę amerykańską, która jest na poziomie trzecim CMMI i używa metodyki Scrum. Żartowali, że dopiero trzeci audytor zrozumiał, iż można zarządzać wymaganiami inaczej, niż w tradycyjny sposób.

Pomimo że CMMI nie narzuca konkretnych technik (mowa jest, podobnie jak w ISO, o zdefiniowanym procesie), powszechnie utożsamia się „zarządzanie wymaganiami” z istnieniem dokumentu o nazwie „specyfikacja wymagań”, zbiera się rozmaite metryki, takie jak długość tego dokumentu, czas i koszt związany z jego powstaniem itp. Tymczasem w Scrum i wszystkich innych metodykach adaptacyjnych bardzo uważnie i świadomie zarządza się wymaganiami, tyle tylko, że inaczej, a nie tradycyjnie.

AKO: Najważniejsze na końcu jest wszak zadowolenie klienta i częste, przyrostowe dostarczanie mu funkcjonalności najlepiej synchronizuje wizję po obu stronach (wykonawcy i odbiorcy).

BKI: Dokładnie tak.

AKO: Poza tym metodyki adaptacyjne są bardziej sexy? ;)

BKI: Takie aspekty nie mają żadnego praktycznego znaczenia, choć — oczywiście — bycie sexy jest ważne ;).

Rozdział 8.

Joel o oprogramowaniu

W ostatnim okresie w kontekście metodyk adaptacyjnych bardzo często, pojawia się postać Joela Spolsky'ego, który jest powszechnie uznawanym autorytetem w obszarze małych firm wytwarzających oprogramowanie (ang. *Micro Independent Software Vendors*). Joel jest właścicielem małej, nowojorskiej firmy Fog Creek Software. Prowadzi również powszechnie znany blog o nazwie „Joel on Software”. Jego opracowania są bardzo często dalekie od standardowego sposobu myślenia; Joel za pomocą swojej kryształowej kuli może np. przepowiedzieć wielkość dochodów w pierwszym miesiącu życia małej, nowo powstałej firmy programistycznej. Wbrew pozorom nie będzie to ani 0 ani 50 tys. dolarów, tylko ni mniej, ni więcej ... 364 dolary¹ ☺.

Blog „Joel on Software” nie propaguje żadnej konkretnej metodyki, ale opublikowane tam artykuły podają szereg recept do szybkiego i praktycznego zastosowania we wszystkich organizacjach. Warto podkreślić, że wiele z nich doczekało się już tłumaczenia na język polski w postaci książek i artykułów. Jednym z najciekawszych opracowań jest „Test Joela”², który opisuje dwanaście głównych pytań do członków zespołu programistycznego:

1. Czy stosujesz system kontroli wersji?

Czy twoja firma korzysta z repozytorium kodu, które umożliwia przechowywanie kolejnych wersji oprogramowania, takich jak VSS, CVS lub SVN?

2. Czy możesz zbudować wersję w jednym kroku?

Ile kroków jest potrzebnych do tego, aby od oprogramowania przejść do pliku wykonywalnego lub działającej wersji rozwiązania?

3. Czy stosujesz codzienne budowanie wersji?

Czy codziennie budowana jest nowa wersja rozwiązania na bazie zmienianego kodu?

¹ <http://www.joelonsoftware.com/articles/Micro-ISV.html>

² „Test Joela: 12 kroków ku lepszym programom” (ang. *The Joel Test: 12 Steps to Better Code*).

4. Czy używasz systemu zarządzania błędami?

Czy wszystkie błędy w oprogramowaniu są odnotowywane w centralnym rejestrze (ang. *Bug Tracking System*)?

5. Czy usuwasz błędy, zanim napiszesz nowy kod?

Czy zanim stworzysz nową wersję rozwiązania, usuwasz wszystkie błędy z wersji poprzedniej?

6. Czy masz harmonogram aktualizowany na bieżąco?

Czy wykonując swoją pracę, wiesz o tym, jaki jest bieżący harmonogram całego projektu?

7. Czy masz specyfikację?

Czy wykonując swoją pracę, masz dokument opisujący całe rozwiązanie?

8. Czy programiści mają komfortowe warunki pracy?

Czy masz odpowiednią przestrzeń, spokój i prywatność, które umożliwiają ci wprowadzenie się w „trans produktywności”?

9. Czy używasz najlepszych dostępnych narzędzi?

Czy masz odpowiednio mocny komputer, dwa monitory (jeżeli tak trzeba) i oprogramowanie, które powoduje, że twoja produktywność zależy głównie od twoich umiejętności?

10. Czy masz testerów?

Czy w twojej organizacji powołano osoby dedykowane do wykonania testów oprogramowania?

11. Czy kandydaci piszą programy podczas rozmowy kwalifikacyjnej?

Czy nowo rekrutowani programiści piszą choć kawałek kodu?

12. Czy praktykujesz korytarzowe testy wygody użytkownika?

Czy weryfikujesz z przygodnymi i przypadkowymi osobami łatwość użytkowania (ang. *user friendly*) stworzonego przez siebie oprogramowania?

Każde pytanie odpowiada jednemu punktowi i jeżeli powyższy test zakończy się wynikiem poniżej jedenastu punktów, Joel przepowiada konieczność wprowadzenia zmian. W warunkach polskich wynik na poziomie ośmiu punktów jest wynikiem bardzo dobrym, a średnia to około sześć, siedem punktów³.

W jednym z kolejnych rozdziałów tej książki omówiona jest metodyka CMMI, która umożliwia precyzyjne opisanie stopnia dojrzałości organizacyjnej w pięciopoziomowej skali. Jej wykorzystanie w organizacji wiąże się zazwyczaj z koniecznością zaangażowania firmy zewnętrznej lub grupy konsultantów, którzy nierzadko muszą spędzić wiele miesięcy, aby opisać stan aktualny i zaplanować kierunki rozwoju. Wykonanie testu

³ <http://www.projectcomplete.pl/blog/entry70>

Joela dla jednej osoby zajmuje nie więcej niż trzydzieści minut i już po trzech maksymalnie pięciu rozmowach można zorientować się w sytuacji bieżącej oraz przygotować pierwszy plan zmian.

Takie skrajnie praktyczne podejście jest cechą charakterystyczną większości prac Joela, którą najlepiej unaocznia „Partyzancki poradnik rekrutacji”. Artykuł ten jest dedykowany osobom, które w przeciągu godziny mają przeprowadzić swoją pierwszą rozmowę rekrutacyjną. Joel przedstawia bardzo proste wytyczne — czy kandydat jest bystry i realizuje wyznaczone cele — oraz wskazuje na konieczność podjęcia jednoznacznej decyzji — tak lub nie. Dodatkowo opisany jest również bardzo ciekawy, szablonowy plan rozmowy rekrutacyjnej.

Za każdym razem gdy pojawi się problem, warto sprawdzić, co myśli o nim Joel. Rada ta stosuje się również tematów tak odległych od IT jak kwestie związane z organizacją przestrzeni biurowej⁴.

⁴ <http://www.jelonsoftware.com/articles/BionicOffice.html>

Rozdział 9.

Przykład — Scrum BlogSerwis

Projekt opisany w podrozdziale „Przykład Prince2 i RUP — BlogSerwis” został wykonany wewnątrz korporacji, własnymi zasobami. Celem całego projektu było stworzenie serwisu webowego, który miał udostępnić użytkownikom możliwość zakładania autorских blogów.

Jeden z kierowników w dziale handlowym otrzymał mandat na realizację projektu i wykorzystanie pracowników działu programistycznego. Proces wytwórczy zaczął się od kilkugodzinnego **spotkania planistycznego**, na którym właściciel produktu zaprezentował swoją wizję całego rozwiązania architektowi oraz kierownikowi programistów. Ten ostatni podjął decyzję, że cały proces powinien zostać zrealizowany zgodnie z metodyką Scrum, ponieważ za jej pomocą najszybciej uda się uzyskać wymierne efekty. Podjęto decyzję, że kierownik z działu handlowego stanie się właścicielem produktu, a kierownik programistów — ScrumMasterem.

Przedstawiono kluczowe funkcjonalności biznesowe i priorytety ich wykonania, gdzie 1 oznacza najważniejsze zadanie, a 4 — najmniej ważne. W ramach systemu wyróżniono dwie grupy użytkowników:

- ◆ User — to właściciel bloga, czyli osoba, która może zarejestrować się w serwisie, zmienić swoje dane, opublikować artykuły i zarządzać komunikatami;
- ◆ Guest — to grupa osób, która nie jest zarejestrowana w systemie, ale ma możliwość dodawania komentarzy do istniejących blogów. Przed dodaniem komentarza użytkownicy muszą wpisać określony ciąg znaków z wygenerowanego obrazka. Rozwiązanie to ma na celu zablokowanie możliwości dodawania komentarzy przez złośliwe oprogramowanie.
- ◆ Admin — to pracownik firmy, która monitoruje zachowanie serwisu i w sytuacjach awaryjnych ma możliwość zablokowania określonych użytkowników lub nadania im nowego hasła.

Postanowiono, że oprócz architekta w projekt zostanie zaangażowany jeszcze jeden programista. Założono cztery tygodniowe sprinty (trzysta dwadzieścia roboczogodzin) i uzgodniono 20% zapas czasu na „nieprzewidziane” sytuacje oraz różnice między idealnym a rzeczywistym dniem roboczym ($\frac{3}{8}$ powinno przekładać się na 37,5% zapasu, ale taka wartość jest zbyt wysoka). Dodatkowo każdemu z zadań zostało przyporządkowane pewne ryzyko, w zależności od tego, z jakim prawdopodobieństwem architekt mógł je oszacować. Na końcu pierwszej sesji cały zespół rozplanował wstępnie, w jakich sprintach zostaną wykonane poszczególne funkcjonalności, i zakończył tworzenie pierwszej wersji **dziennika zadań produktu** — tabela 17.

Tabela 17. Przykładowy dziennik zadań produktu w projekcie Scrum BlogSerwis

ID	Tytuł	Priorytet	Sprint	Wstępne szacunki (godz.)	Ryzyko	Finalne szacunki
1	User — Rejestracja w serwisie	1	1	8	0,1	9
2	User — Logowanie bazujące na własnej puli użytkowników (tzw. forms authentication)	1	1	16	0,1	18
3	User — Możliwość zmiany własnych danych, takich jak login i hasło	1	1	24	0,1	26
4	User — Możliwość dodania własnego zdjęcia	1	1	24	0,5	36
5	User — Zarządzanie artykułami	1	1	40	0,1	44
6	User — Edytowanie artykułów w HTML	1	1	32	0,3	42
7	User/Guest — Publikowanie artykułów	1	1	32	0,5	48
8	Admin — Zarządzanie użytkownikami (blokowanie, zmiana hasła)	1	1	24	0,1	26
Sprint pierwszy				200		249
9	User — Zarządzanie parametrami bloga (tytuł, opis)	2	2	16	0,1	18
10	Guest — Możliwość dodawania komentarzy	2	2	32	0,3	42
11	User — Możliwość zarządzania komentarzami	2	2	48	0,5	72
12	User — Przypisanie słów kluczowych do artykułów	2	2	16	0,2	19
13	User/Guest — Przeszukiwanie artykułów po słowach kluczowych	2	2	16	0,3	21
14	User — Wersje robocze postów (nieopublikowane)	2	2	8	0,1	9
15	Admin — Wykres obciążenia na użytkownika	2	2	8	0,1	9

Tabela 17. Przykładowy dziennik zadań produktu w projekcie Scrum BlogSerwis — ciąg dalszy

ID	Tytuł	Priorytet	Sprint	Wstępne szacunki (godz.)	Ryzyko	Finalne szacunki
16	User — Edycja postów à la Word (prototyp) z wykorzystaniem gotowej kontrolki	3	2	24	0,3	31
17	User — Możliwość dodawania elementów Java Script	3	2	8	0,2	10
18	User — Możliwość dodawania linków	3	2	8	0,1	9
19	User — Możliwość dodawania obrazków	3	3	40	0,4	56
20	User — Proste formatowanie tekstu	3	3	24	0,6	38
21	Audyt bezpieczeństwa	3	3	24	0,5	36
22	User — Różne szaty graficzne (CSS)	4	3	32	0,3	42
23	User — Możliwość utworzenia więcej niż jednego bloga	4	3	16	0,1	18
24	User — Pomoc dla użytkownika	4	3	24	0,3	31
25	Zwiększenie wydajności	4	3	24	0,3	31

Kierownik z działu handlowego opuścił spotkanie, a dołączył do niego programista, który wraz z architektem ma realizować projekt. Resztę dnia cały zespół zastanawiał się, jak dokładnie wykonać zadania produktowe wyznaczone dla pierwszego sprinta. Oto efekty.

1. Konieczne jest wyspecyfikowanie działań przygotowawczych. Ze względu na małą ich liczbę nie zdecydowano się na odrębny sprint, lecz w sprincie pierwszym wyróżniono grupę zadań 0.
2. Okazało się, że wycena zadań metodą dół-góra przyniosła pewne zmiany wycen.
3. Z listy zadań produktowych wyrzucono pozycję 23. „User — Możliwość utworzenia więcej niż jednego bloga”. Stwierdzono, że każdy, kto chce mieć drugi blog, powinien zarejestrować się ponownie pod nowym adresem e-mail.
4. Pozycja 9. „User — Zarządzanie parametrami bloga (tytuł, opis)” musi zostać wykonana w pierwszym sprincie. Zanim zostanie opublikowany artykuł, konieczna jest publikacja bloga.
5. Projekt ma charakter „prototypowy”, a większość prac planowana jest w warstwie interfejsu użytkownika. W związku tym, planuje się stosunkowo niskie pokrycie kodu testami jednostkowymi.
6. Postanowiono w połowie pierwszego sprinta zaangażować testera, który ręcznie sprawdzi, czy wykonana przez programistów funkcjonalność jest gotowa do wykorzystania. Do spotkania dołączył więc tester, z którym zaplanowano nową grupę zadań A, czyli konieczność wykonania testów integracyjnych przy zamknięciu pierwszego sprinta.

Po dołączeniu testera do zespołu wykonawczego zwiększono pulę godzin do wykorzystania na czterysta roboczogodzin (z trzystu dwudziestu), a po uwzględnieniu 20% rezerwy na trzysta dwadzieścia roboczogodzin (z dwustu pięćdziesięciu sześciu). Utworzono **dziennik zadań pierwszego sprinta**, który wyglądał tak, jak w tabeli 18.

Tabela 18. Dziennik zadań pierwszego sprinta w projekcie *Scrum BlogSerwis*

ID	Tytuł	Wstępne szacunki (godz.)	Szacunki (godz.)	Ryzyko	Finalne szacunki (godz.)
	1	Utworzenie katalogu kodu w repozytorium	1	0,1	1
	2	Utworzenie współdzielonego projektu webowego	1	0,1	1
	3	Utworzenie współdzielonej bazy danych	1	0,1	1
	4	Bezpieczne podłączenie projektu web do bazy danych	2	0,2	2
0	Przygotowanie projektu	0	5		5
	1	Projekt strony web, bazującej na CSS	3	0,2	4
	2	Walidacja pól wejściowych	2	0,3	3
	3	Utworzenie bazy danych	1	0,1	1
	4	Zapis danych w nowej bazie DB	2	0,1	2
	5	Wysłanie e-maila rejestracyjnego	4	0,3	5
	6	Testy jednostkowe	2	0,1	2
1	User — Rejestracja w serwisie	9	14		17
	1	Projekt strony web, bazującej na CSS	2	0,1	2
	2	Walidacja pól wejściowych	1	0,2	1
	3	Komunikacja z DB	1	0,1	1
	4	Błędne logowanie	1	0,1	1
	5	Pozytywne logowanie	1	0,1	1
	6	Testy jednostkowe	2	0,1	2
2	User — Logowanie bazujące na własnej puli użytkowników (tzw. forms authentication)	18	8		8
	1	Projekt strony web, bazującej na CSS	8	0,3	10
	2	Walidacja pól wejściowych	4	0,3	5
	3	Komunikacja z DB	4	0,2	5
	4	Testy jednostkowe	4	0,1	4

Tabela 18. Dziennik zadań pierwszego sprinta w projekcie Scrum BlogSerwis — ciąg dalszy

ID	Tytuł	Wstępne szacunki (godz.)	Szacunki (godz.)	Ryzyko	Finalne szacunki (godz.)
3	User — Możliwość zmiany własnych danych	26	20		24
1	Projekt strony web, bazującej na CSS		2	0,1	2
2	Pobranie i zapisanie zdjęcia w DB		10	0,5	15
3	Prezentacja zdjęcia w blogu		4	0,3	5
4	Testy jednostkowe		8	0,5	12
4	User — Możliwość dodania własnego zdjęcia	36	24		34
1	Projekt strony web prezentującej listę artykułów — dodanie nowego, usunięcie (CSS)		8	0,2	10
2	Usuwanie, zmiana artykułu		4	0,1	4
3	Komunikacja z DB		2	0,1	2
4	Testy jednostkowe		4	0,1	4
5	User — Zarządzanie artykułami	44	18		20
1	Projekt strony web prezentującej nowy artykuł (CSS)		4	0,3	5
2	Sprawdzenie, czy artykuł może zostać opublikowany		8	0,5	12
3	Dodanie możliwości edycji artykułu		3	0,1	3
4	Komunikacja z DB		4	0,1	4
5	Testy jednostkowe edycji artykułu		4	0,2	5
6	Testy destrukcyjne polegające na próbie wprowadzenia wadliwych danych		8	0,5	12
7	Testy całego modułu artykułów		4	0,3	5
6	User — Edytowanie artykułów w HTML	46	35		46
1	Projekt strony web zarządzającej parametrami bloga (CSS)		4	0,1	4
2	Sprawdzenie danych przed publikacją		2	0,2	2
3	Komunikacja z DB		1	0,1	1
4	Publikacja (aktualizacja)		3	0,3	4
5	Testy jednostkowe		4	0,3	5
9	User — Zarządzanie parametrami bloga (tytuł, opis)	18	14		16
1	Publikowanie artykułu		8	0,5	12

Tabela 18. Dziennik zadań pierwszego sprinta w projekcie Scrum BlogSerwis — ciąg dalszy

ID	Tytuł	Wstępne szacunki (godz.)	Szacunki (godz.)	Ryzyko	Finalne szacunki (godz.)
	2 Aktualizacja artykułu		12	0,5	18
	3 Komunikacja z DB		3	0,1	3
	4 Testy jednostkowe		8	0,5	12
7	User/Guest — Publikowanie artykułów	46	31		45
	1 Strona z listą użytkowników (CSS)		4	0,2	5
	2 Sortowanie listy		2	0,1	2
	3 Blokowanie lub odblokowanie użytkownika (Usera)		3	0,1	3
	4 Poprawki w stronie logowania		2	0,1	2
	5 Ponowne nadanie hasła użytkownikowi		2	0,1	2
	6 Wysłanie e-maila z nowym hasłem		4	0,3	5
8	Admin — Zarządzanie użytkownikami (blokowanie, hasła)	26	17		19
	1 Przygotowanie środowiska testowego		4	0,3	5
	2 Instalacja rozwiązania na systemie testowym		4	0,5	6
	3 Testy dla „leniwego blogera”		8	0,5	12
	4 Testy dla „pracowitego blogera”		16	0,5	24
	5 Testy komentarzy		8	0,5	12
	6 Testy wydajnościowe		24	0,5	36
	7 Poprawki		40	0,5	60
A	Testy integracyjne	0	104		155
	TOTAL		269		389

Koszt wykonania zadań opiewał na trzysta osiemdziesiąt dziewięć godzin, a do dyspozycji było trzysta dwadzieścia godzin! W związku z tym, po intensywniej dyskusji wykonano krótki telefon do właściciela produktu. Ostatecznie podjęto końcowe decyzje o przesunięciu zadań 4. „User — Możliwość dodania własnego zdjęcia” i 8. „Admin — Zarządzanie użytkownikami (blokowanie, hasła)” z pierwszego sprinta do drugiego sprinta. Dzięki tym działaniom koszt pierwszego sprinta został zredukowany do trzystu trzydziestu sześciu godzin. Wspólnie ustalono, że ogólną rezerwę czasową można zmniejszyć do poziomu 16% ze względu na to, że szacunki poszczególnych zadań posiadają już pewne zapasy związane z ryzykiem.

Projekt rozpoczął się i cały zespół przy porannej kawie omawiał krótko postępy prac, plany na dzień następny i możliwe przeszkody. Przez pierwszą połowę sprinta tester uczestniczył w tych spotkaniach okazjonalnie. W trakcie spotkań często pojawiała się tendencja do wchodzenia w niskopoziomą dyskusję, ale ScrumMaster krótko ucinął tego typu rozmowy — obaj programiści siedzieli obok siebie, więc bezproblemowo mogli kontynuować dyskusję przy swoich biurkach. Raz w tygodniu na spotkaniach pojawiał się właściciel produktu. Jeden raz na takim spotkaniu zjawił się dyrektor działu handlowego i zgłosił wątpliwość dotyczącą zakresu projektu. ScrumMaster poprosił o przedyskutowanie tej kwestii po spotkaniu. W trakcie realizacji projektu odnotowano również wiele sytuacji odbiegających od planów. Oto one.

1. 2. dzień — funkcjonalność wysyłania e-maili zabrała więcej czasu niż planowano, ze względu na problemy z konfiguracją serwera SMTP.
2. 4. dzień — wystąpiły problemy z określeniem „publikacyjności” artykułu, gdyż okazało się, że dostępne biblioteki nie sprawdzały wygenerowanego kodu HTML w wystarczający sposób. Konieczne były dodatkowe zasady weryfikacji i założenie, że końcowy użytkownik zna podstawową składnię kodu HTML.
3. 10. dzień — okazało się, iż aktualizacja artykułu jest czynnością o wiele bardziej skomplikowaną niż podejrzewano.
4. 11. dzień — tuż przed rozpoczęciem testów jeden z programistów rozchorował się i nie było go dwa dni.
5. 13. dzień — wykonano pierwszą publikację i zainstalowano ją w środowisku testowym.
6. 15. dzień — ze względu na „niedoczas” programiści, oprócz prac nad poprawkami, wzięli również udział w testach (trwały one do dwudziestego dnia).
7. 20. dzień — zakończono testy pierwszej publikacji i zespół przygotował drugą publikację gotową do kolejnej serii testów. Właściciel produktu podjął decyzję o przygotowaniu wewnętrznej prezentacji rozwiązania.

Przez cały okres trwania projektu ScrumMaster odnotowywał w dzienniku zadań pierwszego sprinta stopień jego realizacji — tabela 19.

Na podstawie powyższych danych ScrumMaster wygenerował również diagram spalania — rysunek 59.

Jak widać na diagramie spalania, największy wpływ na cały pierwszy sprint miała dwudniowa choroba jednego z programistów, która rozpoczęła się jedenastego dnia.

Po zaraportowaniu zakończenia pierwszego sprinta dyrektor i kierownik z działu handlowego postanowili podsumować go ze ScrumMasterem. W trakcie tego spotkania zaprezentowano rezultaty pracy. Nie udało się uniknąć kilku drobnych potknięć technicznych. Przedstawiono jednak całość rozwiązania od momentu rejestracji użytkownika, poprzez utworzenie nowego artykułu, aż do momentu jego opublikowania i możliwości odczytu przez użytkownika typu Guest. Dyrektor był bardzo pozytywnie zaskoczony, że w tak krótkim okresie czasu udało się uzyskać tak wiele. Wskazał braki w dokumentacji, ale ze zrozumieniem przyjął informację, że system pomocy dla użytkownika jest planowany na kolejne sprinty.

Tabela 19. Przykładowa realizacja pierwszego sprinta w projekcie Scrum BlogSerwis

		Kolejne dni projektu																				
ID	Tytuł	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	Utworzenie katalogu kodu w repozytorium	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	Utworzenie współdzielonego projektu webowego	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	Utworzenie współdzielonej bazy danych	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	Bezpieczne podłączenie projektu web do bazy danych	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	Przygotowanie projektu	5																				
1	Projekt strony web, bazującej na CSS	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	Walidacja pól wejściowych	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	Utworzenie bazy danych	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	Zapis danych w nowej bazie DB	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	Wysłanie e-maila rejestracyjnego	5	5	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	Testy jednostkowe	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	User — Rejestracja w serwisie	17																				
1	Projekt strony web, bazującej na CSS	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	Walidacja pól wejściowych	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	Komunikacja z DB	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	Błędne logowanie	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	Pozytywne logowanie	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	Testy jednostkowe	2	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabela 19. Przykładowa realizacja pierwszego sprinta w projekcie Scrum BlogSerwis — ciąg dalszy

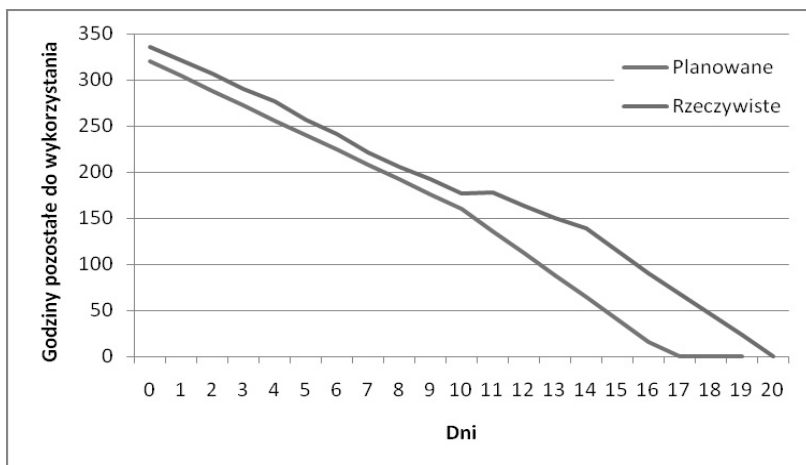
ID	Tytuł	Kolejne dni projektu																				
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	User — Logowanie bazujące na własnej puli użytkowników (tzw. forms authentication)	8																				
1	Projekt strony web, bazującej na CSS	10	10	10	10	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	Walidacja pól wejściowych	5	5	5	5	5	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	Komunikacja z DB	5	5	5	5	5	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	Testy jednostkowe	4	4	4	4	4	4	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	User — Możliwość zmiany własnych danych	24																				
1	Projekt strony web prezentującej listę artykułów — dodanie nowego, usunięcie (CSS)	10	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	Usuwanie, zmiana artykułu	4	4	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	Komunikacja z DB	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	Testy jednostkowe	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	User — Zarządzanie artykułami	20																				
1	Projekt strony web prezentującej nowy artykuł (CSS)	5	5	5	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	Sprawdzenie, czy artykuł może zostać opublikowany	12	12	12	12	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	Dodanie możliwości edycji artykułu	3	3	3	3	3	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	Komunikacja z DB	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	Testy jednostkowe edycji artykułu	5	5	5	5	5	5	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabela 19. Przykładowa realizacja pierwszego sprinta w projekcie Scrum BlogSerwis — ciąg dalszy

ID	Tytuł	Kolejne dni projektu																				
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
6	Testy destrukcyjne polegające na próbie wprowadzenia wadliwych danych	12	12	12	12	12	12	12	6	0	0	0	0	0	0	0	0	0	0	0	0	0
7	Testy całego modułu artykułów	5	5	5	5	5	5	5	3	0	0	0	0	0	0	0	0	0	0	0	0	0
6	User — Edytowanie artykułów w HTML	46																				
1	Projekt strony web zarządzającej parametrami bloga (CSS)	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	Sprawdzenie danych przed publikacją	2	2	2	2	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	Komunikacja z DB	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
4	Publikacja (aktualizacja)	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0
5	Testy jednostkowe	5	5	5	5	5	5	5	3	0	0	0	0	0	0	0	0	0	0	0	0	0
9	User — Zarządzanie parametrami bloga (tytuł, opis)	16																				
1	Publikowanie artykułu	12	12	12	12	12	12	12	12	12	8	0	0	0	0	0	0	0	0	0	0	0
2	Aktualizacja artykułu	18	18	18	18	18	18	18	18	18	14	7	0	0	0	0	0	0	0	0	0	0
3	Komunikacja z DB	3	3	3	3	3	3	3	3	3	0	0	0	0	0	0	0	0	0	0	0	0
4	Testy jednostkowe	12	12	12	12	12	12	12	12	12	12	8	16	8	0	0	0	0	0	0	0	0
7	User/Guest — Publikowanie artykułów	45																				
1	Przygotowanie środowiska testowego	5	5	5	5	5	5	5	5	5	5	5	5	5	5	0	0	0	0	0	0	0

Tabela 19. Przykładowa realizacja pierwszego sprinta w projekcie Scrum BlogSerwis — ciąg dalszy

ID	Tytuł	Kolejne dni projektu																				
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	Instalacja rozwiązania na systemie testowym	6	6	6	6	6	6	6	6	6	6	6	6	6	6	0	0	0	0	0	0	0
3	Testy dla „leniwego blogera”	12	12	12	12	12	12	12	12	12	12	12	12	12	12	7	0	0	0	0	0	0
4	Testy dla „pracowitego blogera”	24	24	24	24	24	24	24	24	24	24	24	24	24	24	22	12	0	0	0	0	0
5	Testy komentarzy	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	0	0	0	0
6	Testy wydajnościowe	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	18	0
7	Poprawki	60	60	60	60	60	60	60	60	60	60	60	60	60	60	45	30	20	10	5	0	0
A	Testy integracyjne	155																				
	Dzienne spalanie — planowane	16	16	16	16	16	16	16	16	16	16	16	16	24	24	24	24	24	16	0	0	0
	SUMA — Planowane	336	320	304	288	272	256	240	224	208	192	176	160	136	112	88	64	40	16	0	0	0
	Dzienne spalanie — rzeczywiste	15	14	17	13	20	16	20	15	13	16	-1	15	13	11	24	25	22	22	23	23	23
	SUMA — Rzeczywiste	336	321	307	290	277	257	241	221	206	193	177	178	163	150	139	115	90	68	46	23	0



Rysunek 59. Przykładowy wykres spalania

Po burzliwej dyskusji wyznaczono zupełnie inny, nowy kierunek rozwoju całego serwisu, który planowano zintegrować z istniejącą bazą wiedzy. Okazało się, że rozwój części administracyjnej stał się zupełnie zbędny.

Po spotkaniu ScrumMaster zaprosił architekta, programistów i testera na oddzielnie spotkanie, aby podsumować przebieg projektu. Porównano metodykę Scrum do standardowych praktyk typu kaskadowego.

Oto wady metodyki Scrum odkryte podczas pracy zespołu.

- ♦ Zadania nie były przypisane do programistów i nie można było sprawdzić trafności estymacji zadań na członka zespołu.
- ♦ Brak możliwości nadania zadaniom ograniczeń typu: „Zadanie B nie może być rozpoczęte, zanim nie skończy się zadanie A”.
- ♦ Brak możliwości „zaznaczenia” dni wolnych w kontekście kraju, firmy, osoby.

Oto pozytywne cechy Scrum.

- ♦ Podejście bardzo „zespala” zespół.
- ♦ Bardzo duża elastyczność w trakcie realizacji projektu.
- ♦ Krótkie, realne czasy dostarczenia pierwszych wersji „poglądowych”.
- ♦ Mniejsza doza „biurokracji”.

Wujek Dobra Rada — odcinek 11. „Osiodłać pasję”



NIE MOŻNA KUPIĆ LUDZKIEJ PASJI WKŁADANEJ W ZADANIE
ANI PRZEWIDZIEĆ, JAKI EFEKT PRZYNIESIE ONA SAMA.

Część IV

Metodyki organizacyjne a praktyka

Metodyki zarządcze i wytwórcze systematyzują sposób przygotowania, realizacji i zamknięcia projektów. Wiele działań prowadzonych przez zespoły IT nie jest jednak wykonywanych w reżimie projektowym. Głównym celem działów informatycznych jest przecież utrzymanie i rozwój infrastruktury technicznej, tak aby pracownicy innych działów mogli realizować własne cele. Co więcej, nie wystarczy już dzisiaj sam fakt, że coś działa, ale istotna jest również jakość świadczonych usług. Czy przepustowość łączy jest wystarczająca? Czy czasy reakcji wdrożonego systemu są zadowalające? Czy help desk udziela kompetentnych odpowiedzi?

Aby uzyskać te informacje, konieczne jest odpowiednie skatalogowanie i monitorowanie usług świadczonych przez działy IT. Tym właśnie tematem zajmują się takie metodyki jak ITIL. Kto nigdy o niej nie słyszał, powinien zastanowić się nad doбором swojej lektury ☺. Inne rozwiązanie, które służy do porządkowania procesów IT, weryfikacji ich dojrzałości oraz zarządzania ryzykami, to metodyka COBIT. Jest to jedno z ciekawszych rozwiązań w obszarze wprowadzania ładu korporacyjnego. W tej części książki poruszona zostanie również tematyka kluczowym miar jakościowych. Jednym z najbardziej ambitnych podejść w tym obszarze jest metodyka Six Sigma, której celem jest rygorystyczna eliminacja wszelkich błędów procesowych. Tak restrykcyjne podejście do kwestii jakościowych dotyczy głównie firm, które świadomie zmierzają do osiągnięcia jak najwyższego poziomu dojrzałości organizacyjnej. Ciekawym podejściem, umożliwiającym mierzenie tej dojrzałości jest model CMMI.

Rozdział 10.

Capability Maturity Model Integration (CMMI)

Zmierzenie dojrzałości organizacyjnej firmy lub działów IT jest przedsięwzięciem skomplikowanym i dlatego stale tworzone są kolejne mechanizmy służące do tego celu. Najbardziej powszechnie znane są normy Międzynarodowej Organizacji Normalizacyjnej. W Polsce zyskały one dużą popularność i wiele firm chwali się uzyskaniem certyfikatów ISO. Przykładowo w przypadku kwestii bezpieczeństwa wykorzystuje się następujące normy:

- ◆ ISO 15048 — ocena bezpieczeństwa sprzętu i oprogramowania,
- ◆ ISO 13335 — wytyczne służące do zapewniania bezpieczeństwa systemów informatycznych,
- ◆ ISO 17799 — wytyczne dotyczące zarządzania bezpieczeństwem informacji,
- ◆ ISO 27001 — budowa systemów zarządzania bezpieczeństwem informacji,
- ◆ BS 25999 — zarządzanie ciągłością działania.

Generalnie jednak normy ISO nie przystają do realiów świata IT i o wiele lepiej sprawdza się tutaj model CMMI. Zapewnia on dobrą platformę do właściwej weryfikacji i zamodelowania procesów związanych z działami informatycznymi. CMMI nie jest metodyką, a raczej mechanizmem oceny dojrzałości firmy. Ocena ta bazuje — co prawda — na modelowych grupach procesowych, ale zostały one wymienione po to, aby organizacja mogła samodzielnie doskonalić się w konkretnych obszarach.

Szczypta historii

W latach 80. ubiegłego wieku amerykański Departament Obrony ogłosił przetarg na opracowanie sposobu oceny partnerów współpracujących z sektorem publicznym. W 1991 roku została opublikowana pierwsza wersja „modelu dojrzałości organizacyjnej” (ang. *Capability Maturity Model*) opracowana we współpracy z Software Engineering Institute (SEI) i niezależną organizacją Mitre Corporation. Dwa lata później firma SEI opublikowała nową wersję (1.1) tego modelu.

W kolejnych latach SEI upubliczniła szereg dedykowanych modeli dojrzałości:

- ♦ SW-CMM — wytwarzanie oprogramowania (dziedzina bazowa),
- ♦ SA-CMM — odbiór albo przejęcie oprogramowania przez klientów,
- ♦ SE-CMM — inżynieria oprogramowania opisująca pełen cykl życia oprogramowania,
- ♦ IDP-CMM — zintegrowane wytwarzanie produktów,
- ♦ P-CMM (People-CMM) — zarządzanie zasobami ludzkimi.

Powstanie CMM zbiegło się w czasie z takimi inicjatywami jak:

- ♦ SPICE (ang. *Software Process Improvement and Capability dEtermination*), który jest obecnie normą ISO/IEC 15504,
- ♦ SECAM (ang. *Systems Engineering Capability Assessment Model*) zaprojektowany przez International Council on System Engineering (INCOSE).

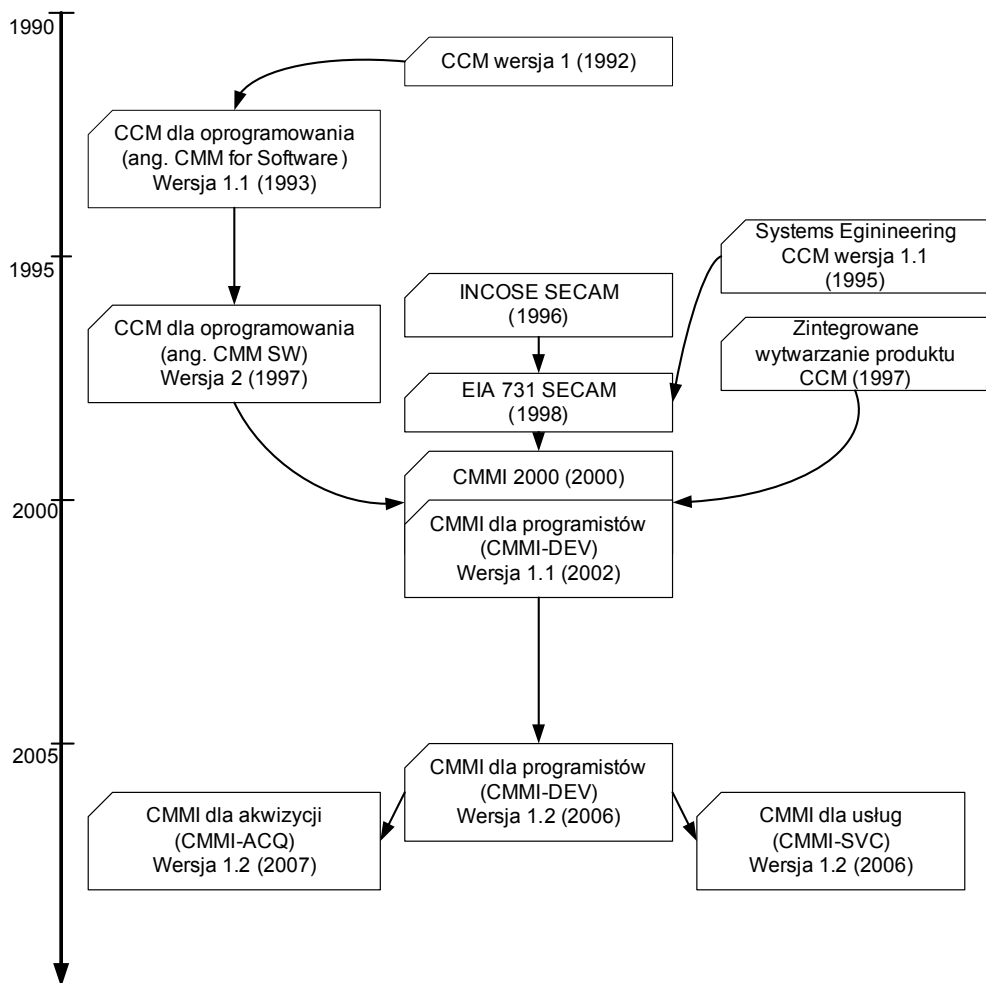
Z czasem SEI postanowiła bardziej zunifikować modele i tak powstał CMMI 2000 (ang. *Capability Maturity Model Integration*). W 2006 roku pojawiła się publikacja CMMI w wersji 1.2. Proces unifikacji nie zakończył się powodzeniem, ponieważ od 2007 roku SEI znowu zaczęła znowu publikować różne, dedykowane modele. Do maja 2008 roku powstały trzy:

- ♦ CMMI dla programistów (CMMI-DEV) — odpowiednik starego SW-CMM, zajmuje się wytwarzaniem usług i oprogramowania,
- ♦ CMMI dla akwizycji (CMMI-ACQ) — odpowiednik starego SA-CMM, zajmuje się odbiorami usług i oprogramowania,
- ♦ CMMI dla usług (CMMI-SVC) — odpowiednik starego SE-CMM, zajmuje się cyklem życia usług i oprogramowania.

Każdy z tych modeli jest niezależny od sektora gospodarczego i może być stosowany w przemyśle lotniczym, bankowości, w sektorze publicznym, motoryzacyjnym, przemysłowym i telekomunikacji.

Pełną ewolucję modelu CMM do CMMI przedstawiono na rysunku 60.

Poniższe opracowania koncentrują się na bazowym modelu CMMI-DEV 1.2.



Rysunek 60. Ewolucja modelu CMMI



W praktyce

W Polsce, wraz z napływem zachodnich korporacji, pojawiają się pierwsze organizacje, które mogą pochwalić się oficjalnym wdrożeniem CMMI. Krakowska Motorola posiada 5. poziom CMMI i oprócz tego wdrożyła komplementarny model SixSigma. Na ich straży stoi dr Marek Rydzy¹. W GE Money Banku Janusz Stankiewicz — dyrektor departamentu IT — wdrożył procesy oparte m.in. właśnie na CMMI, a w Warszawie HP mówi o 3. poziomie CMMI.

¹ Computerworld, nr 12-13/807, 25 marca 2008, „Enface” dr Marek Rydzy, Dorota Konowrocka

Jednym z popularyzatorów tego rozwiązania w Polsce jest firma Software Mind. Oto wypowiedź Grzegorza Młynarczyka, wiceprezesa Software Mind, z 14 lutego 2007 roku:

Główną przyczyną mniejszej popularności modelu CMMI/SPI (ang. Software Process Improvement) w Europie Centralnej i Wschodniej jest, naszym zdaniem, brak informacji o możliwościach i korzyściach, jakie daje zastosowanie takich modeli w firmach. Pracując przy innowacyjnych projektach dla wymagających klientów, zauważyliśmy rosnące potrzeby związane z organizowaniem skomplikowanych procesów wytwórczych. Jako firma z branży IT kładziemy bardzo duży nacisk na odpowiednią optymalizację procesów produkcji. Wielokrotnie sprawdzaliśmy w codziennej pracy czasami skrajnie odmienne podejścia do procesu produkcji oprogramowania i na bazie tych doświadczeń możemy stwierdzić, iż model CMMI dla wielu organizacji stanowi bardzo dobre rozwiązanie. Korzyści z zastosowania tego modelu płyną zarówno dla firm, które produkują oprogramowanie, jak również dla klientów współpracujących z wieloma dostawcami IT².

Poziomy CMMI

CMMI definiuje pięć poziomów dojrzałości (ang. *maturity level*) organizacji. Oto one.

- ♦ Poziom 0 — organizacja nie działa.
- ♦ Poziom 1. — wszystko dzieje się samoistnie i w sposób niekontrolowany, a organizacja ma marginalny wpływ na przebieg wypadków. Firmy na poziomie 1. działają dzięki zaangażowaniu ze strony poszczególnych osób.
- ♦ Poziom 2. — organizacja jest faktycznie zarządzana i mają miejsce powtarzalne procesy.
- ♦ Poziom 3. — organizacja jest zdefiniowana i opisana, działa w sposób powtarzalny, zgodnie z przyjętą strategią i procesami biznesowymi.
- ♦ Poziom 4. — organizacja jest w stanie liczbowo zmierzyć wydajność poszczególnych procesów.
- ♦ Poziom 5. — organizacja stale optymalizuje swoje procesy na bazie miar wydajności.

Każdy z obszarów procesowych może osiągnąć jeden z pięciu poziomów zdolności (ang. *capability levels* — CL).

- ♦ 0 — niekompletny — cele nie są realizowane.
- ♦ 1. — wykonywany — cele są realizowane, ale głównie dzięki zaangażowaniu konkretnych osób, a nie organizacji.

² http://www.softwaremind.pl/pl/CMMI%C2%AE_-_Dlaczego_powinno_Cie_to_obchodzic_, Monika Błotnicka, 2007-02-14

- ♦ 2. — zarządzany — cele są realizowane na bazie przygotowanych planów.
- ♦ 3. — zdefiniowany — organizacja wdrożyła politykę normalizującą procesy.
- ♦ 4. — zarządzany ilościowo — realizacja procesów jest mierzalna ilościowo.
- ♦ 5. — optymalizowany — procesy są ciągle udoskonalane i dostosowywane do rzeczywistości organizacyjnej.

Procesy

CMMI składa się z dwudziestu dwóch obszarów procesowych, z których każdy należy do jednej z czterech kategorii:

1. Zarządzanie projektem (sześć obszarów),
2. Zarządzanie procesami (pięć obszarów),
3. Inżynieria (sześć obszarów),
4. Wsparcie (pięć obszarów).

Zarządzanie projektem

Ta kategoria opisuje procesy związane z planowaniem, monitorowaniem i kontrolą projektów. Opisane tu obszary w dużej mierze pokrywają się z tematyką metodyk zarządzanych, takich jak Prince2 i PMBoK.

- ♦ Planowanie projektu (ang. *Project Planning* — PP) to opracowanie i aktualizacja planu zadań projektowych. Oznacza konieczność uzgadniania z zespołem zakresu i czasu prac niezbędnych do realizacji projektu. Dodatkowo obszar ten opisuje konieczność właściwej komunikacji z interesariuszami projektu, co zakłada m.in. uzyskanie zgody na realizację stworzonych planów.
- ♦ Monitorowanie i kontrola projektów (ang. *Project Monitoring and Control* — PMC) to bieżące monitorowanie postępów prac i ewentualne uruchomienie działań naprawczych w przypadku znaczących rozbieżności względem planu bazowego.
- ♦ Zarządzanie umowami z dostawcami (ang. *Supplier Agreement Management* — SAM) zajmuje się relacjami z dostawcami zewnętrznymi. Obszar ten zawiera wybór dostawców, podpisanie i wykonanie umów oraz mechanizmy odbioru wykonanych produktów.
- ♦ Zintegrowane zarządzanie projektem (ang. *Integrated Project Management* — IPM) opisuje mechanizmy uruchomienia projektu, komunikacji z właściwymi interesariuszami projektu (ang. *relevant stakeholders*) i koordynacji prac zarządczych. Oznacza to zdefiniowanie bazowego zbioru procesów i standardów. Każdy z projektów może zaimplementować wybrany podzbiór tych procesów.

- ♦ Zarządzanie ryzykiem (ang. *Risk Management* — RSKM) to identyfikacja potencjalnych problemów i zarządzanie zagrożeniami. Obszar ten powinien koncentrować się na kwestiach, które mogą zagrozić osiągnięciom kluczowych celów projektowych. Zakłada się zaangażowanie właściwych interesariuszy projektu.
- ♦ Ilościowe zarządzanie projektem (ang. *Quantitative Project Management* — QPM) koncentruje się na ustanowieniu projektowych celów jakościowych i wydajnościowych oraz zapewnieniu ich mierzalności. Jest to osiągnięte poprzez wybranie odpowiednich podprocesów i zdefiniowanie dla nich miar, które będą monitorowane i analizowane w czasie. Obszar ten zawiera również identyfikację stosownych działań naprawczych.

Zarządzanie procesami

Ta kategoria zawiera obszary związane z czynnościami organizacyjnymi wykonywanymi poza projektami.

- ♦ Modyfikacja procesów organizacyjnych (ang. *Organizational Process Focus* — OPF) to planowanie, implementacja i wdrożenie udoskoaleń procesów, bazujące na wiedzy o słabych oraz silnych stronach procesów organizacyjnych. Obszar ten definiuje konieczność ostrożnego wprowadzania takich udoskoaleń. Zakłada się bardzo staranny etap planowania oraz pilotażowe wdrożenie usprawnienia na testowej grupie użytkowników.
- ♦ Szkolenia (ang. *Organizational Training* — OT) koncentrują się na budowie umiejętności i wiedzy wśród pracowników. Cel ten jest osiągnięty poprzez kompleksowe zarządzanie szkoleniami od momentu identyfikacji potrzeb szkoleniowych do mierzenia efektywności przeprowadzonych szkoleń. Zdefiniowane potrzeby szkoleniowe powinny występować powszechnie w większości projektów lub zespołów wsparcia. Potrzeby te muszą być zgodne ze strategicznymi, biznesowymi celami całej organizacji. Proces ten nie zajmuje się potrzebami szkoleniowymi w kontekście konkretnego projektu lub zespołu wsparcia.
- ♦ Wydajność procesów organizacyjnych (ang. *Organizational Process Performance* — OPP) wdraża model ilościowej reprezentacji jakości i wydajności procesów oraz wytwarzanych produktów. W ramach tego procesu definiowane są kluczowe miary oraz wartości bazowe, które te miary powinny osiągnąć. Zebrane wyniki są poddawane wstępnej analizie i przekazane właściwym osobom. Przykładowe miary mogą opisywać długość cyklu wytwórczego, efektywność usuwania defektów, ilość defektów, czasy reakcji lub koszty. Jedynie wybrane procesy powinny być mierzone statystycznie i wskazane jest zdefiniowanie kryteriów, które pozwolą jednoznacznie wybrać takie procesy.
- ♦ Wdrożenie innowacji organizacyjnych (ang. *Organizational Innovation and Deployment* — OID) koncentruje się na wyborze i wdrożeniu usprawnień, które wymiennie udoskoala procesy organizacyjne albo wytwarzane produkty. Takie usprawnienia powinny mieć uzasadnienia związane z celami biznesowymi

organizacji. Usprawnienia mogą wpływać na dodatkową funkcjonalność produktu, jego jakość, zwiększoną produktywność, skrócone czasy wytwarzania, skrócone czasy wdrożenia albo większą satysfakcję klienta.

- ♦ Definicja procesów organizacyjnych (ang. *Organizational Process Definition* — OPD) to ustanowienie i utrzymanie zbioru procesów organizacyjnych oraz zapewnienie środowiska pracy zgodnego ze standardami. Proces ten ma na celu uporządkowanie i utrzymanie biblioteki procesów, standardów i dobrych praktyk funkcjonujących w organizacji.

Inżynieria

Ta kategoria opisuje techniczne procesy związane z czynnościami wytwórczymi i utrzymanowymi.

- ♦ Stworzenie wymagań (ang. *Requirements Development* — RD) to analiza i udokumentowanie wymagań klienta, produktów i komponentów. Wymagania opisują nie tylko funkcjonalności, ale definiują również ograniczenia związane z wybranymi rozwiązaniami projektowymi.
- ♦ Zarządzanie wymaganiami (ang. *Requirements Management* — REQM) obejmuje zarządzanie wymaganiami produktowymi. Częścią tych prac zarządczych jest właściwe sterowanie zmianami oraz ich synchronizacja względem planów i prac produktowych. W ramach tego obszaru identyfikowane są również niespójności pomiędzy wymaganiami a planami i pracami produktowymi.
- ♦ Integracja produktu (ang. *Product Integration* — PI) to „złożenie” komponentów w końcowy produkt przy zapewnieniu integralności dostarczonego rozwiązania. Główny nacisk jest położony na przyrostowy sposób dochodzenia do tego celu poprzez iteracyjne dokładanie kolejnych komponentów i funkcjonalności.
- ♦ Rozwiązania techniczne (ang. *Technical Solution* — TS) to projektowanie, tworzenie i implementacja rozwiązań zgodnych z wymaganiami. Obszar ten jest stosowany do każdego produktu i komponentu oraz procesów związanych z wytwarzaniem produktów. Uwzględnione są również kwestie związane z wyborem najlepszych rozwiązań technicznych. Warto każdorazowo rozpatrzyć wariant zakupu gotowych rozwiązań komercyjnych (ang. *commercial off-the-shelf COTS*).
- ♦ Walidacja (ang. *Validation* — VAL) to demonstracja, że produkty i komponenty mogą być należycie użyte w docelowym środowisku. Innymi słowy, jest to sprawdzenie, że produkty zostały właściwie zbudowane. W tym obszarze bardzo często występuje zaangażowanie użytkowników końcowych oraz innych interesariuszy projektu. Walidacja jest często realizowana równolegle z „Weryfikacją”.
- ♦ Weryfikacja (ang. *Verification* — VER) to zapewnienie, że produkty są zgodne ze stworzonymi wymaganiami klienta, produktów i komponentów. W przeciwnym przypadku proces ten specyfikuje konieczność wykonania poprawek. Proces zawiera zarówno mechanizmy wyboru produktów do weryfikacji, jak i opis metod oraz kryteriów weryfikacji. Występuje również technika wzajemnych

weryfikacji (ang. *peer reviews*), gdzie członkowie zespołu projektowego weryfikują wzajemnie wykonane przez siebie prace. O ile walidacja definiuje właściwe zbudowanie produktu, to weryfikacja definiuje, że zbudowano właściwy produkt.

Wsparcie

Ta kategoria opisuje wszelkie działania związane ze wsparciem wytwarzania produktów i ich utrzymaniem.

- ♦ Zarządzanie konfiguracją (ang. *Configuration Management* — CM) to ustanowienie i utrzymanie integralności rozwiązań z użyciem mechanizmów konfiguracyjnych. Oznacza to identyfikację potrzeb konfiguracyjnych, kontrolę zmian wprowadzanych w konfiguracji, bezpieczne przechowanie stanu konfiguracji i audyty.
- ♦ Mierzenie i analiza (ang. *Measurement and Analysis* — MA) to zapewnienie „mierzalności” stanu projektu poprzez zdefiniowanie miar, mechanizmów kolekcjonowania i składowania niezbędnych danych oraz wdrożenie odpowiednik technik analitycznych.
- ♦ Procesowe i produktowe zapewnienie jakości (ang. *Process and Product Quality Assurance* — PPQA) to zapewnienie personelu i zarządzania ukierunkowanego na jakość procesów oraz jakość związanych z nimi produktów. Osiągnięte jest to poprzez zapewnienie mechanizmów oceny procesów i produktów oraz dokumentowanie przypadków niezgodności ze standardami. Informacje o takich przypadkach muszą zostać przekazane do właściwych osób.
- ♦ Analiza decyzji i rozwiązań (ang. *Decision Analysis and Resolution* — DAR) to obszar, którego głównym celem jest eliminacja subiektywności podejmowanych decyzji poprzez formalizację procesu. Oznacza to zidentyfikowanie możliwych rozwiązań alternatywnych i przygotowanie odpowiednich kryteriów wyboru. Dopiero po tych działaniach możliwe jest podjęcie obiektywnej i uzasadnionej decyzji.
- ♦ Nieformalna analiza i rozwiązania (ang. *Causal Analysis and Resolution* — CAR) to identyfikacja przyczyn defektów (problemów). Kluczem do sukcesu jest tutaj wykrycie przyczyn problemu. Obszar ten zajmuje się również uruchomieniem działań zapobiegawczych w celu uniknięcia defektów w przyszłości. Zakłada się, że wdrożone zostały już mechanizmy mierzenia wydajności procesów i dlatego możliwa jest szybka analiza efektów wprowadzonych zmian.

Każdy z obszarów procesowych jest opisany szczegółowo przez zbiory celów dedykowanych (ang. *Specific Goals* — SG). Mogą one zostać osiągnięte za pomocą dedykowanych praktyk (ang. *Specific practices* — SP) oraz pięciu zbiorów celów generycznych (ang. *Generic Goals* — GG). Cele generyczne mogą z kolei zostać osiągnięte przez zastosowanie generycznych praktyk (ang. *Generic practices* — GP).

Istnieją dwa sposoby podejścia do modelu CMMI (reprezentacje), a zależą one od motywacji organizacji. W zależności od przyjętej reprezentacji wdrażane są różne grupy celów generycznych.

1. Reprezentacja stopniowana (ang. *staged representation*)

W tym przypadku organizacja koncentruje się na zdefiniowaniu swojego poziomu dojrzałości CMMI (ang. *maturity model*) w danym momencie. Sprawdzane jest najpierw, czy dla konkretnego poziomu dojrzałości wdrożono wszystkie wymagane obszary procesowe i czy zrealizowane są wymagane cele generyczne. Po pierwszej ocenie organizacja może narzucić sobie cel osiągnięcia kolejnego poziomu CMMI i plan wdrożenia określonej puli obszarów procesowych. Przykładowo do osiągnięcia 2. poziomu CMMI niezbędne jest wdrożenie obszarów:

- ♦ Zarządzanie wymaganiami (ang. *Requirements Management* — REQM),
- ♦ Planowanie projektu (ang. *Project Planning* — PP),
- ♦ Monitorowanie i kontrola projektów (ang. *Project Monitoring and Control* — PMC),
- ♦ Zarządzanie umowami z dostawcami (ang. *Supplier Agreement Management* — SAM),
- ♦ Zarządzanie konfiguracją (ang. *Configuration Management* — CM),
- ♦ Mierzenie i analiza (ang. *Measurement and Analysis* — MA),
- ♦ Procesowe i produktowe zapewnienie jakości (ang. *Process and Product Quality Assurance* — PPQA).

Innymi słowy, reprezentacja stopniowa oznacza, że dopiero po osiągnięciu poziomu 2. można przygotować się do osiągnięcia poziomu 3. W tabeli 20. zaprezentowano listę obszarów, których uruchomienie jest wymagane w kontekście konkretnego poziomu CMMI.

2. Reprezentacja ciągła (ang. *continuous representation*)

W tym przypadku wyznaczenie bieżącego poziomu dojrzałości (ang. *maturity model*) nie jest celem głównym. Główny nacisk jest położony na szczegółową analizę obszarów procesowych i stałe mierzenie poziomu umiejętności w tych obszarach (ang. *capability level*).

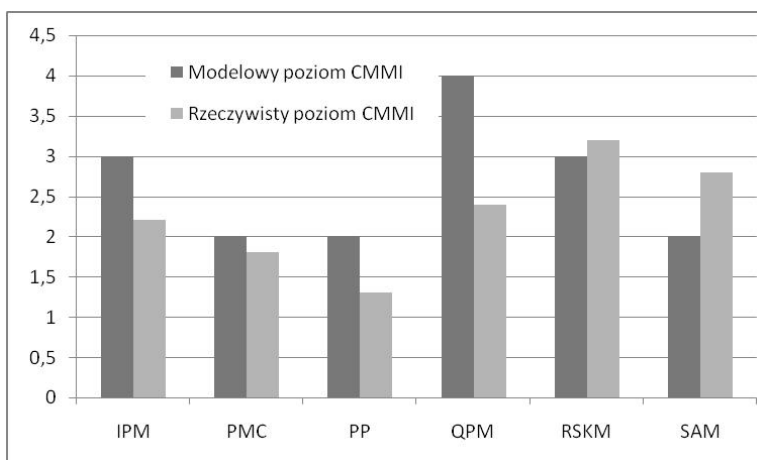
Takie podejście umożliwia bardzo precyzyjne dopasowanie modelu do specyfiki firmy. Dzięki temu można szybko wyznaczyć słabe i silne punkty organizacji, a następnie wskazać obszary do poprawy. Obszary te mogą zostać wyznaczone według potrzeb organizacji i metodyka nie narzuca tutaj żadnych ograniczeń.

Głównym celem reprezentacji ciągłej jest stałe monitorowanie i udoskonalanie wyznaczonych obszarów procesowych. Przykładowy wykres kontrolny obszarów procesowych z kategorii „Zarządzanie projektami” przedstawiony został na rysunku 61.

Reprezentacja ciągła daje możliwość zdefiniowania poziomu dojrzałości poprzez mechanizm „odpowiednika poziomu” (ang. *equivalent staging*). Można przyjąć, że cała organizacja osiągnęła odpowiedni poziom dojrzałości (ang. *maturity level ML*), jeżeli zostały wdrożone wszystkie wymagane procesy i osiągnęły one odpowiedni poziom umiejętności (ang. *capability level CL*). Odpowiednikiem poziomu dojrzałości w tej reprezentacji jest profil docelowy (ang. *target profile*).

Tabela 20. *Kategorie procesów w CMMI*

Obszar procesowy	Kategoria	Poziom CMMI
Monitorowanie i kontrola projektów (PMC)	Zarządzanie projektem	2
Planowanie projektu (PP)	Zarządzanie projektem	2
Zarządzanie umowami z dostawcami (SAM)	Zarządzanie projektem	2
Zarządzanie ryzykiem (RSKM)	Zarządzanie projektem	3
Zintegrowane zarządzanie projektem (IPM)	Zarządzanie projektem	3
Ilościowe zarządzanie projektem (QPM)	Zarządzanie projektem	4
Modyfikacja procesów organizacyjnych (OPF)	Zarządzanie procesami	3
Szkolenia (OT)	Zarządzanie procesami	3
Wydajność procesów organizacyjnych (OPP)	Zarządzanie procesami	4
Wdrożenie innowacji organizacyjnych (OID)	Zarządzanie procesami	5
Definicja procesów organizacyjnych (OPD)	Zarządzanie procesami	5
Zarządzanie wymaganiami (REQM)	Inżynieria	2
Rozwiązania techniczne (TS)	Inżynieria	3
Walidacja (VAL)	Inżynieria	3
Weryfikacja (VER)	Inżynieria	3
Integracja produktu (PI)	Inżynieria	3
Stworzenie wymagań (RD)	Inżynieria	3
Mierzenie i analiza (MA)	Wsparcie	2
Procesowe i produktowe zapewnienie jakości (PPQA)	Wsparcie	2
Zarządzanie konfiguracją (CM)	Wsparcie	2
Analiza decyzji i rozwiązań (DAR)	Wsparcie	3
Nieformalna analiza i rozwiązania (CAR)	Wsparcie	5

**Rysunek 61.** *Przykładowe porównanie rzeczywistego poziomu z minimalnie wymaganym poziomem procesów CMMI*

Obszar procesowy	ML	CL1	CL2	CL3	CL4	CL5
Zarządzanie wymaganiami (REQM)	2	Profil docelowy 2				
Planowanie projektu (PP)	2					
Monitorowanie i kontrola projektów (PMC)	2					
Zarządzanie umowami z dostawcami (SAM)	2					
Mierzenie i analiza (MA)	2					
Procesowe i produktowe zapewnienie jakości (PPQA)	2					
Zarządzanie konfiguracją (CM)	2	Profil docelowy 3				
Stworzenie wymagań (RD)	3					
Rozwiązania techniczne (TS)	3					
Integracja produktu (PI)	3					
Weryfikacja (VER)	3					
Walidacja (VAL)	3					
Modyfikacja procesów organizacyjnych (OPF)	3					
Definicja procesów organizacyjnych (OPD)	3					
Szkolenia (OT)	3					
Zintegrowane zarządzanie projektem (IPM)	3					
Zarządzanie ryzykiem (RSKM)	3	Profil docelowy 4				
Analiza decyzji i rozwiązań (DAR)	3					
Wydajność procesów organizacyjnych (OPP)	4	Profil docelowy 5				
Ilościowe zarządzanie projektem (QPM)	4					
Wdrożenie innowacji organizacyjnych (OID)	5					
Nieformalna analiza i rozwiązania (CAR)	5					

Rysunek 62. Mechanizm „odpowiednika poziomu”

Decyzja o wyborze reprezentacji ciągłej lub stopniowej nie jest zobowiązująca, ponieważ przejście od jednej reprezentacji do drugiej jest stosunkowo łatwe. Pełne porównanie reprezentacji zostało zaprezentowane w tabeli 21.

Podsumowanie

Wdrożenie modelu CMMI jest długotrwałe i wymaga wsparcia nie tylko kluczowych pracowników, ale również decydentów strategicznych, ponieważ zazwyczaj ingeruje w kluczowe obszary działalności firmy. W takich wdrożeniach należy liczyć się z oporem organizacyjnym, ponieważ wielu pracowników będzie chciało robić rzeczy „po staremu”. Dlatego tak ważne jest właściwe zakomunikowanie wszystkim członkom zespołu, jakie są cele, priorytety i mechanizmy wdrożenia.

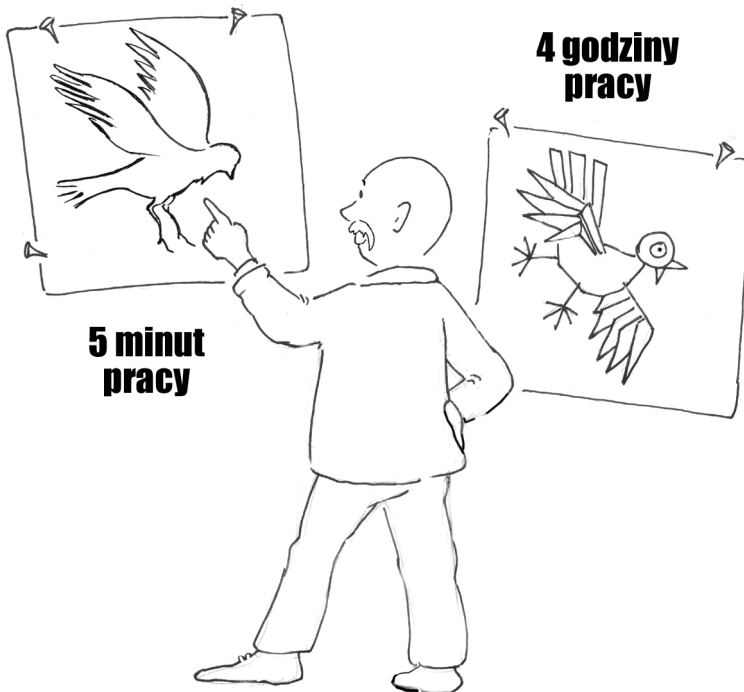
Tabela 21. Porównanie reprezentacji ciągłej i stopniowanej CMMI

Reprezentacja stopniowana	Reprezentacja ciągła
Organizacja wybiera obszary procesowe, bazując na planowanym, docelowym poziomie dojrzałości.	Organizacja wybiera obszary procesowe i ich docelowe poziomy umiejętności, bazując na wewnętrznych celach udoskonalania wyznaczonych procesów.
Ulepszenia są mierzone poziomami dojrzałości (ang. <i>maturity level</i>).	Ulepszenia są mierzone poziomami umiejętności (ang. <i>capability level</i>).
Poziomy dojrzałości są używane do planowania i monitorowania wydajności procesu ulepszania.	Poziomy umiejętności są używane do planowania i monitorowania wydajności procesu ulepszania.
Nie ma potrzeby stosowania odpowiedników.	Poziom dojrzałości może zostać wyznaczony za pomocą „odpowiednika poziomu” (ang. <i>equivalent staging</i>).
Kolejność doskonalenia obszarów procesowych jest zależna od poziomu dojrzałości — ograniczona elastyczność.	Kolejność doskonalenia obszarów procesowych może być dowolna — pełna elastyczność.
Prosty i przejrzysty sposób zwiększania dojrzałości organizacji.	Doskonalenie wszystkich obszarów nie zawsze jest celowe, ale wybór obszarów procesowych do doskonalenia może być błędny.
Lepiej dopasowana do większych organizacji.	Lepiej dopasowana do mniejszych organizacji, ze skutecznymi mechanizmami kontroli.

Należy jasno podkreślić, że najwyższy poziom CMMI nie zawsze jest najlepszy. Każda organizacja musi sama odpowiedzieć na pytanie, po co wdraża to rozwiązanie i co chce osiągnąć. CMMI na poziomie 5. ma sens w organizacjach, które mogą działać w modelu „linii produkcyjnej”, gdy mamy do czynienia z wysoką powtarzalnością wykonywanych zadań. W organizacji realizującej różnorodne projekty, o różnej skali i specyfice, CMMI na poziomie 5. byłby najprawdopodobniej źle dopasowany. W takim przypadku wystarczy najprawdopodobniej osiągnięcie poziomu 2. lub 3.

Tak samo jak niejedno wdrożenie norm ISO, również wdrożenie CMMI może być sztuczne. Jeżeli kluczowi pracownicy hurtowo tworzą dokumentację przed audytem CMMI, jest to oznaką, że cały mechanizm nie działa właściwie lub narzucone cele nie przystają do realiów firmy. Właściwe wdrożony CMMI zapewnia jednak stopniowe wprowadzanie „ładu korporacyjnego” i uporządkowanie procesów.

Wujek Dobra Rada — odcinek 12. „Idealne nie znaczy najlepsze”



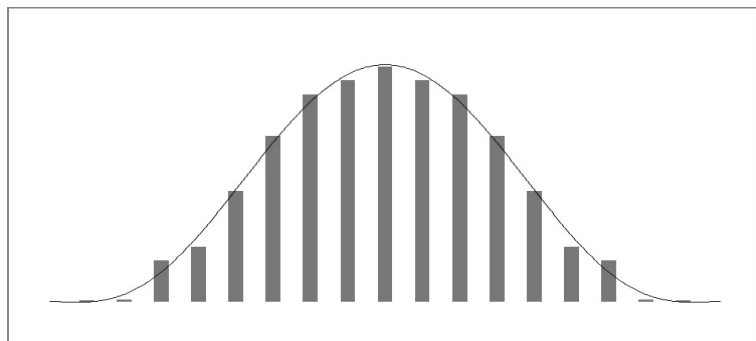
KTOŚ ZAŻYCZYŁ SOBIE POKOJU NA CAŁYM ŚWIECIE...
I WSZYSCY LUDZIE ZNIKNĘLI.

Rozdział 11.

Six Sigma

Jednym z bardziej znanych mechanizmów rachunku prawdopodobieństwa jest rozkład normalny, czyli krzywa Gaussa — rysunek 63. Jest ona również nazywana krzywą normalną błędów, ponieważ bardzo dobrze opisuje rozkład rozbieżności pomiędzy powtarzаныmi pomiarami.

Rysunek 63.
Krzywa Gaussa



Six Sigma to metoda statystyczna, która ma na celu doprowadzenie do sytuacji, że na milion pomiarów jedynie 3,4 wykaże błąd (0,9999966). Jest to bardziej rygorystyczna wersja podejścia Three Sigma, które zakłada 67000 błędów na milion (0,999). Praktyka z linii produkcyjnych wielu dużych korporacji wykazała, że tak “drastyczne” podejście do jakości na każdym etapie procesu jest opłacalne i zasadne. W prostych procesach takie podejście może wydawać się nadmiarowe, ale w przypadku bardziej skomplikowanych ma już duży sens — tabela 22.

Istnieje odchudzona wersja Six Sigma (ang. *Lean Six Sigma*), która łączy w sobie cechy metodyki Six Sigma i wytwarzania odchudzonego (ang. *Lean Manufacturing*). Podejście to charakteryzuje się maksymalnym uproszczeniem procesów wytwórczych w celu jak największej redukcji kosztów oraz niedopuszczania do strat materiałowych. Dzięki tym działaniom można znacząco ograniczyć zatrudnienie oraz jednostkowy koszt wytworzenia.

Tabela 22. *Prawdopodobieństwo sukcesu w przypadku wieloetapowych procesów*

Liczba etapów w procesie	Prawdopodobieństwo sukcesu etapu			
	0,9900000	0,9990000 (Three Sigma)	0,9999000	0,9999966 (Six Sigma)
2	0,9801000	0,9980010	0,9998000	0,9999932
10	0,9043821	0,9900449	0,9990004	0,9999660
100	0,3660323	0,9047921	0,9900493	0,9996601
200	0,1339797	0,8186488	0,9801977	0,9993202
500	0,0065705	0,6063789	0,9512270	0,9983014
1000	0,0000432	0,3676954	0,9048329	0,9966058

Szczypta historii

W połowie lat 80. ubiegłego wieku Motorola w reakcji na rosnącą konkurencję ze strony tanich wyrobów japońskich zaczęła poszukiwania sposobu poprawienia jakości produktów i obniżenia kosztów wytwórczych. Wykorzystała do tego powszechnie znane metody statystyczne. Finalnym efektem było wdrożenie metodyki Six Sigma, która została również zastosowana w firmach GE, Honeywell, Alstom, Raytheon, Texas Instruments, Kodak, Xerox, Home Depot i... Microsoft. W ostatnim przypadku podejście to zostało jednak zastosowane raczej do takich procesów jak dostawa oprogramowania, a nie wytwarzanie projektów informatycznych ☺.



W praktyce

W Polsce, oprócz krakowskiego oddziału firmy Motorola, również TP S.A., Bank Zachodni WBK S.A., ING Bank Śląski, Raiffeisen Bank, ABB i Philips Lighting Poland chwalą się wdrożeniem Six Sigma. Jednym z najbardziej znanych promotorów tego podejścia jest Władysław Popławski, który brał udział we wdrożeniach w GE Capital Bank i Raiffeisen Bank oraz jest założycielem Polskiej Akademii Six Sigma. Innym ciekawym miejscem wymiany doświadczeń może być bardzo aktywna grupa SixSigma w portalu GoldenLine.

Innym ciekawym przykładem zastosowaniu SixSigma jest program „The AXA way”, który bazuje na podejściu DMAIC (opisanym poniżej). Od 2002 roku, w ramach całej korporacji, służy on do stałego udoskonalania procesów oraz zwiększania jakości świadczonych usług. Również polski oddział firmy AXA oraz spółki zależne, takie jak Internet Partner Assistance (IPA), są objęte tym właśnie programem.

Six Sigma zdobyła sobie popularność szczególnie w USA dzięki wielu informacjom o wysokiej, uzyskanej wartości dodanej.

- ♦ Motorola informuje, że ośmioletnie wdrożenie przyniosło redukcję kosztów złej jakości z 40% wartości sprzedanej do 1%, co przekłada się na około 11 mld dolarów.
- ♦ Miliardowe oszczędności GE zostały skomentowane przez jej prezesa (CEO) Jacka Welscha w następujących słowach: *Six Sigma to najważniejsza z inicjatyw podjętych kiedykolwiek przez GE, jest już częścią kodu genetycznego naszej przyszłości.*
- ♦ *Six Sigma jest de facto najważniejszym elementem w całym procesie implementacji oraz zarządzania zmianami. Nie sposób uniknąć tych zmian teraz ani tym bardziej w przyszłości, ponieważ są powodowane przez ciągle zmieniający się rynek i oczekiwania naszych klientów* — Piotr Czarnecki, prezes Raiffeisen Bank Polska SA.

Wdrożenie

Każda osoba, która zajmuje się praktycznie tematyką Six Sigma, patrzy na świat z perspektywy zbioru procesów, które wykonują się w sposób powtarzalny. Każdy z tych procesów i podprocesów można udoskonalić. Głównym celem jest odnalezienie tak krytycznego zbioru procesów, że warto zainwestować w niego zastosowanie tak „ciężkiego narzędzia statystycznego”, jakim jest Six Sigma. Często w trakcie takiej analizy okazuje się, że sedno problemu tkwi w zupełnie innym miejscu, niż wydawało się pierwotnie. Przykładowo kiepska jakość obsługi telefonicznej klienta może wynikać nie z braku kompetencji osób, ale z braku właściwych procedur obsługi tegoż klienta. Wdrożenie mechanizmów IVR (ang. *Interactive Voice Response*) pozwoliłoby na lepsze sprofilowanie potrzeb osób dzwoniących i połączenie ich z pracownikiem o właściwych kompetencjach. Six Sigma w powyższym przypadku pozwoliłaby zaplanować nowy model procesów oraz oszacować potencjalne oszczędności.

Six Sigma może zostać wdrożona samodzielnie i powinna uporządkować firmowe procesy na jeden z trzech sposobów.

- ♦ Usprawnij istniejące procesy (**DMAIC**):
 - ♦ Definiuj (ang. *Defie*) cele klienta i projektu,
 - ♦ Mierz (ang. *Measure*) efektywność procesów,
 - ♦ Analizuj (ang. *Analyze*) pomiary,
 - ♦ Udoskonalaj (ang. *Improve*) efektywność procesów,
 - ♦ Kontroluj (ang. *Control*) efektywność procesów.
- ♦ Przeprojektuj istniejące procesy (**DMADV**):
 - ♦ Definiuj (ang. *Defie*) cele klienta i projektu,

- ♦ Mierz (ang. *Measure*) cele klienta i projektu,
- ♦ Analizuj (ang. *Analyze*), jak zaprojektować procesy,
- ♦ Zaprojektuj (ang. *Design*) procesy,
- ♦ Zweryfikuj (ang. *Verify*) efektywność procesów.
- ♦ Zaprojektuj nowe procesy z myślą o Six Sigma (ang. *Design For Six Sigma* — DFSS).

Oto kluczowe techniki Six Sigma.

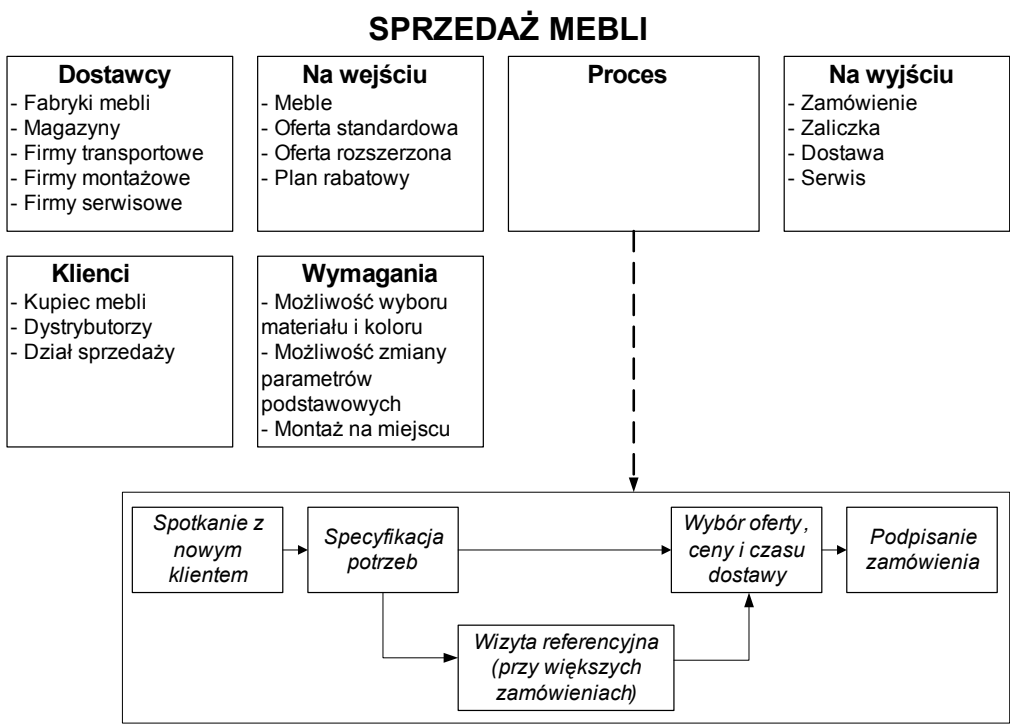
- ♦ Głos klienta (ang. *Voice of the Customer* — VOC) — stała i ścisła współpraca z klientem, szczególnie w trakcie specyfikowania wymagań. Technika ta zawiera w sobie również stały monitoring wymagań klienta po okresie wykonania analiz.
- ♦ Diagram SIPOC (ang. *Supplier, Inputs, Process, Outputs, Customers*) — szczegółowa, procesowa analiza relacji między klientem a wykonawcą składająca się z analiz:
 - ♦ dostawców,
 - ♦ produktów wejściowych,
 - ♦ procesu,
 - ♦ produktów wyjściowych,
 - ♦ klientów,
 - ♦ wymagań.

Przykładowy diagram tego typu został zaprezentowany na rysunku 64.

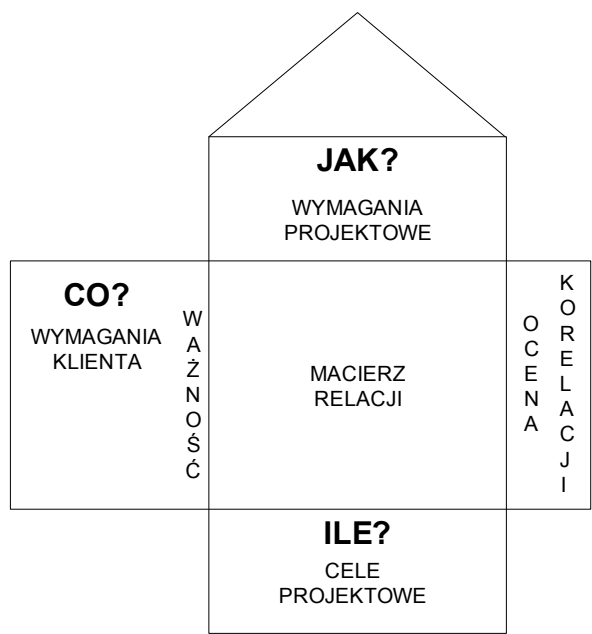
- ♦ Dom jakości (ang. *Quality Function Deployment* — QFD) — matryce przekładające wymagania klienta na grunt wykonawczy i odwrotnie, co przedstawiono na rysunku 65. Technika ta umożliwia:
 - ♦ skategoryzowanie i zbadanie głosu klienta (co?),
 - ♦ zbadanie zależności między wymaganiami projektowymi (jak?),
 - ♦ weryfikację, które elementy systemu będą najtrudniejsze (ile?),
 - ♦ weryfikację pod kątem ofert innych konkurentów.

Metoda jest bardzo atrakcyjna wizualnie, ponieważ wykorzystuje zestaw standardowych ikon do reprezentacji najistotniejszych trendów i relacji. Prosty przykład wykorzystania takiej techniki przedstawiono na rysunku 66.

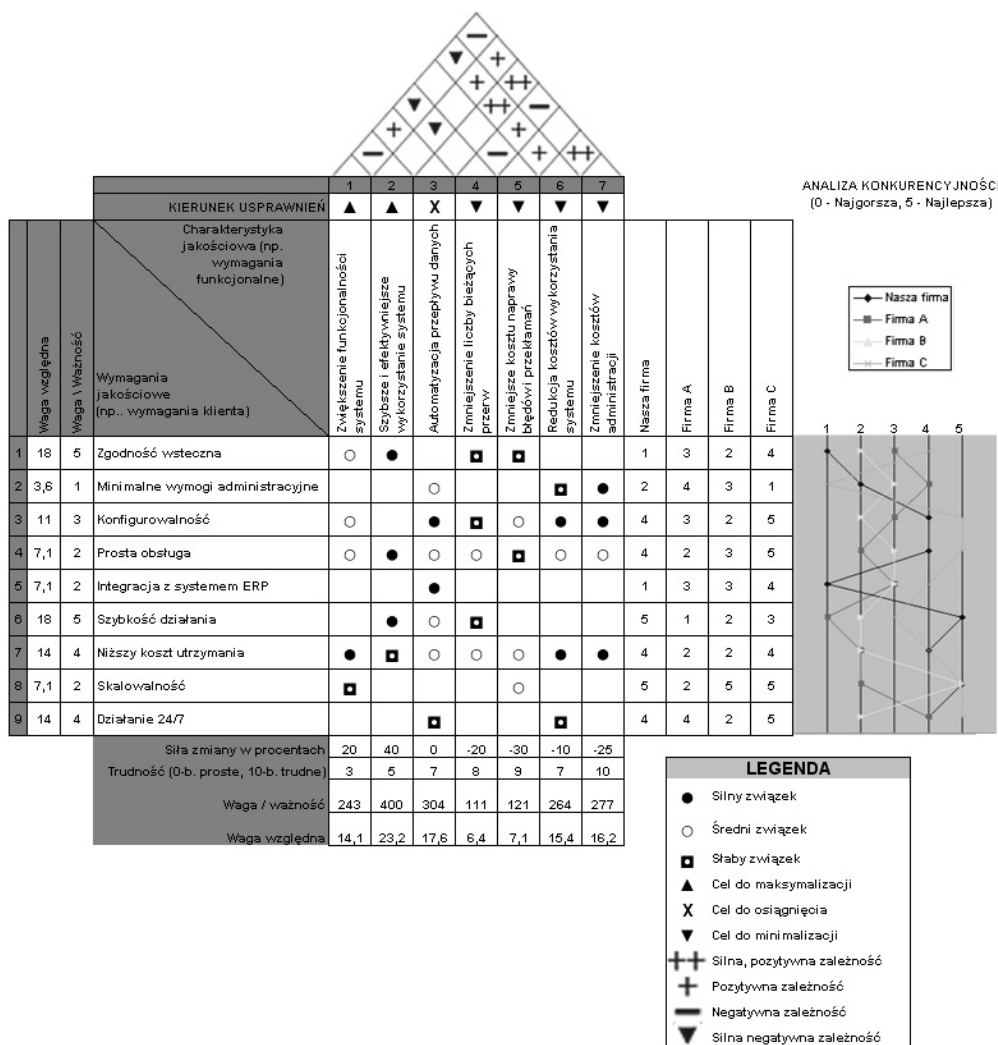
- ♦ Zarządzanie ryzykiem (ang. *Failure Mode Effect Analysis* — FMEA) — szablon analizy, który opisuje przyczynę i skutek wystąpienia potencjalnych lub istniejących wad. Zawiera takie liczbowe parametry jak:
 - ♦ Prawdopodobieństwo wystąpienia (O — ang. *Occurence*),
 - ♦ Trudność wykrycia (D — ang. *Detection*),



Rysunek 64. Przykładowy diagram SIPOC sprzedaży aut



Rysunek 65. Zasada działania „Domu jakości”



Rysunek 66. Przykładowy „Dom jakości”

- ♦ Dotkliwość dla klienta (S — ang. *Severity*).

Iloczyn tych wartości daje liczbową identyfikację ryzyk (ang. *Risk Priority Number* — $RPN = O \cdot D \cdot S$). Zbiór takich wartości umożliwia łatwą identyfikację najbardziej istotnych ryzyk i umożliwia zaplanowanie właściwych działań naprawczych.

- ♦ Planowanie eksperymentów (and. *Design of Experiments*) — technika, która polega na badaniu procesów w środowisku kontrolowanym. Oznacza to abstrahowanie od bodźców zewnętrznych (parametrów, na których nie ma się wpływu) i koncentrację na parametrach wewnętrznych, którymi można sterować. Po analizie wpływu zmian tych parametrów można dobrać taką ich konfigurację, która zwiększy niezawodność procesu w rzeczywistym środowisku produkcyjnym.

- ♦ Karty ocen (ang. *Score Card*) — raporty zawierające zestawienia kluczowych miar projektowych.
- ♦ Wykresy kontrolne — analizy stanów faktycznych i prognozowanie trendów.

Certyfikacja

Istniejące programy Six Sigma dokumentują głównie umiejętność usprawniania istniejących procesów (DMAIC).

Program Zielony Pas (ang. *Green Belt*) jest przeznaczony dla menadżerów, którzy rozpoczynają pracę z metodyką Six Sigma w kontekście konkretnych projektów. Nie jest wymagana wiedza na temat bardziej zaawansowanych narzędzi statystycznych. Zielony Pas jest podzbiorem programu Czarnego Pasa. Zdobywa się go po serii dwupółdniowych sesji dających łącznie 100 godzin szkoleniowych. Koszt takiego szkolenia w maju 2008 roku wynosił około 6 tys. złotych.

Program Czarny Pas (ang. *Black Belt*) jest przeznaczony dla osób, które będą prowadziły i nadzorowały projekty zgodne z metodyką Six Sigma. Zdobywa się go po serii czteroipółdniowych sesji dających łącznie 200 godzin szkoleniowych. Koszt takiego szkolenia w maju 2008 roku wynosił około 12 tys. złotych. Czarny Pas można zdobywać, nie posiadając Zielonego Pasa.

Zarówno w przypadku Zielonego, jak i Czarnego Pasa wymagane jest posiadanie:

- ♦ Sponsora korporacyjnego,
- ♦ Mentora w obrębie organizacji (Champion Six Sigma), który zapewni wsparcie w trakcie realizacji projektu — w praktyce, ze względu na słabe spopularyzowanie metodyki, przyjmuje się często wsparcie ze strony firmy zewnętrznej,
- ♦ Co najmniej jednego potencjalnego i zdefiniowanego problemu biznesowego.

Program Champion jest programem przeznaczonym dla członków zarządu, ścisłego kierownictwa firmy lub osoby odpowiedzialnej za wdrożenie, rozwój i wsparcie programu Six Sigma. Program polega na ogólnym zaznajomieniu się ze strategią metodyki, jej zaletami i ograniczeniami oraz kluczowymi czynnikami decydującymi o powodzeniu wdrożenia. Jest to krótkie, dwudniowe szkolenie, którego koszt to około 1,6 tys. złotych.

Podsumowanie

Nie ma ograniczeń dotyczących obszaru zastosowania metodyki Six Sigma, ale nie zawsze przyniesie ona oczekiwany skutek. Przed jej użyciem należy przeanalizować, czy nadaje się do specyfiki firmy i projektu. Six Sigma powinna zostać użyta do tematów, w których konieczne jest wykonanie solidnej analizy, a kwestią krytyczną są ryzyka

projektowe. Dobry przykład to tworzenie oprogramowania sterującego urządzeniami fizycznymi produkowanymi taśmowo w tysiącach sztuk (np. telefony komórkowe).

Jeżeli nawet upewnimy się, że Six Sigma „jest dla nas”, trzeba pamiętać, że nie ma „przymusu” osiągnięcia 6. poziomu. Zostało bowiem wykazane, że do poziomu 5. można dotrzeć drogą ewolucji, ale powyżej tej bariery konieczne będzie tworzenie nowych procesów z myślą o Six Sigma (DFSS). Warto wiedzieć, że dobra firma na rynku polskim ma procesy na poziomie 2. do 3., a międzynarodowa na 4. poziomie¹.

Six Sigma nie uzyskała zbyt dużej popularności w obszarze projektów informatycznych, co jest zapewne spowodowane trudnościami związanymi ze skonstruowaniem sensownego aparatu statystycznego bazującego na miarach. Jak mawiał Mark Twain, istnieją wszak trzy rodzaje kłamstwa: „Kłamstwo, bezczelne kłamstwo i statystyka”² ☺. Istnieją głosy, które twierdzą, iż Six Sigma nie nadaje się do projektów mających na celu wdrożenie nowego oprogramowania, ale dobrze uzupełnia takie metodyki jak PMBoK³. Prace Caspersa Jonesa oparte na sporej bazie danych empirycznych wykazują przydatność metodyki Six Sigma w fazie planowania i zbierania wymagań klienta⁴. Dla działów utrzymania jakości Six Sigma może być również bardzo dobrym argumentem do automatyzacji testów jednostkowych i funkcjonalnych; tylko w takich warunkach można stosunkowo szybko wykonać dużą liczbę badań „możliwości błędu”.

Większość firm, które z powodzeniem wdrożyły metodę Six Sigma, mówi o zdobyciu przewagi konkurencyjnej. Dla sektora IT nie jest to jednak prawda oczywista. Jeżeli tworzymy plotkarski serwis webowy, Six Sigma będzie miała najprawdopodobniej minimalny sens. Jeżeli jednak budujemy oprogramowanie dla urządzeń lotniczych, pasażerowie na pokładzie samolotu powinni poczuć się nieco lepiej, wiedząc, że oprogramowanie psuje się „jedynie” w 3,4 przypadków na milion ☺.

¹ W. Modliński, „Firma na poziomie 6., czyli mity i prawdy o metodzie Six Sigma, mit 3.”, http://www.4pm.pl/artukul/firma_na_poziomie_6_czyli_mity_i_prawdy_o_metodzie_six_sigma-26-95-2.html

² Wersja ang.: *There are three kinds of lies: lies, damned lies, and statistics*. Faktycznie jest to sentencja Benjamina Disraeli’ego rozpowszechniona przez Marka Twaina. Twain’s Autobiography (1924), wol. I, s. 246.

³ G. A. Gack, *Six Sigma and the Project Management Body of Knowledge*, <http://software.isixsigma.com/library/content/c040721b.asp>

⁴ C. Jones, *Ocena i kontrola ryzyka projektów wdrożenia oprogramowania*, 1994, s.29.

Rozdział 12.

Information Technology Infrastructure Library (ITIL)

ITIL to zbiór dobrych praktyk i procesów opisujących sposób funkcjonowania działów IT. Cechą charakterystyczną tej metodyki jest spojrzenie na wszystkie wykonywane operacje z perspektywy usług i zdefiniowanie działu IT jako ich dostawcy (ang. *service provider*). Każda z usług posiada przyporządkowany zbiór metadanych, takich jak zdefiniowany poziom świadczonych usług (SLA) czy kluczowe miary wydajności (ang. *Key Performance Indicators*). Dzięki temu można łatwo utrzymywać i monitorować stan świadczonych usług. Przykładowo w jednym raporcie można uzyskać spójną informację na temat dostępności wszystkich usług świadczonych przez dział IT. Tego typu działania wprowadzają większy stopień przejrzystości i zwiększony poziom zaufania do IT.

Szczypta historii

Prace nad tą metodyką zaczęły się w Wielkiej Brytanii, w połowie lat 80. ubiegłego wieku na zlecenie administracji rządowej. Właścicielem znaku towarowego ITIL jest OGC (ang. *Office of Government Commerce*). Pierwsza wersja ITIL pojawiła się w 1989 roku.

W 1991 roku została założona organizacja itSMF (ang. *The IT Service Management Forum*), która zrzesza osoby zainteresowane tematyką zarządzania usługami informatycznymi.

W 2000 roku opublikowano drugą wersję metodyki ITIL.

Ukoronowaniem prac rozwojowych nad ITIL było wydanie w grudniu 2005 roku międzynarodowej normy ISO/IEC 20000 Service Management. Norma ta opisuje wymagania dotyczące usług informatycznych i jest rozwinięciem normy ISO 9001. Pozwala ona na, w pełni procesowe, opisanie usług IT i jest ściśle związana z biblioteką ITIL.

Od 1 stycznia 2007 roku APM Group stał się partnerem OGC w kwestii komercyjnej popularyzacji ITIL.

Pod koniec maja 2007 roku opublikowano najnowszą, trzecią wersję ITIL, która składa się z sześciu nowych publikacji.

- ♦ **Wprowadzenie do nowej koncepcji cyklu życia usług** (ang. *The Official Introduction to the ITIL® Service Lifecycle*) to ogólny materiał wprowadzający do pozostałych publikacji.
- ♦ **Strategia usług** (ang. *Service Strategy* — SS) koncentruje się na aspektach biznesowych.
- ♦ **Projektowanie usług** (ang. *Service Design* — SD) opisuje sposób realizacji i utrzymania projektów IT.
- ♦ **Przekazanie usług** (ang. *Service Transition* — ST) koncentruje się na kwestiach wdrożeniowych, zarządzaniem zmianą, konfiguracją i wersją, uwzględniając ryzyko, korzyści i wsparcie.
- ♦ **Eksploracja usług** (ang. *Service Operation* — SO) specyfikuje mechanizmy codziennej działalności operacyjnej.
- ♦ **Ustawiczne udoskonalanie usług** (ang. *Continual Service Improvement* — CSI) wyróżnia wagę nieustannego doskonalenia procesów i usług przy równoczesnym zachowaniu spójności powtarzalnych czynności, włącznie z mechanizmem wycofywania usług z użycia.

Podczas gdy ITIL w wersji drugiej opisywała, co powinno być zrobione, aby udoskonalić proces, wersja trzecia podaje już jasno, jak tej zmiany należy dokonać. W publikacji tej o wiele bardziej dokładnie rozpisane zostały role kluczowych osób w procesie i ich odpowiedzialność. Dodatkowo działania operacyjne zostały powiązane ze strategią przedsiębiorstwa w kontekście portfela usług (ang. *service portfolio*), który pojawił się właśnie w wersji trzeciej. Pozwala to spojrzeć na IT z biznesowego punktu widzenia.

[...] z punktu widzenia CIO największą korzyścią z nowej wersji ITIL jest to, że przynosi ona zbiór rekomendacji, które pozwalają na wykazanie i pomiar zwrotu z inwestycji w ujednoliconą strukturę do zarządzania usługami. ITIL pomaga w wykazaniu, że inwestycje w dobry serwis dają solidny zwrot z inwestycji — Sharon Taylor, główny architekt ITIL¹.

¹ Antoni Bielewicz, „Outsourcing v. 3”, Computerworld nr 9/804, 26 lutego 2008.

Role

Najważniejsze role są definiowane w trzech głównych publikacjach: Strategii, Projektowaniu i Eksploatacji usług. Pozostałe dwie publikacje odnoszą się głównie do „wykorzystania istniejących zasobów”.

Strategia usług

- ♦ Menadżer relacji biznesowych (ang. *Business Relationship Manager* — BRM) to osoba odpowiedzialna za kontakty z klientami i zrozumienie biznesu klienta. Osoba reprezentująca klienta przed kierownikiem projektów w trakcie negocjacji zakresu wymagań.
- ♦ Kierownik projektów (ang. *Product Manager* — PM) to osoba odpowiedzialna za wytworzenie usługi i zarządzanie nią na przestrzeni całego cyklu życia.
- ♦ Kierownik zasobów (ang. *Chief Sourcing Officer*) to osoba kierująca pulą zasobów i tworząca strategię zarządzania tymi zasobami.

Projektowanie usług

- ♦ Menadżer projektowania usług — odpowiedzialny za koordynację i implementację projektów rozwiązań dla usług i procesów, włącznie z kwestiami jakościowymi.
- ♦ Architekt (projektant IT) — odpowiedzialny za koordynację i projektowanie wymaganych technologii, architektur, strategii, projektów i planów.
- ♦ Menadżer katalogu usług — odpowiedzialny za stworzenie i utrzymanie aktualnego katalogu usług.
- ♦ Menadżer dostępności — odpowiedzialny za zapewnienie, że wszystkie usługi są dostępne na takich zasadach jak uzgodniono.
- ♦ Menadżer ciągłości usług IT — odpowiedzialny za zapewnienie, że wszystkie serwisy mogą być odtworzone, zgodnie z przyjętymi potrzebami biznesowymi i horyzontem czasowym.
- ♦ Menadżer przepustowości — odpowiedzialny za zapewnienie przepustowości zdefiniowanej przez obecne i przyszłe wymagania biznesowe.
- ♦ Menadżer bezpieczeństwa — odpowiedzialny za zapewnienie bezpieczeństwa środowiska IT, zgodnie z przyjętymi politykami i wymaganiami.
- ♦ Menadżer poddostawców — odpowiedzialny za zapewnienie, że pieniądze wydane na kontrakty z poddostawcami przełożą się na spełnienie zdefiniowanych potrzeb biznesowych.

Eksploatacja usług

- ♦ Service desk — udostępnia jeden, centralny punkt kontaktu dla wszystkich użytkowników usług. W ramach tego kontaktu istnieje możliwość zapisu wszystkich incydentów, żądań zmian oraz dostępu. Istnieje również jeden,

centralny system umożliwiający zarządzanie tymi elementami. W tej funkcji zawarta jest również:

- ♦ kategoryzacja i priorytetyzacja zgłoszeń,
- ♦ pierwszy etap analizy i diagnozy,
- ♦ eskalacja i zamykanie zgłoszeń,
- ♦ stałe informowanie użytkowników o statusach serwisów, incydentów i zadań.

Takie centralne podejście umożliwia przetworzenie większej ilości zgłoszeń przy mniejszej ilości pracowników oraz ustandaryzowanie tego typu komunikacji.

- ♦ Zarządzanie techniczne — wszystkie osoby, które udostępniają ekspertyzy techniczne oraz zarządzają infrastrukturą IT. Funkcja ta zawiera:
 - ♦ Identyfikację wiedzy i ekspertyzę wymagań,
 - ♦ Definiowanie standardów architektonicznych,
 - ♦ Zaangażowanie w projektowanie i budowę nowych usług oraz praktyki eksploatacyjne,
 - ♦ Wkład do projektowania usług, ich przekazania oraz stałego udoskonalania,
 - ♦ Asystę przy zarządzaniu serwisami,
 - ♦ Pomoc przy definiowaniu standardów i narzędzi,
 - ♦ Asystę przy przekazywaniu zadań, tak jak ma to miejsce w trakcie przetwarzania żądania zmiany,
 - ♦ Pomoc przy zarządzaniu kontraktami i dostawcami.

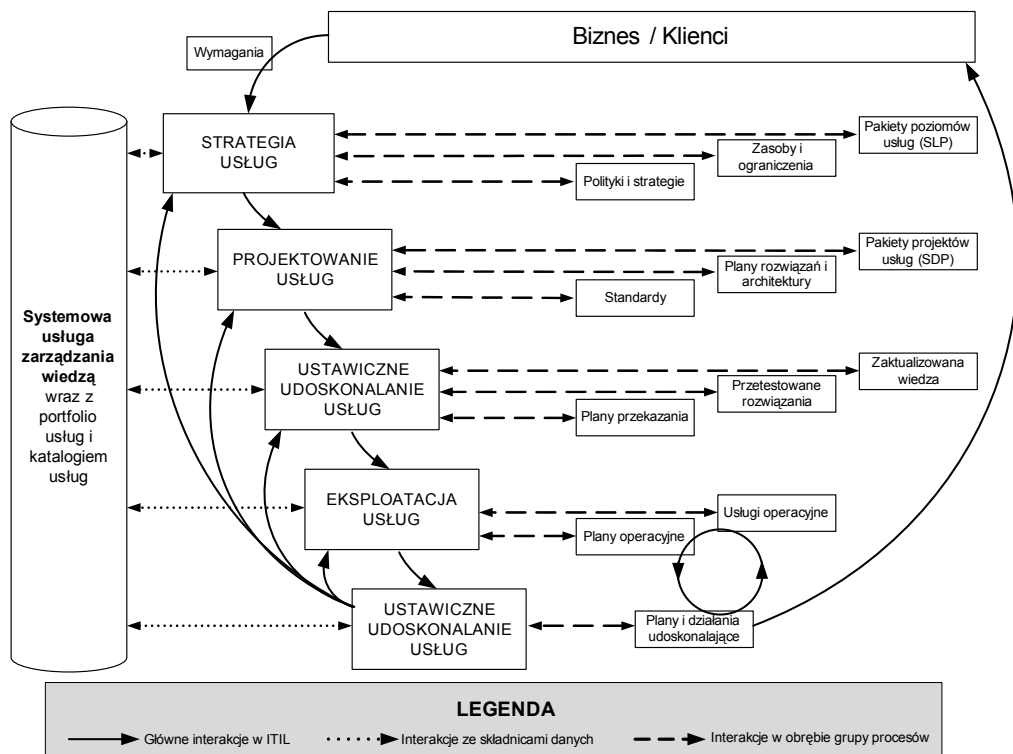
Funkcja jest zazwyczaj realizowana przez zespoły przypisane do konkretnej części infrastruktury sprzętowej.

- ♦ Zarządzanie aplikacją — wszystkie osoby, które udostępniają ekspertyzy techniczne oraz zarządzają aplikacjami. To rola bardzo podobna do zarządzania technicznego, ale zamiast zarządzania infrastrukturą sprzętową koncentruje się na oprogramowaniu, czyli aplikacjach.
- ♦ Zarządzanie IT — jest odpowiedzialne za kompleksowe zarządzanie i utrzymanie całej infrastruktury IT, która udostępnia wszystkie usługi biznesowe. Zawiera:
 - ♦ Kontrolę operacyjną zorganizowaną zazwyczaj w trybie zmianowym, gdzie operatorzy wykonują zgodnie z procedurą rutynowe zadania kontrolne i monitoring,
 - ♦ Zarządzanie usprawnieniami w centrach danych, serwerowniach i ośrodkach zapasowych. Koordynuje również większe projekty, takie jak konsolidacja centrów danych lub serwerów.

Procesy

ITIL w wersji trzeciej składa się sześciu publikacji, z których pierwsza to **Wprowadzenie do nowej koncepcji cyklu życia usług** (ang. *The Official Introduction to the ITIL® Service Lifecycle*). Jest to ogólny materiał wprowadzający do pozostałych publikacji.

Pozostałe pięć publikacji grupuje procesy (usługi) ITIL — rysunek 67 — a wiedza jest przekazywana między nimi za pomocą systemowej usługi zarządzania wiedzą (ang. *Service Knowledge Management System* — SKMS).



Rysunek 67. Architektura ITIL

Strategia usług (ang. *Service Strategy* — SS) opisuje procesy związane z wypracowaniem wizji i decyzji związanych z całym portfolio usług oferowanych przez dział IT. Pod uwagę brane są kwestie finansowe przedsięwzięcia. Publikacja opisuje mechanizmy budżetowania oraz wysokopoziomowego rozliczania z postawionych wymagań biznesowych. Opisany jest tu również pierwszy etap w cyklu życia usługi, czyli specyfikacja wymagań klienta poprzez analizę czynności biznesowych. Analiza ta zawiera również opis potencjalnych korzyści. Na tym etapie definiowana jest też wstępna skala przedsięwzięcia w pakiecie poziomu usługi (ang. *Service Level Package* — SLP). Pakiet ten zawiera:

- ♦ definicje oczekiwanego poziomu dostępności, przepustowości i bezpieczeństwa,
- ♦ oczekiwania dotyczące ciągłości działania usługi,
- ♦ funkcjonalność usługi,
- ♦ wsparcie dla usługi.

Po stworzeniu pakietu SLP weryfikowana jest dostępność zasobów. Na tej podstawie możliwe jest podjęcie strategicznej decyzji o uruchomieniu działań bądź nie.

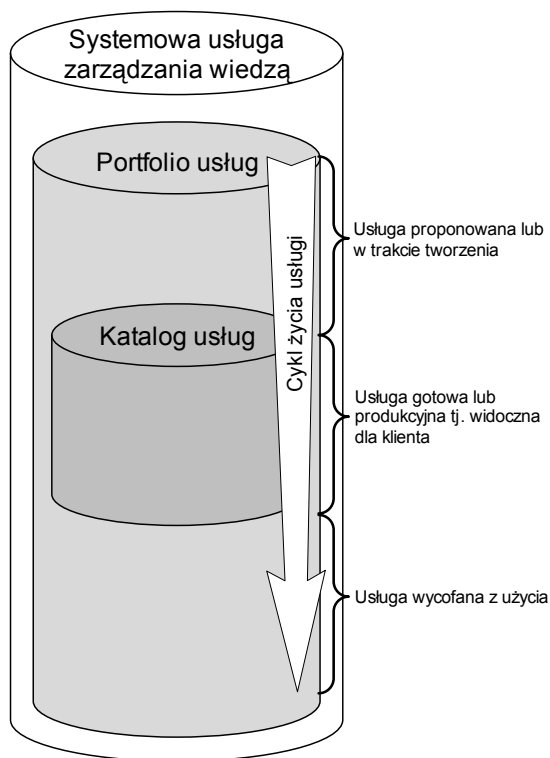
Projektowanie usług (ang. *Service Design* — SD) zajmuje się wszelkimi działaniami związanymi z przygotowaniem usług do działania produkcyjnego. Oznacza to m.in. przejście od pakietu poziomu usługi (ang. *Service Level Package* — SLP) do pakietu projektu usługi (ang. *Service Design Package* — SDP). Pakiet ten opisuje wszystko, co jest potrzebne w dalszych etapach życia usługi, czyli przy jej uruchomieniu, utrzymaniu i wycofaniu — rysunek 68. Oznacza to m.in.:

- ♦ zdefiniowanie kryteriów akceptacji serwisu,
- ♦ zdefiniowanie sposobu wkomponowania usługi do katalogu już istniejących usług,
- ♦ opisanie wymagań związanych z poziomem świadczonych usług (SLA),
- ♦ ustalenie przepustowości usługi pod kątem realnych możliwości IT,
- ♦ ustalenie mechanizmów zapewniających bezpieczny i ciągły dostęp do usługi,
- ♦ ustalenie sposobu zarządzania konfiguracją,
- ♦ wybór ewentualnych poddostawców i ustalenie zasad komunikacji z nimi.

Przekazanie usług (ang. *Service Transition* — ST) opisuje procesy związane z przekazaniem usługi do eksploatacji, czyli wdrożenie. Zanim to jednak nastąpi, sprawdzane jest, czy zaplanowana usługa spełnia postawione jej wymagania funkcjonalne, związane z przyjętymi poziomami ciągłości dostępu oraz bezpieczeństwa. Po wstępnej decyzji o uruchomieniu wdrożenia szczegółowo planowane jest przejście z systemu testowego na produkcyjny. Plany te opisują mechanizmy zarządzania zmianą, konfiguracją i publikacjami wersji usługi. Dodatkowo ustalane są metryki, które umożliwią sprawdzenie, czy usługa spełnia założone wymagania klientów. Przekazanie usług zawiera również opis procesu zarządzania wiedzą poprzez centralną systemową usługę zarządzania wiedzą (ang. *Service Knowledge Management System* — SKMS).

Eksploatacja usług (ang. *Service Operation* — SO) to procesy związane z bieżącym wykorzystaniem usługi. Pojawiają się tutaj następujące pojęcia.

- ♦ Zdarzenie — „coś”, co wydarzyło się z usługą. Może prowadzić do incydentu, problemu lub zmiany, ale niekoniecznie musi być „czymś złym” i może oznaczać np. otrzymanie SMS-a o wygenerowaniu nowego pliku logu.
- ♦ Incydent — „coś nieprzewidzianego”, co wydarzyło się z usługą. Może to „coś” nie zostało jeszcze zdefiniowane jako błąd. Dobrym przykładem jest wykrycie w usłudze wolniejszych czasów reakcji.

Rysunek 68.*Cykl życia usługi ITIL*

- ♦ Błąd — „coś złego”, co wydarzyło się z usługą.
- ♦ Żądanie — prośba od użytkownika dotycząca informacji, porady lub dostępu do usługi.

Za pomocą tych właśnie definicji procesy systematyzują komunikację między dostawcami a klientami usług poprzez „service desk”. Oprócz tych elementów, publikacja opisuje również mechanizmy bieżącego administrowania dostępem i uprawnieniami do usług.

Ustawiczne udoskonalanie usług (ang. *Continual Service Improvement* — CSI) opisuje konieczność stałego i proaktywnego udoskonalania wszystkich procesów, tam gdzie to możliwe. Na bazie wykonywanych pomiarów przeprowadzane są wewnętrzne audyty, które stale weryfikują zasadność utrzymywania usług i ich kształt. Procesy te są odpowiedzialne m.in. za wycofywanie nieużywanych usług lub ograniczanie do nich dostępu, jeżeli nie jest to zasadne z biznesowego punktu widzenia. Działy IT stale generują okresowe raporty, w których opisują:

- ♦ swój biznesowy wkład w działalność całej firmy,
- ♦ stopień realizacji zaplanowanych poziomów świadczenia usług (SLA),
- ♦ podjęte działania udoskonalające i prewencyjne.

W załączniku C — Lista wszystkich procesów ITIL — zamieszczono szczegółową listę wszystkich procesów ITIL.

Wdrożenie

Tak jak w przypadku innych metodyk, kluczową kwestią jest właściwe dobranie zakresu wdrożenia ITIL do potrzeb organizacji. Oznacza to konieczność zdefiniowania celu wprowadzenia konkretnej grupy procesów. Przykładowo może on zostać określony poprzez rozesłanie kwestionariuszy do kluczowych klientów oraz pracowników firmy. Następnie konieczne jest dokonanie analizy kosztów takiego wdrożenia względem planowanych zysków. Aby to wykonać, konieczne jest liczbowe oszacowanie założeń i celów w określonych obszarach. Poniżej podano trzy przykłady analizy kosztów i zysków dla konkretnych procesów ITIL.

Proces: Zarządzanie konfiguracją

Cel: Kontrola środowiska IT oraz zapewnienie, że użytkowany jest tylko autoryzowany hardware i software

Zysk: Liczba osób dedykowana do analizy incydentów może być zredukowana z trzech do dwóch. W przypadku osoby zarabiającej 6000 zł brutto oznacza to oszczędności rzędu 72 tys. zł rocznie.

Proces: Zarządzanie incydentami

Cel: Zapewnienie stałego poziomu świadczonych usług

Zysk: Wprowadzenie tego procesu oznacza redukcję przestoju użytkownika. Jeżeli założymy, że:

- ♦ każdy użytkownik oszczędzi 1 minutę dziennie,
- ♦ mamy 20 dni pracujących w miesiącu,
- ♦ organizacja ma 100 użytkowników,
- ♦ średnia pensja to 6000 zł brutto,

oszczędność wyniesie $(1/480 * 20 * 100 * 6000) * 12 = 300$ tys. zł rocznie.

Proces: Zarządzanie ciągłością usług IT

Cel: Zapewnienie szybkiego czasu odtworzenia po katastrofie

Zysk: Zalało serwerownię i mamy dwa dni konieczne do osiągnięcia pełnej operacyjności. Jeżeli założymy, że:

- ♦ brak planu ciągłości usług,
- ♦ średnio każdy użytkownik straci 10 godzin pracy,
- ♦ mamy 20 dni pracujących w miesiącu,
- ♦ organizacja ma 100 użytkowników,
- ♦ średnia pensja to 6000 zł brutto,

zapłacimy $1,5 * 100 * 6000 = 900$ tys. zł za każdą tego typu awarię.

Źródło: OGC, ITIL Implementaion

Po zdefiniowaniu celu wdrożenia konieczne jest określenie jego zakresu. Jeżeli np. podjęta została decyzja o wdrożeniu rozwiązań „service desk”, nie będzie to możliwe bez zdefiniowania katalogu usług; warto również podjąć trud wdrożenia ujednoliconej bazy danych zarządzania konfiguracjami (ang. *Configuration Management Database*). Rozwiązanie to pozwoli na przechowywanie danych o konfiguracji komponentów środowiska IT (ang. *Configuration Item*), a szczególnie zależności pomiędzy nimi. Publikacje ITIL dostarczają wytyczne i dobre praktyki związane z budową CMDB. Przy wdrażaniu CMDB należy jednak liczyć się z koniecznością migracji danych ze starych struktur do nowych i niejednokrotnie może to być poważne przedsięwzięcie. Utrzymanie CMDB również wiąże się z pewnymi, dodatkowymi kosztami.

ITIL umożliwia stworzenie procedur komunikacyjnych, które zapewnią użytkownikom IT jeden punkt kontaktu (ang. *Single Point Of Contact*). Jeżeli IT jest przeciążone liczbą zgłoszeń od użytkowników, warto zastanowić się nad rozwiązaniami samoobsługowymi (ang. *self service*). Dzięki nim użytkownicy mogą sami rozwiązywać swoje problemy; uzyskamy to szczególnie wtedy, kiedy mamy do czynienia z dużą grupą użytkowników, którzy mogą zadawać te same pytania. Dobrym rozwiązaniem będzie tutaj udostępnienie bazy wiedzy wraz z odpowiednimi mechanizmami wyszukiwania. Dobrym przykładem jest serwis MSDN oferowany przez firmę Microsoft.

Kluczowym elementem każdego wdrożenia jest również zaangażowanie interesariuszy. W trakcie uruchamiania projektu warto przygotować krótkie szkolenia na temat metodyki oraz zaprezentować cele całego projektu. Przykładowy harmonogram wdrożenia przedstawiono w tabeli 23.

Tabela 23. Przykładowy harmonogram wdrożenia narzędzia wspierającego procesy ITIL

Tydzień	1	2	3	4	5	6	7	8	9	10	11	12
Organizacja projektu	X											
Szkolenia z zakresu ITIL		X	X									
Dostarczenie licencji i instalacja systemu na sprzęcie dostarczonym przez zamawiającego			X									
Opracowanie katalogu usług i SLA				X	X	X						
Definicja struktury i parametrów CMDB (ang. <i>Configuration Management Database</i>)					X	X						
Konfiguracja systemu				X	X	X	X					
Migracja danych							X	X				
Testy funkcjonalne i wydajnościowe									X	X		
Szkolenie użytkowników										X	X	
Start systemu												X

Wbrew stereotypom, ITIL można również wdrożyć w małych organizacjach — wdrożenie nie musi trwać dwa lata ☺. OCG udostępnia bardzo ciekawą publikację — „ITIL V3 Small-Scale Implementation Book” — która w tego typu przypadkach ułatwia właściwe dobranie grup procesów oraz technik.

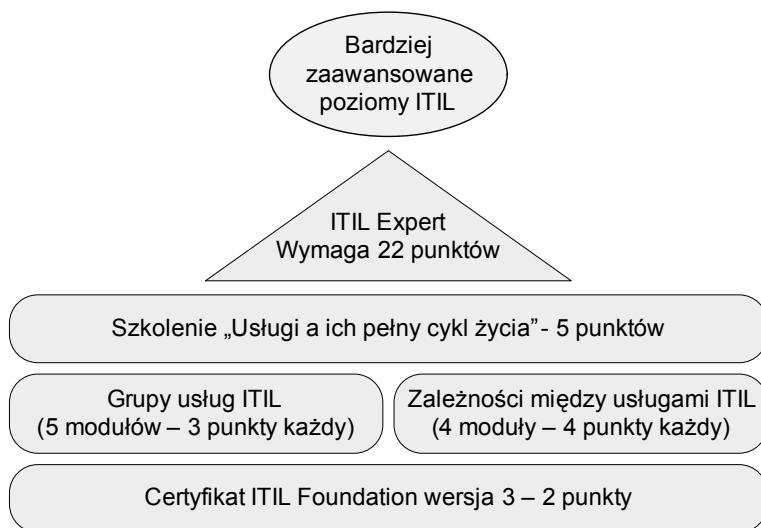
Ciekawym uzupełnieniem metodyki ITIL w dziedzinie zarządzania ryzykiem w organizacji jest inna publikacja OGC — „Management of Risk” (M_o_R). Jako ryzyko rozumie się tutaj również okazję biznesową i tego typu kwestie są zarządzane zarówno na poziomie strategicznym, jak i operacyjnym. M_o_R opisuje:

- ♦ podstawowe reguły zarządzania ryzykiem (ang. *principles*),
- ♦ mechanizmy zaadaptowania do potrzeb konkretnej organizacji (ang. *approach*),
- ♦ sześć modelowych procesów związanych z zarządzaniem ryzykiem (ang. *processes*),
- ♦ mechanizmy zapewnienia, że zarządzanie ryzykiem jest efektywnie wykorzystywane w obrębie organizacji, i stały przegląd rejestrów ryzyk (ang. *embedding and reviewing*).

Certyfikacja

Organizacja itSMF posiada obecnie 5 tys. członków instytucjonalnych (firm) oraz ponad 70 tys. osób w ponad 53 oddziałach w poszczególnych krajach, a w tym i w Polsce.

Wraz z 3. wersją ITIL została diametralnie zmieniona struktura certyfikacji, m.in. zniknęły poziomy ITIL Practitioner i ITIL Service Manager, znane z wersji 2. Zostały one zastąpione przez ITIL Expert. W ramach ITIL w wersji 3. został wdrożony system certyfikacji, taki jak przedstawiono na rysunku 69².



Rysunek 69. System certyfikacji ITIL w 3. wersji

² ITIL version3 Qualifications Explained, John F McDermott http://h41156.www4.hp.com/education/upload/uk/en/ITILV3_qualifications_explained.pdf

W ramach tego programu istnieją dwa poziomy certyfikacji.

- ♦ **ITIL® V3 Foundation** — poziom wprowadzający i weryfikujący podstawową wiedzę w zakresie metodyki. Koszt odbycia takiego egzaminu w systemie Prometric to około 330 zł (150 USD). Istnieje wiele firm oferujących 3 – 4 dniowe szkolenia przygotowujące do tego egzaminu, takich jak CTPartners (grupa kapitałowa Infovide-Matrix), HP lub IBM. Można jednak podejść do tego egzaminu, korzystając wyłącznie z akredytowanych materiałów — nie ma formalnego wymogu odbywania szkoleń.
- ♦ **ITIL® Expert** — jest odpowiednikiem certyfikatu Service Manager w wersji 2. Do jego uzyskania konieczne jest zdobycie 22 punktów, np. poprzez zdobycie certyfikatu Foundation (2 pkt.), ukończenie pięciu kursów z zarządzania w każdym z etapów cyklu życia usługi ($5 \times 3 = 15$ pkt.) i 1 dodatkowego, sumarycznego kursu traktującego o całym cyklu (5 pkt.). Ten dość mozolny proces ma swoją alternatywę w postaci uzyskania certyfikatów w wersji 2. W skrajnym przypadku wystarczy posiadanie certyfikatów Foundation w wersji 2., Manager w wersji 2. i odbycie kursu Managers Bridge w wersji 3. Jest to jednak rozwiązanie o charakterze przejściowym.

Podsumowanie

Bardzo często metodyka ITIL jest niesłusznie identyfikowana wyłącznie z procesami „service desk”. W praktyce jest to metodyka ułatwiająca uporządkowanie całego IT, a w tym i komunikacji z użytkownikami, decydentami oraz innymi interesariuszami. ITIL posiada bardzo ciekawe, usługowe podejście do wszystkich działań IT. Dzięki temu możliwe jest stworzenie portfolio usług i liczbowe określenie poziomów ich świadczenia (ang. *Service Level Management*). Przykładowo w ramach organizacji można określić grupę usług krytycznych, których brak spowodowałby wymierne straty finansowe (np. serwer pocztowy). Takie podejście pozwala o wiele łatwiej uzasadnić konieczność inwestycji w bezproblemowe dostarczanie takich usług.



W praktyce

Metodyka ITIL, ze względu na powinowactwo z normami ISO, jest bardzo popularna w wielu firmach. Przykładem będą takie firmy jak Allianz Polska, Philips Lighting Poland SA czy KGHM Polska Miedź. Każda udana implementacja ITIL to kompleksowo zarządzane portfolio usług IT, takich jak serwisowanie sprzętu. Tego typu strukturalna zmiana jest dla większości organizacji dość bolesna i konieczne jest wytyczenie jasnej mapy drogowej planowanych działań. W efekcie ITIL może być bardzo pomocna przy regulacji i standaryzacji bieżącej komunikacji między IT a biznesem.

Dzięki wprowadzeniu w 2004 roku ITIL pracownik odbierający telefony ma pod ręką spis podobnych zdarzeń z przeszłości wraz z informacjami o tym, jak sobie z nimi poradono. Jeśli problem nie zostanie rozwiązany w ciągu 10 minut, system automatycznie

wysyła wiadomość na pager właściwego eksperta. Daje mu się godzinę; jeśli w tym czasie nie zdoła uporać się z zaistniałą sytuacją, do akcji wkracza — poprzez telekonferencję — kadra zarządzająca działu IT, łącznie z samym Kumarem. Po dwóch godzinach Kumar zwołuje szefów działów biznesowych, by przedyskutować z nimi skutki awarii i sposoby ograniczenia jej negatywnego wpływu na pracę przedsiębiorstwa. Odkąd rok temu zrestrukturyzowano biuro pomocy technicznej zgodnie z wymaganiami ITIL, czas reakcji na problemy techniczne udało się skrócić o 50%. Co więcej, skoro powtarzające się incydenty są monitorowane oraz odnotowywane, personel informatyczny może łatwo dostrzegać tendencje oraz analizować i eliminować przyczyny problemów, które dawniej miały chroniczny charakter. Podobne procesy stosowane przy zmianach systemów czy instalacji nowych wersji oprogramowania sprawiają, że wielu możliwym konfliktom udaje się zawczasu zapobiec. — Jim McGrane, CIO w MeadWestvaco³.

Rozmowa z...

Remigiusz Swirkaitis jest kierownikiem do spraw jakości. Pracował w wielu firmach IT, takich jak CSBI, Computerland, Emax, a następnie Sygnity. Pełnił kluczową rolę w trakcie wielu wdrożeń ISO, CMM, Prince2 i ITIL w dużych firmach z sektora publicznego i prywatnego. Brał też udział w dużych projektach komercyjnych związanych z wdrożeniem oprogramowania dla takich instytucji jak Ministerstwo Pracy i Polityki Społecznej (Projekt Puls, Syriusz) oraz Izba Celną (projekt Bachus).

Adam Koszłajda (AKO): Remigiuszu, z tego co wiem, byłeś ostatnio „zamieszany” we wdrożenie metodyki ITIL w firmie z sektora energetycznego. Jakie były okoliczności tego projektu?

Remigiusz Swirkaitis (RSW): W projekcie byłem szefem zespołu zadaniowego odpowiedzialnego za wdrożenie tych usług na zlecenie firmy Trust&Gray. Wdrożenie tej metodologii ITIL pojawiło się jako jeden z podprojektów związanych z restrukturyzacją procesów w firmie zatrudniającej ponad tysiąc pracowników. Bezpośrednio w usługi serwisowe związane z dostępnością, pojemnością, poziomem usług oraz zmianą było zaangażowane około pięćdziesiąt osób.

AKO: Dlaczego wybraliście akurat ITIL?

RSW: ITIL najbardziej pasował do specyfiki firmy, która miała duże grono dostawców zewnętrznych i wewnętrznych.

AKO: A czy rozpatrywaliście jakieś inne opcje, takie jak COBIT?

³ Artykuł „Siła ITIL-u”, CIO 19 grudnia 2006. Na podstawie amerykańskiego wydania CIO przygotował Adam Maciejewski.

RSW: COBIT był rozpatrywany w czasie audytu wstępnego prowadzonego przez firmę Gordion Consulting. Ostatecznie zdecydowano się na ITIL ze względów organizacyjnych, bo pojęcie usług IT obejmowało bardzo szeroki zakres:

- ♦ automatykę — procesy technologiczne od strony usług IT,
- ♦ wewnętrzne usługi IT — procesy na rzecz wewnętrznych użytkowników sieci i systemów wewnętrznych,
- ♦ usługi telekomunikacyjne — wewnętrzna sieć telekomunikacyjna oraz usługi świadczone dla abonentów w okolicznych gminach.

AKO: Rozumiem, że głównym zadaniem było uporządkowanie zakresu usług, jakie świadczy firma, i właściwe nimi zarządzanie poprzez zawarcie wewnętrznych kontraktów typu SLA?

RSW: SLA było jednym z zagadnień. Kluczową kwestią było jednak zdiagnozowanie stanu usług poprzez wprowadzenie właściwego zarządzania incydentami, problemami, konfiguracją, zmianą i wersją. Wszystkie te działania należało — oczywiście — rozpatrywać w kontekście trzech głównych nurtów działalności usługowej IT, które wymieniałem wcześniej. Bardzo ciekawym zastosowaniem metodologii ITIL było wypracowanie „Planu Kontynuacji Biznesu”.

AKO: A cóż to takiego?

RSW: To dokument wynikający z „Zarządzania ciągłością usług IT”. W odniesieniu do świadczenia usług krytycznych został opracowany plan przywrócenia działania procesów biznesowych i technologicznych w przypadku katastrofy lub poważnej awarii. Takie plany zostały opracowane w trzech obszarach, które wymieniałem na początku, czyli dla usług telekomunikacyjnych, awarii wewnętrznych systemów IT oraz działania podstawowych procesów technologicznych sterowanych przez systemy informatyczne.

AKO: Czy w tych trzech głównych nurtach wdrożyliście wszystkie grupy procesów związane z strategią usług, projektowanie, przekazaniem, eksploatacją, ustawicznym udoskonalaniem usług?

RSW: Uruchomione zostały głównie procesy związane z przekazaniem, projektowaniem, obsługą i eksploatacją. Ze względu na wczesny okres projektu ustawiczne doskonalenie usług zostało przesunięte na kolejny etap wdrożenia po „okrępieńciu” rozwiązania.

AKO: Czy były jakieś konkretne narzędzia, które wykorzystywaliście w trakcie implementacji?

RSW: Nasze działania toczyły się równolegle z wdrożeniem systemu SAP. Dodatkowo wykorzystywaliśmy narzędzie HDEF dostarczane przez spółkę Trust&Gray i dedykowane właśnie do implementacji podejścia ITIL. Za jego pomocą mogliśmy modelować procesy oraz zliczać parametry wymagane przez metodykę, takie jak dostępność, czas reakcji oraz badanie parametrów logicznych związanych z niezaprzeczalnością poszczególnych funkcji w procesie. HDEF posiada narzędzie diagnostyczne badające składnie parametrów logicznych.

AKO: Jak wiele osób korzystało z tego narzędzia?

RSW: W trakcie wdrożenia byli zaangażowani kluczowi liderzy zespołów (kierownicy wydziałów), którzy za pomocą konsultantów modelowali procesy. Co ciekawe, w czasie optymalizacji procesów wyszło na jaw, że wiele było redundantnych. Poza tym, okazało się, że zliczone czasy reakcji, potrzebne na realizację poszczególnych czynności, były wysoce niezadowolające. Jedynym sensownym lekarstwem na tę bolączkę był centralny help desk.

AKO: A jak wyglądał cały proces wdrożenia? Jak długo trwał cały projekt?

RSW: Zaczęliśmy od audytu wewnętrznych procesów, następnie ustanowiliśmy zespoły zadaniowe w zakresie wdrożenia metodyki, uruchomiliśmy szkolenia z zakresu ITIL, szkolenia z narzędzi do modelowania procesów, zamodelowaliśmy procesy, uzyskaliśmy akceptację zarządu i wdrożyliśmy system. Całość zajęła około ośmiu miesięcy. Na marginesie dodam, że w międzyczasie zmienił się zarząd, co wpłynęło na pewną zmianę pierwotnej wizji.

AKO: Jakie są kluczowe czynniki powodzenia takiego wdrożenia?

RSW: Zaangażowanie wyższego kierownictwa ☺. Co ciekawe, w tym przypadku postawiono nam pełną niezależność i mieliśmy dostęp do absolutnie wszelkich zasobów, co przesądziło o skuteczności wdrożonego modelu. Dodatkowo narzędzie HDEF dawało możliwość monitorowania online reakcji na wprowadzane zmiany.

AKO: Jaka była największa porażka i sukces tego wdrożenia?

RSW: Niestety, z przyczyn od nas niezależnych nie można było tego powiązać z zarządzaniem finansowym, czyli przewidywaniem poziomu wydatków potrzebnych do uruchomienia usług IT na dany okres, zapewnieniem beneficjentów systemu, że wydatki aktualne mogą być porównywane z planowanymi i brakiem możliwości obliczania kosztów dostarczanych usług. Bardzo interesująca była kwestia budowania dostępności usług — integracja trzech platform IT w kontekście wspólnego help desk. Przy tak rozproszonych usługach, które są dedykowane zarówno dla klienta wewnętrznego, jak i zewnętrznego, zbudowanie wspólnej platformy było dużym wyzwaniem.

AKO: Czy uważasz, że w tym przypadku ITIL się sprawdził?

RSW: Jestem zafascynowany metodyką ITIL ☺. Być może dlatego, że wcześniej zajmowałem się wdrażaniem CMM, ISO i Prince w dużych korporacjach. ITIL niesamowicie strukturyzuje temat wdrożenia procesów w IT, obnaża też pewne nadbudowy i powielanie się usług.

AKO: Czyli ITL jest dobry na wszystko ☺ ?

RSW: ITIL nie jest remedium na wszystkie problemy IT. To podobny problem, z jakim zmagają się konsultanci w trakcie wdrożeń CMM i ISO. Uważam, że należy unikać rozdrabniania się na szczegóły i koniecznie trzeba skupić się na krytycznych problemach biznesowych firmy klienta. Co ciekawe, im bardziej są one krytyczne, tym większa redundancja procesów w różnych systemach. Można powiedzieć, że jestem wyznawcą zasady brzytwy Ochama.

AKO: Masz w swoim domu na ścianie jakąś maczetę?

RSW: Tak, z napisem „Non sunt multiplicanda entia sine necessitate”.

AKO: To znaczy?

RSW: Bytów nie mnożyć, fikcji nie tworzyć, tłumaczyć fakty jak najprościej. William Ocham miał prostą zasadę — jak studenci czytali książki, kazał wycinać strony niezrozumiałe i nic niewnoszące do toku myślenia.

AKO: Pytanie, czy ta strona się ostanie? ☺

Rozdział 13.

Control Objectives for Information and related Technology (COBIT)

COBIT to metodyka, która wspiera zarządzanie, kontrolę i audyt systemów informatycznych. W tym celu dostarcza zbiory miar, wskaźników, procesów oraz technik.

Metodyka składa się z czterech głównych domen. Oto one.

- ◆ Planowanie i organizacja (ang. *Plan and Organize* PO) — domena zajmuje się tematyką obiegu informacji oraz niezbędnych do tego technologii, z wyróżnieniem wymagań organizacyjnych i infrastrukturalnych w stosunku do IT.
- ◆ Nabycie i implementacja (ang. *Acquire and Implement* AI) — domena zajmuje się identyfikacją wymagań IT, zdobyciem niezbędnych do tego technologii, implementacją rozwiązań i opracowaniem planu ich utrzymania.
- ◆ Dostarczenie i wsparcie (ang. *Deliver and Support* DS) — domena zajmuje się kwestią dostarczania rozwiązań oraz ich utrzymania i wsparcia, włącznie z kwestiami bezpieczeństwa i szkoleń.
- ◆ Monitoring i ewaluacja (ang. *Monitor and Evaluate* ME) — domena, w której zawarto mechanizmy kontrolne istniejącej infrastruktury IT; mają one zapewnić, że istniejące rozwiązania służą spełnieniu celów firmy. Dodatkowo domena ta zajmuje się zapewnieniem zgodności ze strategią firmy w ramach wewnętrznych i zewnętrznych audytów.

Kluczowe produkty COBIT 4.1 to:

- ◆ Streszczenie dla decydentów na temat ładu korporacyjnego (ang. *Board Briefing on IT Governance, 2nd Edition*) pozwala zrozumieć decydentom, jaka jest ich odpowiedzialność za ład korporacyjny i dlaczego jest on istotny.

- ♦ Wytyczne zarządcze (ang. *Management guidelines/maturity models*) pomagają ustalić zakres odpowiedzialności, miary wydajnościowe oraz niedociągnięcia organizacyjne.
- ♦ Środowisko COBIT (ang. *Frameworks*) opisuje, jak procesy IT dostarczają biznesowi konieczne informacje poprzez trzydzieści cztery wysokopoziomowe cele kontrolne (po jednym dla każdego procesu). Struktura opisuje siedem kryteriów informacyjnych (efektywność, wydajność, poufność, integralność, dostępność, kompletność i solidność) oraz kluczowe zasoby IT (ludzie, aplikacje, informacje i infrastruktura).
- ♦ Cele kontrole (ang. *Control objectives*) to szczegółowy opis dwustu czterestu celów kontrolnych, które umożliwiają stworzenie przejrzystej polityki i dobrych praktyk kontroli IT, nawet w przypadku zmieniającego się środowiska.
- ♦ Poradnik wdrożeniowy (ang. *IT Governance Implementation Guide: Using COBIT® and Val IT™, 2nd Edition*) zawiera ogólną mapę drogową wdrożenia ładu korporacyjnego z wykorzystaniem COBIT i Val IT.
- ♦ Praktyki kontrolne (ang. *COBIT® Control Practices: Guidance to Achieve Control Objectives for Successful IT Governance, 2nd Edition*) zawierają wytyczne związane ze sposobem wykonania kontroli oraz informacjami, dlaczego warto je przeprowadzać. Kontrole są definiowane jako polityki, procedury, praktyki i inne mechanizmy organizacyjne. Mają na celu zapewnienie osiągnięcia celów biznesowych i zapobieganie niepożądanym zdarzeniom lub ich wykrywanie i naprawianie.
- ♦ Przewodnik zapewnienia usług IT (ang. *IT Assurance Guide: Using COBIT*) to mechanizmy audytu działań wraz z sugestiami kroków testowych dla wszystkich procesów IT i celów kontrolnych.

Przy implementacji metodyki szczególnie istotny jest dokument na temat środowiska COBIT oraz dwie ostatnie pozycje. Niestety, wersja 4.1 tych dokumentów dostępna jest jedynie dla członków ISACA, co wiąże się z koniecznością zainwestowania 40 dolarów w warszawskim oddziale ISACA. Studenci i emeryci są zwolnieni z tej opłaty.

Szczypta historii

COBIT został stworzony przez organizację ISACA (ang. *Information Systems Audit and Control Association*) oraz ITGI (ang. *IT Governance Institute*). Pierwsza edycja miała miejsce w 1996 roku, a w grudniu roku 2006 opublikowano wersję 4., która w maju 2007 roku została zaktualizowana do wersji 4.1. Ostatnie zmiany w standardzie zawierają takie „nowinki” jak:

- ♦ wzajemne referencje wejść i wyjść pomiędzy procesami,
- ♦ diagramy opisujące, w jakich procesach powinni brać udział kluczowi decydenci w firmie,
- ♦ wsparcie dla modelu dojrzałości,

- ♦ uproszczone opisy celów,
- ♦ relacje między elementami biznesowymi a celami IT i procesami.

W roku 2009 planowane jest oficjalne polskie tłumaczenie metodyki COBIT w wersji 4.1.

Ostatnia wersja dokumentacji COBIT jest bardzo dojrzałym i atrakcyjnym wizualnie kompendium wiedzy, które staje się ciekawym narzędziem służącym do wprowadzenia ładu korporacyjnego (ang. *IT Governance*) i zarządzania ryzykiem w obrębie organizacji.

W 2002 roku, w reakcji na skandale giełdowe z udziałem takich amerykańskich firm jak Enron lub WorldCom, w USA podpisano akt Sarbanes-Oxley, a w Europie Basel II. Akty obligowały firmy giełdowe do większej przejrzystości prowadzonej księgowości. Rozwiązania takie dopingują te firmy do stosowania takich metodyk jak COBIT. Namacalnym tego przykładem jest podręcznik kontroli wykonania zadań opublikowany przez Europejski Trybunał Obrachunkowy (ETO) w 2007 roku. W celu wewnętrznej kontroli zarządzania finansami sugeruje on zastosowanie modelu COBIT.

W Polsce NBP opublikował w 2002 roku rekomendację D dotyczącą zarządzania ryzykami towarzyszącymi systemom informatycznym i telekomunikacyjnym używanym przez banki. Dokument ten opisuje m.in. konieczność wykonywania regularnych, niezależnych audytów informatycznych opracowywanych na bazie uznanych standardów ISACA, Insitute of Internal Auditors Inc., COSO (ang. *Committee of Sponsoring Organizations of the Trade Commission*). W grupie tej wymieniany jest również COBIT.

Innym ciekawym zastosowaniem tej metodyki jest nadzór nad systemami informatycznymi pod kątem zgodności z ustawą o ochronie danych osobowych. W ramach dokumentu „UODO Survival Kit”, dostępnego na polskiej stronie ISACA, wykonane zostało m.in. mapowanie celów kontrolnych COBIT na poszczególne artykuły tejże ustawy. Coraz częściej zdarza się również, że instytucje zajmujące się wypłatami unijnymi wymagają od polskich agencji rządowych wykonania audytów z wykorzystaniem takich metodyk jak COBIT lub ValIT.

W firmach prywatnych COBIT umożliwia uporządkowanie procesów IT i precyzyjne zarządzanie ryzykami. Jako że podstawowa specyfikacja COBIT jest darmowa, warto porównać jej techniki z tym, co proponuje ITIL. Należy jednak wyraźnie zaznaczyć, że obie metodyki nie są dla siebie konkurencyjne. O ile ITIL koncentruje się na właściwym zarządzaniu usługami IT, COBIT skupia się głównie na kwestiach związanych z ładem korporacyjnym.

Poniższy opis bazuje na publikacji COBIT 4.0.

Role

COBIT wyróżnia następujące role:

- ♦ dyrektor wykonawczy (CEO),
- ♦ dyrektor finansowy (CFO),

- ♦ dyrektor informacyjny (CIO),
- ♦ decydent biznesowy,
- ♦ właściciel procesu biznesowego,
- ♦ kierownik operacyjny,
- ♦ główny architekt,
- ♦ kierownik implementacyjny,
- ♦ kierownik administracji IT,
- ♦ kierownik projektu,
- ♦ pracownicy odpowiedzialni za audyty, ryzyka, bezpieczeństwo, zgodność ze standardami i inne osoby nieobarczone bezpośrednią odpowiedzialnością operacyjną IT.

Procesy

Opis każdego z procesów COBIT jest w pełni ustandaryzowany i składa się z czterech bloków. Są to wysokopoziomowe cele kontrolne, szczegółowe cele kontrolne, wytyczne zarządcze i opis wymagań do procesów na poszczególnych poziomach dojrzałości.

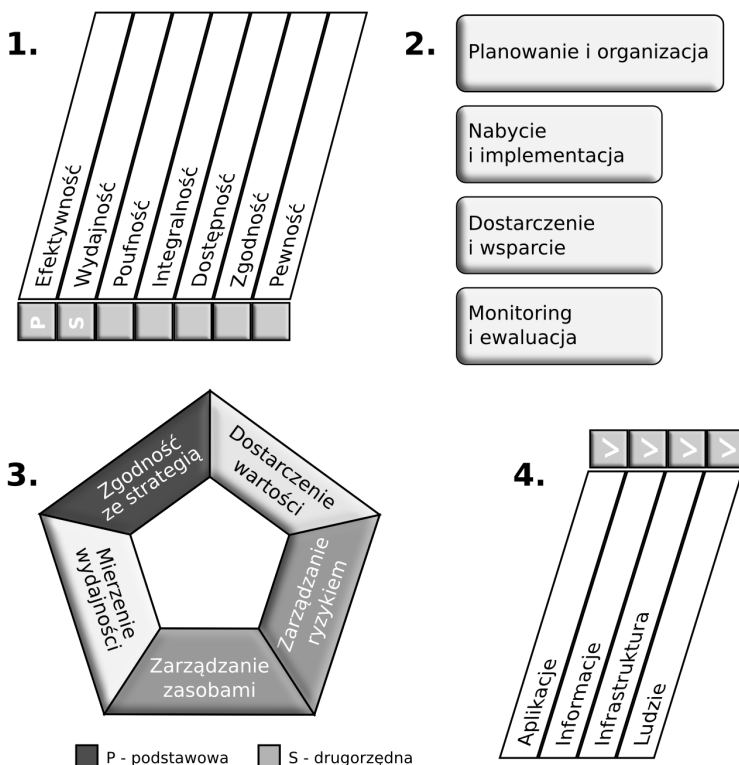
Wysokopoziomowe cele kontrolne opisują generalny cel procesu, czyli m.in. jakie spełnia potrzeby biznesowe i w jaki sposób. Charakterystyczne dla COBIT są tutaj cztery składowe, z których każda jest reprezentowana przez inny element graficzny (rysunek 70).

- ♦ Kluczowe kryteria (1) — w określonej puli kryteriów (efektywność, wydajność, poufność, integralność, dostępność, zgodność, pewność) definiowane jest, które kryterium jest pierwszorzędne (ang. *primary* — P), a które drugorzędne (ang. *secondary* — S) w kontekście konkretnego procesu. Dla procesu „PO1. Wyznaczanie planu strategicznego IT” jest to np. efektywność (P) i wydajność (S).
- ♦ Przynależność do domeny (2) — ustalenie, w której z trzech domen się znajduje. I tak „PO1. Wyznaczanie planu strategicznego IT” należy do domeny „Planowania i organizacji”.
- ♦ Kluczowe aspekty ładu korporacyjnego (3) — w określonej puli aspektów ładu korporacyjnego (dostarczenie wartości, zarządzanie ryzykiem, zarządzanie zasobami, mierzenie wydajności, zgodność ze strategią) definiowane jest, które kryterium jest pierwszorzędne (P), a które drugorzędne (S) w kontekście konkretnego procesu. W przypadku „PO1. Wyznaczanie planu strategicznego IT” jest to zgodność ze strategią (P), zarządzanie ryzykiem (S) i zarządzanie zasobami (S).

Rysunek 70.

Wysokopoziomowe cele kontrolne procesu COBIT „PO1.

Wyznaczanie planu strategicznego IT”



- ♦ Typ angażowanych zasobów (4) — wyróżnia typ angażowanych przez proces zasobów. Proces „PO1. Wyznaczanie planu strategicznego IT” angażuje wszystkie możliwe zasoby, czyli aplikacje, informacje, infrastrukturę i ludzi, ale już kolejny proces — „PO2. Definiowanie architektury informatycznej” — angażuje tylko aplikacje i informacje.

Szczegółowe cele kontrolne „rozpisują” jeden, główny cel wysokopoziomowy na poszczególne cele niższego poziomu. Dla procesu „PO1. Wyznaczanie planu strategicznego IT” są to np.:

- ♦ PO1.1. Zarządzanie wartością IT (ang. *IT Value Management*),
- ♦ PO1.2. Koordynacja IT z biznesem (ang. *Business-IT Alignment*),
- ♦ PO1.3. Oszacowanie aktualnej wydajności (ang. *Assessment of Current Performance*),
- ♦ PO1.4. Strategiczny plan IT (ang. *IT Strategic Plan*),
- ♦ PO1.5. Taktyczny plan IT (ang. *IT Tactical Plans*),
- ♦ PO1.6. Zarządzanie portfolio IT (*IT Portfolio Management*).

Wytyczne zarządcze ustalają kwestie komunikacji między procesami, odpowiedzialności za proces oraz kluczowe praktyki zarządcze. Dla „PO1. Wyznaczanie planu strategicznego IT” są to:

- ♦ Informacje o procesach wejściowych (tabela 24) i wyjściowych (tabela 25).

Tabela 24. Tabela danych wejściowych procesu COBIT „PO1. Wyznaczanie planu strategicznego IT”

Z procesu	Dane wejściowe (pol.)	Dane wejściowe (ang.)
PO5	Raporty kosztów i zysków	<i>Cost/benefits reports</i>
PO9	Oszacowanie ryzyka	<i>Risk assessment</i>
PO10	Zaktualizowane portfolio projektów	<i>Updated project portfolio</i>
DS1	Nowe (zaktualizowane) wymagania usługi oraz zaktualizowane portfolio usług	<i>New/updated service requirements; updated service portfolio</i>
*	Strategia biznesowa i priorytety	<i>* Business strategy and priorities</i>
*	Portfolio programów	<i>* Programme portfolio</i>
ME1	Wejściowe dane wydajnościowe do planowania IT	<i>Performance input to IT planning</i>
ME4	Raport o stanie ładu korporacyjnego IT; strategiczne wytyczne dla IT	<i>Report on IT governance status; enterprise strategic direction for IT</i>

Tabela 25. Tabela danych wyjściowych procesu COBIT „PO1. Wyznaczanie planu strategicznego IT”

Dane wyjściowe (pol.)	Dane wyjściowe (ang.)	Do procesów				
Strategiczny plan IT	<i>Strategic IT plan</i>	PO2...PO6	PO8	PO9	AI1	DS1
Taktyczny plan IT	<i>Tactical IT plan</i>	PO2...PO6	PO9	AI1	DS1	
Portfolio projektów IT	<i>IT project portfolio</i>	PO5	PO6	PO10	AI6	
Portfolio usług IT	<i>IT service portfolio</i>	PO5	PO6	PO9	DS1	
Strategia zarządzania zasobami IT	<i>IT sourcing strategy</i>	DS2				
Strategia nabycia zasobów do IT	<i>IT acquisition strategy</i>	AI5				

- ♦ Tabela RACI definiująca matrycę zakresu odpowiedzialności pracowników z poszczególnymi funkcjami w organizacji w stosunku do czynności zdefiniowanych przez konkretny proces. Możliwe relacje to: odpowiedzialny (ang. *Responsible*), zdolny wytłumaczyć (ang. *Accountable*), konsultowany (ang. *Consulted*), informowany (ang. *Informed*). Przykład dla procesu „PO1. Wyznaczanie planu strategicznego IT” przedstawiono na rysunku 71.
- ♦ Rozbudowany system szablonów miar i metryk opisujący „cele działań” (ang. *Activity Goals*). Są one mierzone za pomocą „kluczowych miar wydajności” (ang. *Key Performance Indicator*), które prowadzą do „celów procesowych” (ang. *Process Goals*). Cele te z kolei są mierzone za pomocą „kluczowych miar celów procesowych” (ang. *Process Key Goal Indicators*), które prowadzą do „celów IT” (ang. *IT Goals*). „Cele IT” są mierzone za pomocą „kluczowych

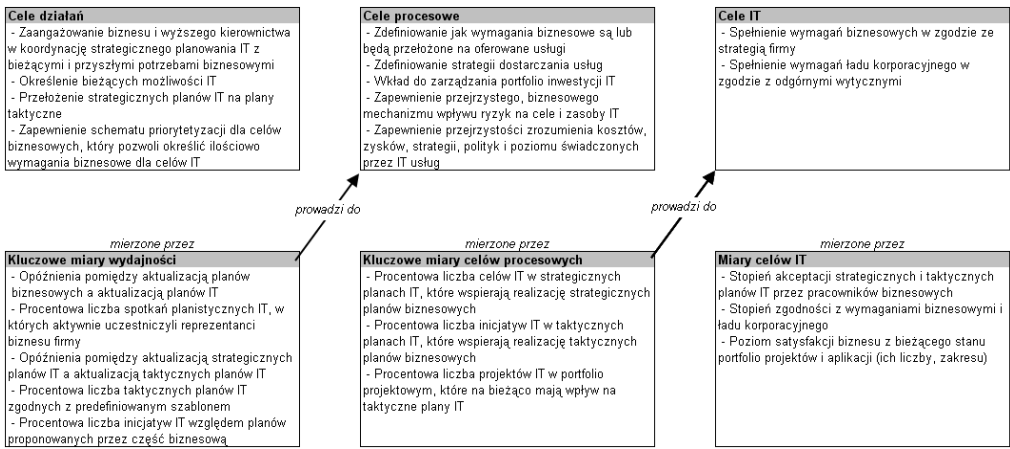
Wykres RACI
Proces PO1 Definiowanie planu strategicznego

	Dyrektor wykonawczy (CEO)	Dyrektor finansowy (CFO)	Decydent biznesowy	Dyrektor Informatyczny (CIO)	Właściciel procesu biznesowego	Kierownik operacyjny	Główny architekt	Kierownik implementacyjny	Kierownik implementacyjny	Biuro projektowe (PMO)	Inni pracownicy (m.in. odpowiedzialni za audyty i bezpieczeństwo)
Połączenie pomiędzy celami biznesowymi a celami IT	C	I	A/R	R	C						
Identyfikacja krytycznych zależności i bieżącej wydajności	C	C	R	A/R	C	C	C	C	C		C
Zbudowanie strategicznego planu IT	A	C	C	R	I	C	C	C	C	I	C
Zbudowanie taktycznego planu IT	C	I		A	C	C	C	C	C	R	I
Analiza portfolio programowego, projektowego i usług	C	I	I	A	R	R	C	R	C	C	I

Wykres RACI identyfikuje, kto jest odpowiedzialny (ang. Responsible), zdolny wytłumaczyć (ang. Accountable, konsultowany (ang. Consulted), informowany (ang. Informed).

Rysunek 71. Przykładowa tabela RACI dla procesu COBIT „PO1. Wyznaczanie planu strategicznego IT”

miar celów IT” (ang. IT Key Goal Indicators). Przykład takich miar i metryk dla procesu „PO1. Wyznaczanie planu strategicznego IT” przedstawiono na rysunku 72.



Rysunek 72. Miary metryki procesu COBIT PO1

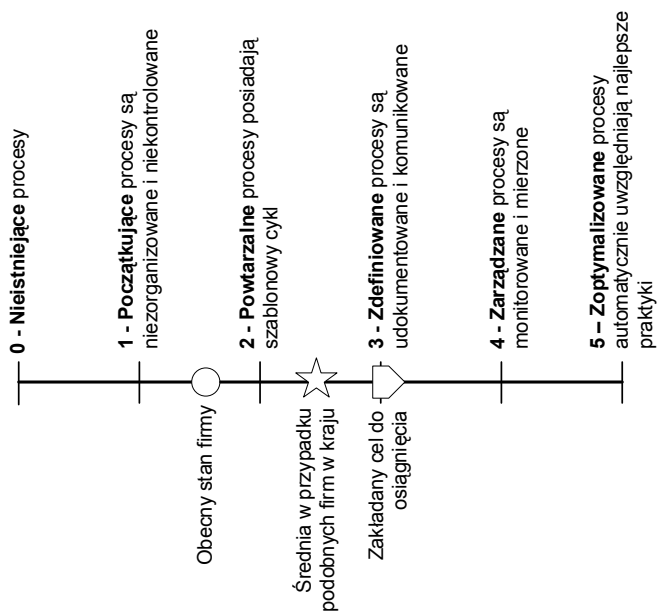
Modele dojrzałości opisują „wygląd” procesu w kontekście pięciu możliwych poziomów dojrzałości. Poziomy te są bardzo podobne do pięciu poziomów dojrzałości znanych z CMMI. Przykładowy audyt zewnętrzny może wykazać stan faktyczny analogiczny do tego, co przedstawiono na rysunku 73.

W kontekście konkretnego procesu, takiego jak proces „PO1. Wyznaczanie planu strategicznego IT”, mogą one być zdefiniowane następująco.

0 — Nieistniejący proces, gdy planowanie strategiczne IT nie jest wykonywane i kierownictwo nie ma świadomości ani nie wyraża potrzeby wspierania takich działań.

Rysunek 73.

Poziomy dojrzałości
COBIT



1 — Początkujący proces, gdy kierownictwo wie o konieczności strategicznego planowania IT, ale jest ono wykonywany reaktywnie, w odpowiedzi na specyficzne wymagania biznesowe. Kwestie te są okazjonalnie dyskutowane na spotkaniach IT, a ryzyka są nieformalnie identyfikowane w poszczególnych projektach.

2 — Powtarzalny, ale intuicyjny proces, gdy strategiczne planowanie IT jest uzgadnianie z kierownictwem biznesowym na żądanie, a aktualizacja takich planów pojawia się w odpowiedzi na wymagania kierownictwa. Strategiczne decyzje są podejmowane w kontekście poszczególnych projektów bez całościowej, spójnej strategii organizacyjnej. Kluczowe, strategiczne decyzje dotyczące ryzyka i potencjalnych korzyści są podejmowane intuicyjnie.

3 — Zdefiniowany proces, gdy istnieją polityki definiujące, kiedy i jak wykonywać strategiczne plany IT. Plany te są wykonywane zgodnie z ustalonym, ustrukturyzowanym i udokumentowanym procesem znanym całemu personelowi. Plany strategiczne są realistyczne i zawierają spójną definicję ryzyk, które organizacja chce świadomie podjąć. Przygotowane plany strategiczne IT są przedyskutowywane w trakcie zarządczych spotkań biznesowych.

4 — Zarządzany i mierzalny proces, gdy strategiczne planowanie IT jest standardową praktyką. Proces ten jest zdefiniowaną funkcją zarządczą i ustalone są zakresy odpowiedzialności. Kierownictwo może monitorować proces planowania strategicznego IT, podejmować na jego bazie świadome decyzje i mierzyć jego efektywność. Długo- i krótkoterminowe plany strategiczne są uszczegółowiane na niższych szczeblach, które mają możliwość zainicjowania w nich zmian. Istnieje dobrze zdefiniowany proces, który determinuje użycie wewnętrznych i zewnętrznych zasobów wymaganych w trakcie wytwarzania systemów i innych operacji.

5 — Zoptymalizowany proces, gdy strategiczne planowanie IT jest udokumentowanym, żywym procesem, który jest stale weryfikowany pod kątem celów biznesowych i w rezultacie przedstawia rozpoznawalną wartość biznesową. W ramach tego procesu ryzyka i wartości dodane są stale analizowane oraz aktualizowane. W odpowiedzi na zmieniające się technologie i sytuację biznesową aktualizuje się również plany długoterminowe. Plany te zawierają informacje, jak nowe technologie mogą tworzyć nowe możliwości biznesowe i zwiększyć konkurencyjność organizacji. Uwzględniane są dobrze znane i zaufane normy oraz standardy.

... i tak dla każdego z trzydziestu czterech procesów na stu trzydziestu siedmiu stronach ☺.

Taki schematyczny opis wszystkich procesów nuży po przeczytaniu dziesiątej strony, jeżeli czytamy oficjalne materiały od deski do deski, ale umożliwia bardzo szybkie wyszukanie konkretnej paczki informacji, niezależnie od kryteriów poszukiwań. Jeśli więc interesują nas kwestie związane z poufnością, najpierw będziemy przyglądali się „kluczowym kryteriom” procesu. Jeśli interesuje nas, za co naprawdę odpowiedzialny jest dyrektor informacyjny (CIO) w ramach COBIT, skorzystamy najpierw z tabeli RACI. Gdy pełnimy rolę audytora i mamy zdefiniować poziom dojrzałości poszczególnych procesów, zgłębimy głównie ostatnią część opisu każdego z procesów.

Ponadto COBIT definiuje listę szablonowych celów biznesowych pogrupowanych w cztery główne perspektywy.

Perspektywa finansowa (1 – 5):

- 1)** rozszerzenie udziału w rynku,
- 2)** zwiększenie zysku,
- 3)** zwrot z inwestycji,
- 4)** optymalizacja wykorzystania zasobów,
- 5)** zarządzanie ryzykiem biznesowym.

Perspektywa klienta (6 – 10):

- 6)** udoskonalenie obsługi klienta,
- 7)** oferowanie konkurencyjnych produktów i usług,
- 8)** dostępność usług,
- 9)** szybkie odpowiadanie na potrzeby zmieniającego się rynku,
- 10)** optymalizacja kosztu dostawy usług.

Perspektywa wewnętrzna (11 – 17):

- 11)** automatyzacja i integracja łańcucha przekazywania wartości,
- 12)** udoskonalenie i utrzymanie biznesowej funkcjonalności procesów,
- 13)** zmniejszenie kosztów procesów,

- 14)** zgodność z wymaganiami prawnymi i regulacjami zewnętrznymi,
- 15)** przejrzystość,
- 16)** zgodność z politykami wewnętrznymi,
- 17)** udoskonalenie i utrzymanie produktywności operacyjnej oraz pracowników.

Perspektywa uczenia się i wzrostu (18 – 20):

- 18)** innowacyjność produktów i biznesu,
- 19)** otrzymywanie solidnych i praktycznych informacji podczas podejmowania decyzji strategicznych,
- 20)** zdobywanie i utrzymanie wykwalifikowanych oraz zmotywowanych pracowników.

Podaje też szablony dwudziestu ośmiu celów IT.

- 1.** Spełnianie wymagań biznesowych w zgodzie ze strategią biznesową.
- 2.** Spełnianie wymagań ładu korporacyjnego w zgodzie z ogólnymi wytycznymi.
- 3.** Zapewnienie satysfakcji użytkowników z oferowanych usług i poziomu ich świadczenia.
- 4.** Optymalizacja wykorzystania informacji.
- 5.** Stworzenie adaptacyjnego IT.
- 6.** Zdefiniowanie, jak funkcjonuje biznes, i kontrola oraz zapewnienie, że wymagania są efektywnie i wydajnie przełożone na zautomatyzowane rozwiązania.
- 7.** Wdrożenie oraz utrzymanie zintegrowanych i ustandaryzowanych systemów aplikacji.
- 8.** Wdrożenie oraz utrzymanie zintegrowanej i ustandaryzowanej infrastruktury IT.
- 9.** Wdrożenie oraz utrzymanie umiejętności IT, które odpowiadają strategii IT.
- 10.** Zapewnienie obopólnej satysfakcji w relacjach z firmami zewnętrznymi.
- 11.** Płynna integracja aplikacji i rozwiązań technologicznych w procesach biznesowych.
- 12.** Zapewnienie przejrzystości zrozumienia kosztów, zysków, strategii, polityk i poziomu świadczonych przez IT usług.
- 13.** Zapewnienie właściwego użycia i wydajności aplikacji oraz rozwiązań technologicznych.
- 14.** Księgowanie i ochrona wszystkich aktywów IT.
- 15.** Optymalizacja infrastruktury, zasobów i możliwości IT.
- 16.** Redukcja defektów i ponownej pracy w trakcie dostarczania rozwiązań i usług.
- 17.** Ochrona osiągniętych celów IT.

18. Zapewnienie przejrzystego, biznesowego mechanizmu wpływu ryzyk na cele i zasoby IT.
19. Zapewnienie, że krytyczne i poufne informacje nie są dostępne dla nieuprawnionych osób.
20. Zapewnienie automatycznych transakcji biznesowych i zaufania do wymiany informacji.
21. Zapewnienie, że usługi i infrastruktura IT mogą właściwie przetrwać i odtworzyć się po uszkodzeniu na skutek błędu, ataku lub katastrofy.
22. Zapewnienie, że zakłócenia i zmiany usług IT będą miały minimalny wpływ na biznes.
23. Uczynienie usług IT dostępnymi, gdy są wymagane.
24. Udoskonalenie efektywności kosztowej IT i jej wpływu na zyskowność biznesową.
25. Dostarczenie projektów na czas i w budżecie, zgodnie ze standardami jakościowymi.
26. Utrzymanie integralności informacji i rozwój infrastruktury.
27. Zapewnienie zgodności IT z prawem i zewnętrznymi regulacjami.
28. Zapewnienie, że IT demonstruje efektywną kosztowo jakość usług, stałe doskonalenie i gotowość do przyszłych zmian.

COBIT łączy te grupy celów z poszczególnymi procesami. Gdy organizacja koncentruje się na realizacji wybranych celów biznesowych i (lub) celów IT, umożliwia to łatwy wybór grupy niezbędnych procesów.

Bardziej szczegółowe informacje o poszczególnych procesach zawarto w załączniku D — Lista wszystkich procesów COBIT.

Ciekawym uzupełnieniem metodyki COBIT jest metodyka Val IT promowana przez IT Governance Institute (ITGI). Obie zajmują się kwestią ładu korporacyjnego, ale z nieco innej perspektywy. Podczas gdy COBIT koncentruje się głównie na kwestiach wykonawczych, Val IT zajmuje się kwestiami związanymi z decyzjami inwestycyjnymi oraz realizacją zysków. O ile COBIT odpowiada na pytanie: „Czy robimy rzeczy we właściwy sposób i otrzymujemy właściwie rezultaty?”, ValIT odpowiada na pytanie: „Czy robimy właściwe rzeczy i osiągamy zyski?”.

Certyfikacja

W Polsce popularyzacją tej metodyki zajmuje się ISACA, która ma obecnie 65 tys. członków w 140 krajach. Warszawski oddział tej organizacji regularnie współorganizuje konferencje poświęcone m.in. metodyce COBIT. Piąta taka konferencja, „Jakość2008”, miała miejsce w dniach 28 – 29 maja 2008 roku.

Istnieją trzy certyfikaty COBIT:

- ♦ *Certified Information Systems Auditor (CISA)* — skierowany do osób, które chciałyby pełnić rolę audytora zewnętrznego. Można go uzyskać od 1978 roku i na świecie ma go ponad 50 tys. osób, a w Polsce 160. Egzamin trwa cztery godziny i składa się z dwustu pytań wielokrotnego wyboru, za które można uzyskać maksymalnie 800 punktów. Minimum konieczne do zdania egzaminu to 450 punktów. Wymagane jest pięcioletnie, zweryfikowane doświadczenie w audytowaniu, kontroli lub kwestiach bezpieczeństwa systemów informatycznych zdobyte w czasie ostatnich dziesięciu lat — studia mogą zostać zaliczone jako dwa lata takiego doświadczenia. Szacunkowy koszt egzaminu to od 620 do 960 zł (310 – 480 USD).
- ♦ *Certified Information Security Manager (CISM)* — przeznaczony dla osób na stanowiskach kierowniczych związanych z bezpieczeństwem informacji. Istnieje od 2003 roku i na świecie posiada go 7 tys. osób, a w Polsce 32 osoby. Pozostałe cechy są identyczne z CISA.

Oprócz certyfikatów związanych z COBIT, organizacja ISACA wraz z IT Governance Institute (ITGI), oferuje również ciekawy certyfikat w obszarze ładu korporacyjnego (ang. *IT Governance*) — *Certified in the Governance of Enterprise IT (CGEIT)*. Certyfikat ten obejmuje swym zakresem tematykę związaną z COBIT 4.1, ValIT oraz szereg innych materiałów, których pełną listę można odnaleźć na stronach ISACA. Certyfikat ten jest przeznaczony dla osób zarządzających, doradzających i zapewniających zgodność z ładem korporacyjnym. Egzaminy organizowane są dwa razy do roku w grudniu i czerwcu. Ich koszt to wydatek rzędu 1100 – 1700 zł (345 – 525 USD).

Podsumowanie

COBIT jest metodyką dedykowaną zarządowi, kierownictwu biznesowemu, osobom zarządzającym działami IT oraz profesjonalistom do spraw bezpieczeństwa, audytu i kontroli. Koncentruje się na tematyce ładu korporacyjnego oraz zarządzania ryzykiem. Unikalną cechą tej metodyki są szablonowe cele biznesowe i cele IT oraz ich mapowanie na procesy. Dzięki temu jest to bardzo skalowalne rozwiązanie, które można łatwo dopasować do aktualnych potrzeb organizacji.

COBIT pozostaje również jedną z ciekawszych metodyk audytorskich. Jest często stosowana w instytucjach publicznych i prywatnych, a szczególnie w tych z sektora bankowego.

W przeciwieństwie do OCG i PMI, duża część dokumentacji ISACA jest darmowa, a dostęp do pozostałej części jest stosunkowo tani. Być może właśnie dlatego liczba członków tej organizacji stale rośnie i jest obecnie na poziomie 80 tys. członków w 70 krajach (50 tys. posiada certyfikat CISA).

Część V

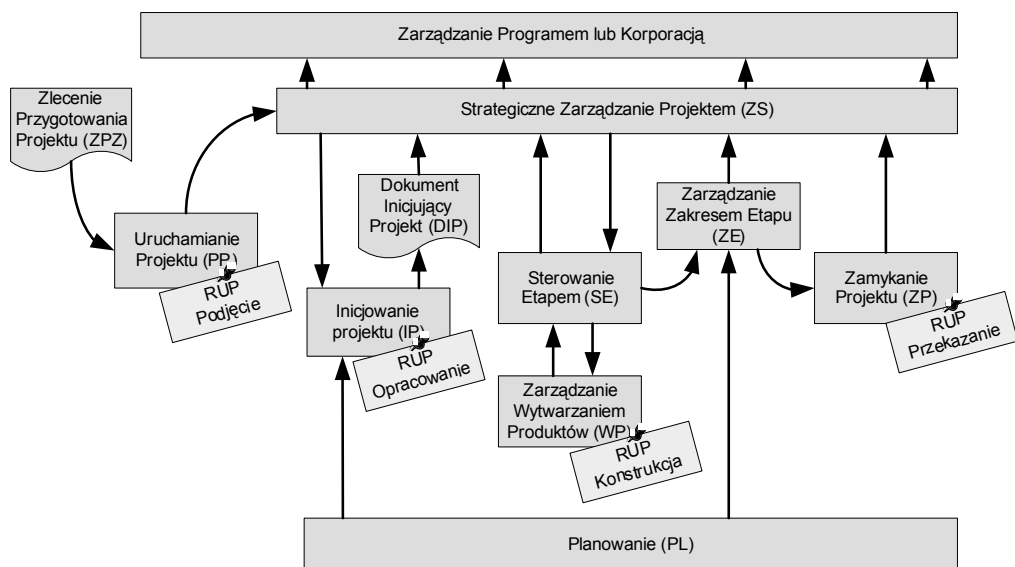
Rozwiązania kombinowane

Poszczególne grupy metod nie wykluczają się nawzajem, ponieważ każda z nich koncentruje się na innym aspekcie działań.

- ◆ Metodyki zarządcze skupiają się na formalizowaniu mechanizmów decyzyjnych związanych z projektami oraz zapewniają właściwe mechanizmy kontrolne w obrębie projektu.
- ◆ Metodyki wytwórcze koncentrują się na wytwarzaniu produktów.
- ◆ Metodyki organizacyjne zajmują się głównie zwiększaniem dojrzałości organizacyjnej firmy.

Można sobie wyobrazić sytuację, gdzie w ramach jednej organizacji współistnieje kilka metodyk. Jest to szczególnie interesujące w organizacjach, które ogólnie wymagają formalnych decyzji i kontroli przebiegu projektów, ale istnieje w nich silna, oddolna tendencja do osiągnięcia większej swobody w obszarze wytwarzania produktów. W takim przypadku połączenie metodyki, takiej jak Prince2 lub PMBoK, z metodyką RUP, MSF, a nawet SCRUM czy XP, może dać bardzo ciekawe efekty.

Jeżeli za punkt odniesienia przyjmiemy metodykę Prince2, faza podjęcia RUP może rozpocząć się już w trakcie procesu uruchamiania projektu (PP), a osiągnięcie *Lifecycle Objective Milestone* to nic innego jak pozytywna decyzja z procesu strategicznego zarządzania projektem (ZS). Analogicznie, dalsza część prac analitycznych, która ma dać końcowy i pełny plan wykonania projektu jako dokument DIP, może zostać wykonana m.in. za pomocą fazy opracowania z RUP. Zlecenie wytworzenia produktów do wykonania (WP), to przecież przekazywanie prac do produkcji, czyli w przypadku RUP do fazy konstrukcji. Ostateczne wdrożenie systemu może być oddzielnym etapem Prince2 lub częścią procesu zamykania projektu (ZP), która jest odpowiedzialna za przekazanie finalnego produktu z zespołu projektowego do zespołu utrzymaniowego. Od strony RUP wsparcie zapewnione jest w ramach fazy przekazania. Całościowy obraz takiego łączenia procesów przedstawiono na rysunku 74.



Rysunek 74. Prince2 a RUP

Warto tutaj nadmienić, że definicje ról w Prince2 i RUP dość dobrze się dopełniają. Prince2 koncentruje się głównie na rolach zarządczych i najbardziej „specjalistyczną rolą” jest kierownik zespołu zadaniowego. RUP natomiast skupia się na rolach „technicznych” i enigmatycznie opisuje role zarządcze.

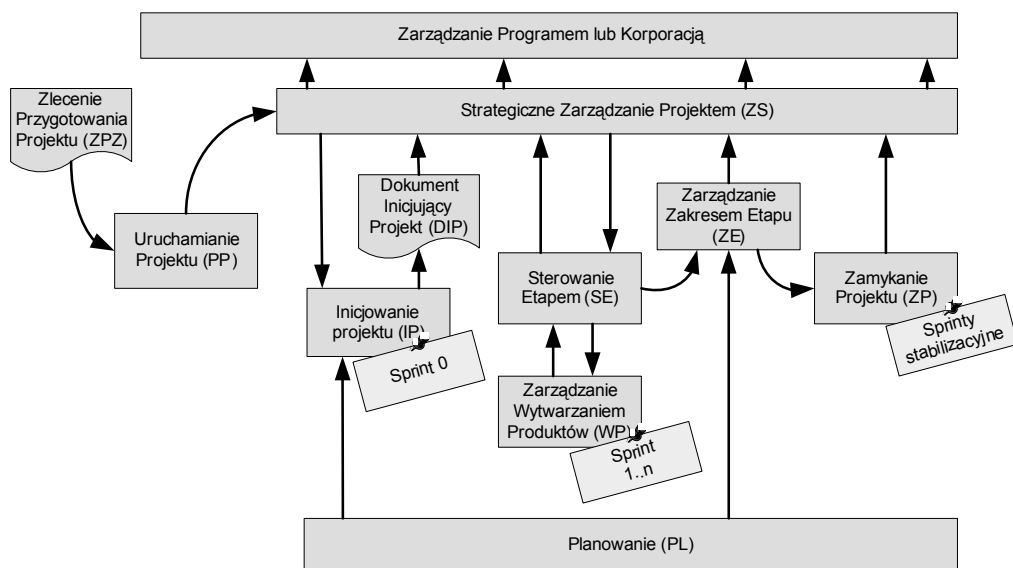
W podobny sposób Prince2 może zostać połączony z metodyką Scrum. W tym przypadku cała odpowiedzialność za realizację kontraktu spoczywa „na barkach” metodyki Prince2. Z tej właśnie metodyki warto wykorzystać mechanizmy tworzenia uzasadnienia biznesowego. Po wstępnej akceptacji planów biznesowych, w trakcie szczegółowego przygotowywania planów technicznych, można już uruchomić sprint zerowy. Pojęcie to w przypadku Scrum oznacza etap „rozwrotu” zespołu wytwórczego i wyjątkowo nie musi dostarczyć żadnego produktu o konkretnej wartości biznesowej. Mechanizm ten idealnie pasuje do techniki dowodów wykonalności z Prince2. Na tej podstawie będzie możliwe utworzenie o wiele bardziej rzeczywistych harmonogramów i planowanych kosztów wykonania poszczególnych etapów projektu. Etapy Prince2 mogą, ale nie muszą, pokrywać się z etapami metodyki Scrum. Jeden etap Prince może składać się z:

- ♦ kilku sprintów,
- ♦ dodatkowego czasu na testy, audyty lub inne prace wykonywane pomiędzy sprintami,
- ♦ dodatkowego zapasu w harmonogramie na nieprzewidziane okoliczności lub zmiany.

Wdrożenie projektu może zostać zrealizowane przez prince’owy proces zamknięcia wykonywany równolegle ze sprintami stabilizacyjnymi. Scrum zakłada przekazanie w ramach sprintów gotowych produktów biznesowych, ale zanim zostanie podpisany ostateczny protokół odbioru często mamy do czynienia z wieloma drobnymi propozycjami

zmian i błędami. Dobrym rozwiązaniem jest tutaj zastosowanie kilku iteracji testów, z których każdy zakończony jest konkretnym punktem kontrolnym i osiągnięciem określonej bramki jakościowej. Przykłady takich bramek to ostateczna akceptacja zakresu funkcjonalnego, akceptacja wydajności rozwiązania, przekazanie projektu. Po stronie wykonawcy oznacza to zarezerwowanie pewnych zasobów na poszczególne iteracje testów. Iteracje te mogą być interpretowane właśnie jako sprinty stabilizacyjne.

Całościowy obraz takiego łączenia procesów przedstawiono na rysunku 75.

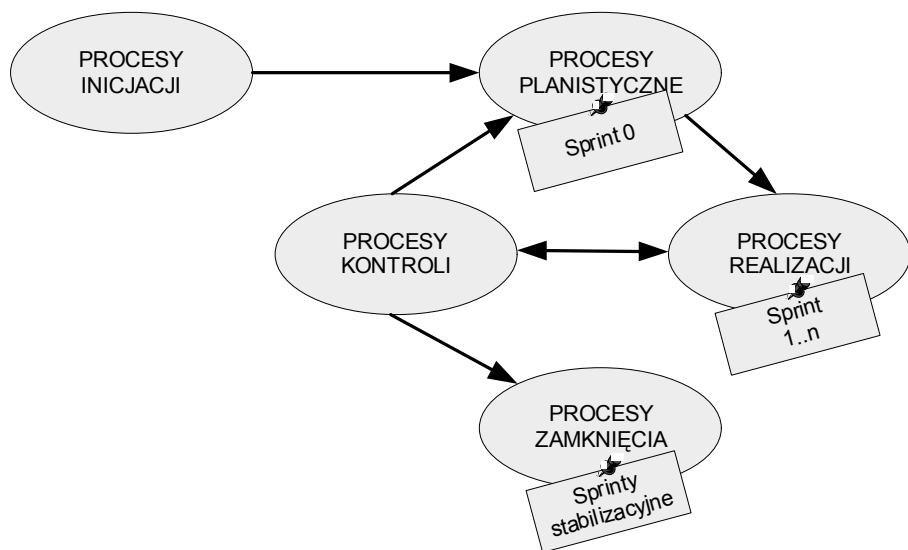


Rysunek 75. *Prince2 a Scrum*

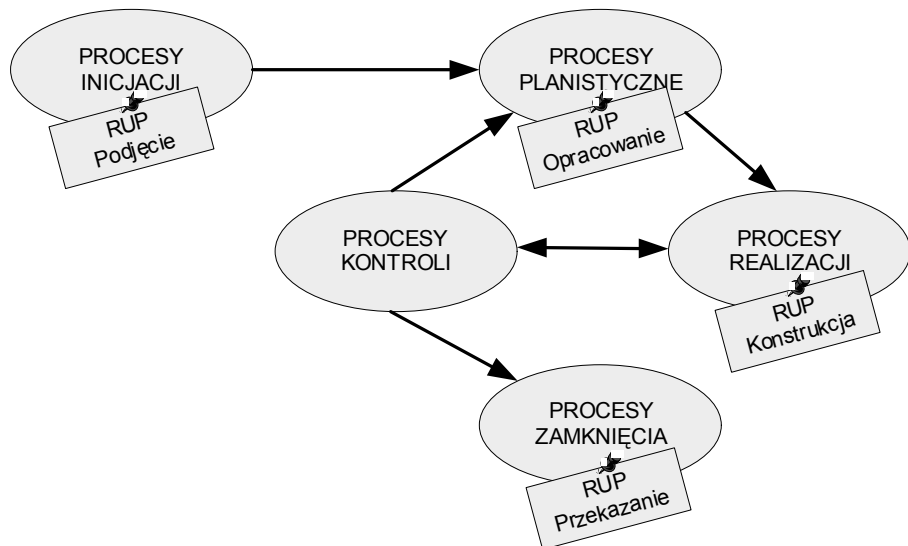
Na podobnej zasadzie można „zintegrować” metodyki PMBoK i Scrum. PMBoK posiada równie ciekawe techniki tworzenia uzasadnień biznesowych w ramach procesów inicjacyjnych. Procesy planistyczne mogą zawierać sprint zerowy, w którym zostaną opracowane dowody wykonalności. Procesy realizacyjne mogłyby w takiej sytuacji uruchamiać jeden lub kilka kolejnych sprintów. Wdrożenie można by rozpatrywać jako sprinty stabilizacyjne lub działania nieobjęte podejściem Scrum — rysunek 76.

Analogicznie PMBoK może zostać skojarzony z RUP. W tym przypadku faza podjęcia RUP może być wykonywana w trakcie procesów inicjacyjnych PMBoK, które mają za zadanie utworzenie dokumentu otwarcia. Przejście do procesów inicjacyjnych z PMBoK mogłoby uruchomić fazę opracowania RUP, a procesy realizacyjne z PMBoK oznaczałyby uruchomienie fazy konstrukcji RUP. Analogicznie, faza przekazania RUP byłaby odpowiedzialna za realizację zadań procesów zamknięcia PMBoK. Istnieje tutaj wariant, w którym wdrożenie nadzorowane jest jako proces realizacyjny lub jest wręcz oddzielnym etapem PMBoK — rysunek 77.

W czterech wymienionych powyżej przypadkach można zamienić RUP na MSF i Scrum na XP. Po drobnych poprawkach uda się zapewne stworzyć równie ciekawe rozwiązania. Możliwych kombinacji metodyk oraz sposobów ich współistnienia jest z resztą więcej.



Rysunek 76. PMBoK a Scrum



Rysunek 77. PMBoK a RUP

Wdrażanie rozwiązań kombinowanych powoduje zwiększone ryzyko problemów komunikacyjnych i zrzucania odpowiedzialności. Dlatego też w ramach analizy *post mortem* warto udokumentować wszystkie praktyki, które sprawdziły się w działaniu, i te, które zawiodły. Zbudowanie takiej bazy wiedzy umożliwi z czasem bardziej przejrzysty i uregulowany sposób działania. Odpowiednio duży zbiór takich „nauczek” może doprowadzić do sytuacji, w której organizacja zacznie promować własną metodykę ☺.

Oprócz pary metodyk zarządcza-wytwórcza, firma może również stosować mechanizmy zarządzania katalogiem usług oferowanym przez ITIL. Podczas wprowadzania ładu korporacyjnego byłaby to metodyka COBIT. Należy tutaj zwrócić uwagę na pokrywające się elementy; i tak w przypadku pary PMBoK-COBIT są to obszar wiedzy zaopatrzenia (12) i zarządzania zasobami ludzkimi (9). Niemniej jednak, metodyki zarządcze i organizacyjne uzupełniają się nawzajem. Przykładowo COBIT i ITIL nadają się dość dobrze do zarządzania programami, czyli tematyką wykraczającą poza zakres Prince2 i PMBoK. Pojawia się więc pytanie, które rozwiązania kombinowane przynoszą zazwyczaj wartość dodaną.

Próba odpowiedzi na to pytanie została zawarta w tabeli 26. Takie pary metodyk jak Prince2-PMBoK oraz MSF-RUP wyraźnie ze sobą współzawodniczą. Oczywiście, każda z tych metodyk może wykorzystać pewne konkretne techniki z metodyki konkurencyjnej, ale ich koegzystencja w firmie raczej nie przyniesie wartości dodanej. Są też pewne grupy metodyk pasujące do siebie w sposób naturalny. Podane powyżej przykłady par Prince2 lub PMBoK oraz MSF, RUP, Scrum są tego najlepszym przykładem. Parę taką można wzbogacić dodatkowo o jedną z metodyk organizacyjnych. Można wyobrazić sobie sytuację, w której organizacja posiada uporządkowane usługi (ITIL), realizuje projekty zgodnie z Prince2, a programiści i analitycy aktywnie wykorzystują metodykę RUP.

Tabela 26. Wzajemna korelacja pomiędzy metodykami

		Metodyki zarządcze		Metodyki wytwórcze				Metodyki organizacyjne			
		Prince2	PMBoK	RUP	MSF	XP	Scrum	CMMI	Six Sigma	ITIL	COBIT
Metodyki zarządcze	Prince2		-4	4	4	3	4	3	4	3	3
	PMBoK	-4		4	4	3	4	3	4	3	3
Metodyki wytwórcze	RUP	4	4		-5	0	-1	3	4	3	3
	MSF	4	4	-5		0	-1	3	4	3	3
	XP	3	3	0	0		4	4	4	4	4
	Scrum	4	4	-1	-1	4		4	4	4	4
Metodyki organizacyjne	CMMI	3	3	3	3	4	4		4	1	-1
	Six Sigma	4	4	4	4	4	4	4		4	4
	ITIL	3	3	3	3	4	4	1	4		1
	COBIT	3	3	3	3	4	4	-1	4	1	

LEGENDA skali ocen

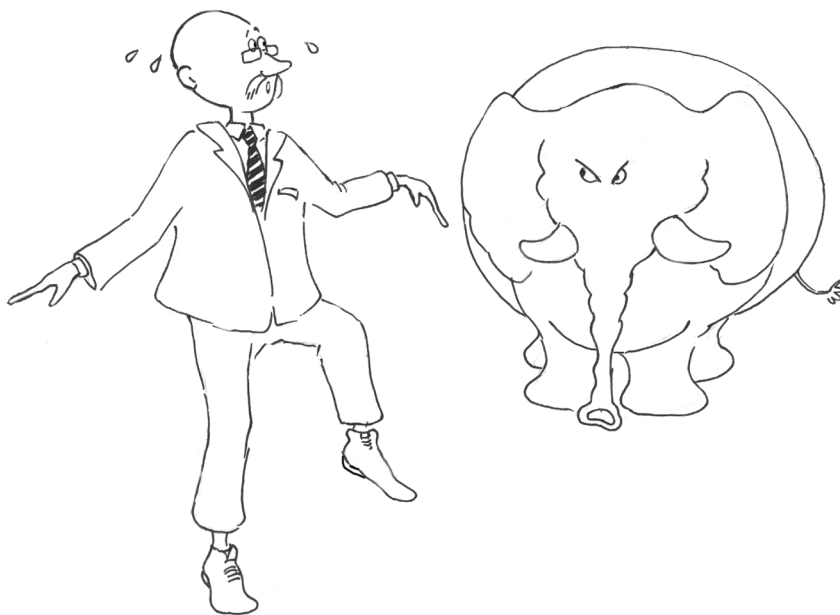
- 5 Konkurencyjne metodyki, bardzo duże koszty łączenia przy minimalnych korzyściach
- 3 W większości pokrywająca się tematyka, widać wyraźną przewagę kosztów łączenia nad korzyściami
- 1 Istnieje wartość dodana z kombinowania metodyk, ale koszty łączenia zazwyczaj przerastają korzyści
- 0 Stosunek obojętny lub porównywalne koszty i korzyści łączenia
- 1 Dobrze pasujące metodyki, korzyści łączenia zazwyczaj przewyższają koszty
- 3 Bardzo dobrze pasujące metodyki, widać wyraźną przewagę korzyści łączenia nad kosztami
- 5 Idealne dopasowanie metodyk, bardzo duże korzyści łączenia przy minimalnych kosztach

Rozwiązania kombinowane są bardziej zaawansowane i należy je stosować wtedy, gdy ma się potwierdzoną pewność, że jedna z wymienionych metodyk działa już poprawnie, a „czegoś jeszcze brakuje”. Warto wprowadzać takie rozwiązania przyrostowo, sprawa po sprawie lub projekt po projekcie. Zgodnie z lekarską zasadą *primum non nocere*, należy zwrócić szczególną uwagę na poszanowanie działających, dobrych praktyk.

Podsumowanie

Podstawową zasadą jest brak zasad. Nie ma metodyki, która rozwiązałaby wszystkie problemy, tak samo jak nie ma *perpetum mobile*. Wszystkie opisane w książce metodyki to szablony rozwiązań i zbiory dobrych praktyk, które można i należy dostosować do własnych potrzeb. Jeżeli zdrowy rozsądek mówi, że trzeba postąpić wbrew metodyce lub złamać procedurę, należy przeliczyć ryzyko i podjąć właściwą decyzję, nawet jeśli wiedzie przez nieprzetarte szlaki! Takie podejście jest zgodne z ideą ciągłej poprawy jakości wdrażanych procesów.

Wujek Dobra Rada — odcinek 14. „Bura procedura”



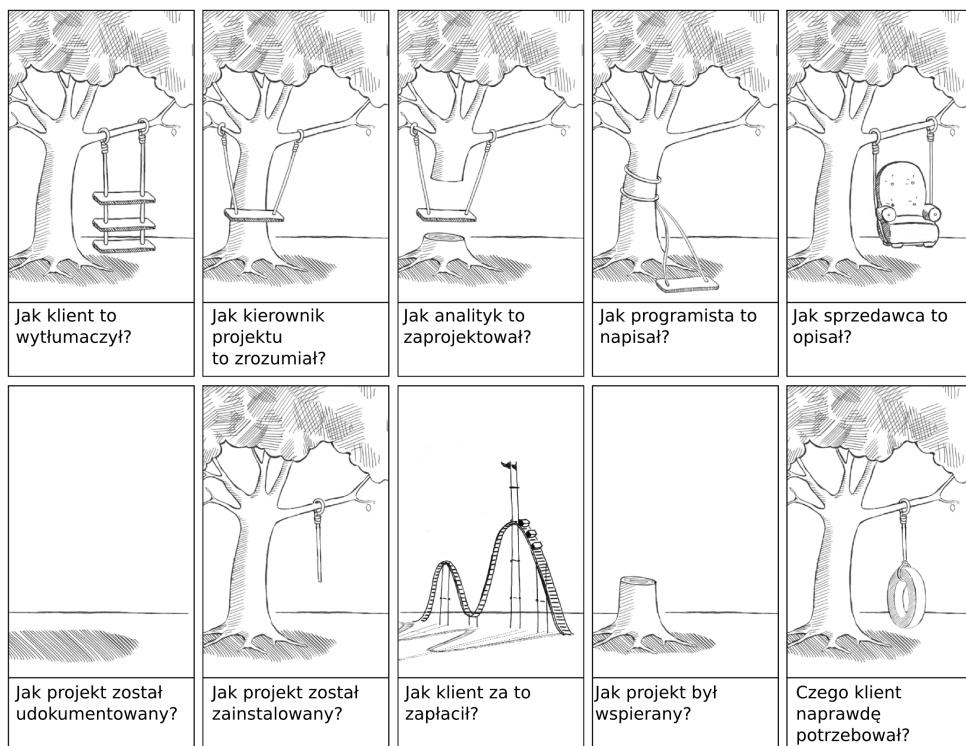
KAŻDĄ PROCEDURĘ MOŻNA OBEJŚĆ.
PYTANIE, CZY MA TO SENS I CZY SIĘ OPŁACA?

Ze względu na powyższe, organizacje o wysokiej kulturze stale renegocjują zasady wzajemnej, wewnętrznej współpracy. Należy jednak pamiętać o tym, że istnieje pojęcie „złego kompromisu”.

Kompromis jest wtedy, gdy jedno chce w góry, a drugie nad morze i w rezultacie obydwójce lądują w Wielkopolsce, czyli nikt nie zrealizował swoich celów. W takiej sytuacji należy jechać na południe Francji, gdzie są i góry, i morze¹.

Jedynym z największych wyzwań w projektach informatycznych jest osiągnięcie zadowolenia klientów przy zachowaniu finalnej zyskowności. Zazwyczaj oznacza to, że w bardzo ograniczonym czasie, za małe pieniądze trzeba przewidzieć, czego użytkownik końcowy **będzie** sobie życzył w momencie odbioru projektu. Większość metodyk nieadaptacyjnych, takich jak Prince2, wymaga zdefiniowania wymagań klientów na początku projektu. Zapewnia to jasność sytuacji od strony formalnej, ale nie zawsze służy zadowoleniu klienta. Dlatego też tak wielu zwolenników zdobywają metodyki adaptacyjne, które przedkładają wzajemną satysfakcję obu stron nad zgodność formalną. Jeżeli obie strony negocjują w duchu wygrana-wygrana (ang. *win-win*)² i są zainteresowane pozytywną, długofalową współpracą, takie podejście ma duże szanse powodzenia.

A teraz nadszedł czas, aby wziąć ten zasobnik z narzędziami i samemu coś zmajstrować ☺.



¹ B. Wojcieszke; „Polityka”, Pomocnik Psychologiczny, „Miłość ci wszystko wypaczy”, 9 lutego 2008.

² R. Caldini, *Wywieranie wpływu na ludzi*, Gdańskie Wydawnictwo Psychologiczne, Gdańsk, 1998.

Dodatki

Dodatek A

Lista wszystkich procesów PMBoK

Każdy z czterdziestu czterech procesów PMBoK znajduje się w jednej z pięciu grup procesów i jednym z dziewięciu obszarów wiedzy.

	4. Integracja	5. Zakres	6. Czas	7. Koszt	8. Jakość	9. Zasoby ludzkie	10. Komunikacja	11. Ryzyko	12. Zaopatrzenie
Inicjacja	4.1						10.1		
Planowanie	4.2	5.1 – 5.3	6.1 – 6.5	7.1 – 7.2	8.1	9.1	10.2	11.1 – 11.5	12.1
Wykonanie	4.3				8.2	9.2 – 9.4	10.3-10.4		12.2
Kontrola	4.4 – 4.5	5.4 – 5.5	6.6	7.3	8.3		10.5	11.6	12.3
Zamykanie	4.6								12.4

Procesy inicjacji (ang. *initating processes*) to wszelkie operacje związane z uruchomieniem projektu.

♦ **Nazwa polska: 4.1. Opracowanie dokumentu otwarcia**

Nazwa angielska: 4.1. *Develop Project Charter*

Wejście:

- Zapis dotyczący zakresu planowanych prac (ang. *Project Statement of Work*),
- Biznesowe przypadki użycia,
- Umowy (o ile takowe istnieją),
- Inna dokumentacja, która może mieć wpływ na proces, taka jak dokumentacja poprzedniej wersji produktu lub rejestr nauczerek.

Przygotowanie dokumentu, który powinien zawierać kluczowe, ogólne informacje na temat projektu w kontekście wszystkich dziewięciu obszarów wiedzy PMBoK. Dokument ten powinien zostać zaakceptowany przez właściwe gremium decyzyjne składające się z kluczowych interesariuszy (ang. *Stakeholders*).

Obszar wiedzy:

Integracja

Techniki:

Osąd eksperta (ang. *Expert Judgment*).

Wyjście:

Dokument otwarcia (ang. *Project charter*).

♦ **Nazwa polska: 10.1 Identyfikacja** interesariuszy

Nazwa angielska: 10.1 Identify Stakeholders

Wejście:

Dokument otwarcia (ang. *Project charter*),

Umowy (o ile takowe istnieją),

Inna dokumentacja, która może mieć wpływ na projekt (np. dokumentacja poprzedniej wersji produktu, analiza przedwstępna itp.).

Identyfikacja wszystkich osób i organizacji, które mają wpływ na projekt, oraz dokumentacja informacji na temat ich interesów oraz zaangażowania. Krytyczną kwestią dla sukcesu projektu jest wczesna identyfikacja tych kwestii, bo dzięki temu można opracować najlepszą strategię realizacyjną, która powinna być okresowo aktualizowana.

Obszar wiedzy: Komunikacja

Techniki:

Analiza interesariuszy (ang. *Stakeholders Analysis*),

Osąd eksperta.

Wyjście:

Rejestr interesariuszy (ang. *Stakeholder Register*),

Strategia zarządzania interesariuszami (ang. *Stakeholder Management Strategy*).

Procesy planistyczne (ang. *planning processes*) to uszczegółowienie zaakceptowanych ram projektowych i szereg taktycznych odpowiedzi na temat sposobu realizacji zadań, których wynikiem jest kompletny, szczegółowy plan prac.

♦ **Nazwa polska: 4.2. Opracowanie planu zarządzania projektem**

Nazwa angielska: 4.2. Develop Project Management Plan

Wejście:

Dokument otwarcia (ang. *Project charter*),

Efekty procesów planistycznych,

Inna dokumentacja, która może mieć wpływ na proces, np. dokumentacja poprzedniej wersji produktu lub rejestr nauczek.

Jest to proces nadrzędny w stosunku do wszystkich innych procesów planistycznych, poczynając od procesu planowania zakresu, a kończąc na procesie planowania zakupów i akwizycji. W ramach tego procesu kierownik projektu decyduje, które podprocesy planistyczne powinny być uruchomione, a które nie, oraz dba o ich wzajemną koordynację. Proces ten jest również odpowiedzialny za zebranie efektów prac wszystkich podprocesów planistycznych i zintegrowanie ich w ramach jednego, zbiorczego dokumentu.

Obszar wiedzy:

Integracja.

Techniki:

Osąd eksperta.

Wyjście:

Plan zarządzania projektem (ang. *Project Management Plan*).

♦ Nazwa polska: 5.1. Zbieranie wymagań

Nazwa angielska: 5.1. *Collect Requirements*

Wejście:

Dokument otwarcia (ang. *Project charter*),

Rejestr interesariuszy (ang. *Stakeholders Register*).

Proces udokumentowania wymagań interesariuszy w kontekście realizacji celów projektowych. Proces ten ma zapewnić wystarczająco dużo szczegółów, aby możliwe było zmierzenie poziomu realizacji tych wymagań w trakcie realizacji projektu.

Obszar wiedzy:

Zakres.

Techniki:

Wywiady — spotkania, które polegają na poznaniu potrzeb jednego lub wielu interesariuszy,

Grupy specjalistyczne — jest to nieco bardziej konwersacyjny sposób ustalania zakresu w stosunku do wywiadów,

Warsztaty — mają na celu uzgodnienie kwestii, w których poszczególni interesariusze się różnią, oraz ściślejszą współpracę pomiędzy zespołami wytwórczymi a końcowymi odbiorcami,

Techniki grupowej analizy kreatywnej — techniki grupowego odnajdywania rozwiązań poprzez:

„Burzę mózgów” (ang. *Brainstorming*),

Technikę nominowanych grup (ang. *Nominal group technique*),

Technikę delficką (ang. *The Delphi Technique*),
Mapę pomysłów (ang. *Idea/mind mapping*),
Diagramy pokrewieństwa (ang. *Affinity diagrams*);

Techniki podejmowania decyzji grupowych — techniki podejmowania decyzji na bazie zasady:

Jednomysłności — wszyscy muszą się zgodzić na jeden wariant,
Większości — ponad 50% procent uczestników zgadza się na jeden wariant,
Demokracji — wariant, za którym opowiada się najwięcej osób, zostaje wybrany.
Dyktatu — jedna osoba wybiera wariant;

Ankiety,

Obserwacja,

Prototypowanie.

Wyjście:

Dokument wymagań (ang. *Requirements Documentation*),
Plan zarządzania wymaganiami (ang. *Requirements Management Plan*),
Matryca śledzenia wymagań (ang. *Requirements Traceability Matrix*)
— jest to tabela łącząca wymagania z przyczyną ich powstania, taką jak potrzeba biznesowa. Tego typu podejście ułatwia monitorowanie wymagań w trakcie poszczególnych etapów cyklu życia projektu.

◆ Nazwa polska: 5.2. Definiowanie zakresu projektu

Nazwa angielska: 5.2. *Define Scope*

Wejście:

Dokument otwarcia (ang. *Project charter*),
Dokument wymagań (ang. *Requirements Documentation*),
Inna dokumentacja, która może mieć wpływ na proces, np. dokumentacja poprzedniej wersji produktu lub rejestr nauczek.

Dokument otwarcia może być traktowany jako lista życzeń zdefiniowana przez interesariuszy, a dotycząca projektu (ang. *wish-list*). Dokument ten można traktować jako pierwszą odpowiedź kierownika projektu, który definiuje tutaj, jakie funkcjonalności i w jakim kształcie można zrealizować w założonym czasie, budżecie, zasobach i jakości.

Obszar wiedzy:

Zakres.

Techniki:

Osąd eksperta — walidacja sposobu zrealizowania produktów,
Analiza produktowa — zbiór elementów i funkcjonalności, które w trakcie opisu i analizy zostają zmienione w produkt lub zbiór produktów,

Identyfikacja możliwości — próba poszukiwań innych, możliwych rozwiązań np. poprzez „burzę mózgów”,

Warsztaty.

Wyjście:

Zakres projektu (ang. *Project Scope statement*).

♦ **Nazwa polska: 5.3. Utworzenie struktury pakietów roboczych**

Nazwa angielska: 5.3. Create WBS (*Work Breakdown Structure*)

Wejście:

Zakres projektu (ang. *Project Scope Statement*),

Dokument wymagań (ang. *Requirements Documentation*),

Inna dokumentacja, która może mieć wpływ na proces, np. dokumentacja poprzedniej wersji produktu lub rejestr nauczek.

W tym procesie następuje przejście z „grupy funkcjonalności do wykonania” do „grupy produktów”. WBS jest właśnie drzewem produktów do wykonania. Jego wysoko położonymi węzłami są takie elementy jak oprogramowanie, szkolenia lub dokumentacja. Elementy te są następnie stopniowo dekomponowane na bardziej szczegółowe elementy, takie jak „aplikacja do przeglądania logów systemowych” czy „dokumentacja logów systemowych”. Ciekawym podejściem jest przyporządkowanie funkcjonalności do produktów.

Obszar wiedzy:

Zakres.

Techniki:

Dekompozycja — odnajdywanie modułów lub submodułów całości przedsięwzięcia.

Wyjście:

Struktura pakietów roboczych (ang. *Work Breakdown Structure*) wraz ze słownikiem,

Zaktualizowany zakres projektu (ang. *Project Scope Statement*) i stworzenie bazowego planu zakresu projektu (ang. *Scope Baseline*),

Zaktualizowana dokumentacja projektowa, taka jak „Dokument wymagań” (ang. *Requirements Documentation*).

♦ **Nazwa polska: 6.1. Zdefiniowanie czynności**

Nazwa angielska: 6.1. *Define Activities*

Wejście:

Bazowy planu zakresu projektu (ang. *Scope Baseline*),

Inna dokumentacja, która może mieć wpływ na proces, taka jak polityki planowania, rejestr nauczek i dobrych praktyk.

Każdy z liści w strukturze pakietów roboczych jest dekomponowany do konkretnych działań „wytwórczych”. Mogą one być rozumiane jako zadanie do wykonania w kontekście harmonogramu projektu. Zdefiniowana lista czynności nie powinna zawierać żadnych działań, które nie mają związku z wytworzeniem konkretnego produktu. W obrębie tego procesu powinny być również zdefiniowane opcjonalne i obowiązkowe kamienie milowe projektu, które będą jasno sygnalizowały wykonanie (zakończenie) pewnej grupy prac lub inne istotne momenty w trakcie trwania projektu.

Należy jasno zaznaczyć, że w obrębie tego procesu definiowane jest jedynie „co musi (powinno) być zrobione”, a dopiero w późniejszym etapie ustalane są takie szczegóły takie jak: „w jakiej kolejności”, „kto”, „jak długo” i „kiedy”.

Obszar wiedzy:

Czas.

Techniki:

Dekompozycja — rozdrobnienie dużych zadań na mniejsze,

Planowanie przyrostowe (ang. *Rolling Wave Plannig*) — w pełni przejrzyste, najbliższe w czasie elementy są planowane w szczegółach; a im bardziej niejasne i odleglejsze w czasie elementy, tym bardziej ogólne planowanie,

Wzorce, formularze i standardy,

Osąd eksperta.

Wyjście:

Lista czynności (ang. *Activity List*) wraz z atrybutami,

Lista kamieni milowych (ang. *Milestone List*).

◆ **Nazwa polska: 6.2. Szeregowanie czynności**

Nazwa angielska: 6.2. *Sequence Activities*

Wejście:

Lista czynności (ang. *Activity List*) wraz z atrybutami,

Lista kamieni milowych (ang. *Milestone List*),

Zakres projektu (ang. *Project Scope Statement*),

Inna dokumentacja, która może mieć wpływ na proces.

W tym procesie definiowane są zależności między czynnościami, czyli odpowiedź na pytanie „w jakiej kolejności” są one wykonywane.

Obszar wiedzy:

Czas.

Techniki:

Metoda Diagramów Następstw (ang. *Precedence Diagramming Method*),

Analiza zależności (ang. *Dependency Determination*),

Zastosowanie przyspieszeń i opóźnień (ang. *Applying Leads and Lags*),

Szablony harmonogramu sieciowego (ang. *Schedule Network Templates*).

Wyjście:

Diagram zależności czynności (ang. *Project Schedule Network Diagrams*),
Zaktualizowana dokumentacja projektowa, taka jak lista czynności
(ang. *ActivityList*).

♦ **Nazwa polska: 6.3. Szacowanie zasobów czynności**

Nazwa angielska: 6.3. *Estimate Activity Resources*

Wejście:

Lista czynności (ang. *Activity List*) wraz z atrybutami,
Kalendarze zasobów (ang. *Resorce Calendars*),
Inna dokumentacja, która może mieć wpływ na proces.

W tym procesie określany jest typ i ilość zasobów koniecznych do wykonania czynności. W praktyce oznacza to zarezerwowanie na potrzeby projektu odpowiednich osób i sprzętu w pewnych przedziałach czasowych. Na tej podstawie mają miejsce negocjacje z „właścicielem” zasobów i tworzona jest struktura zasobów. Następnie do zdefiniowanych czynności przyporządkowywany jest wykonawca, czyli następuje odpowiedź na pytanie „kto” wykonuje daną czynność.

Obszar wiedzy:

Czas.

Techniki:

Osąd eksperta,
Analiza alternatyw,
Publikowanie estymacji,
Estymacja od szczegółu do ogółu (ang. *Bottom-up*) — metoda estymacji wszystkich części cząstkowych, których suma daje estymację czasu wykonania zadania nadrzędnego,
Oprogramowanie do zarządzania projektami.

Wyjście:

Wymagania zasobów do czynności (ang. *Activity Resource Requirements*),
Struktura zasobów (ang. *Resource Breakdown Structure*) — hierarchia organizacyjna projektu,
Zaktualizowana dokumentacja projektowa, taka jak lista czynności (ang. *Activity List*) wraz z atrybutami, Kalendarze zasobów (ang. *Resorce Calendars*).

♦ **Nazwa polska: 6.4. Szacowanie czasu trwania czynności**

Nazwa angielska: 6.4. *Estimate Activity Durations*

Wejście:

Lista czynności (ang. *Activity List*) wraz z atrybutami,

Kalendarze zasobów (ang. *Resource Calendars*),

Inna dokumentacja, która może mieć wpływ na proces.

W procesie szacowany jest koszt wykonania każdej czynności, czyli odpowiedź na pytanie „jak długo”. Bezpośredni wpływ na te szacunki ma pula zasobów przyporządkowanych do wykonania tej czynności; planowany czas wykonania czynności i zasoby czynności powinny być ze sobą nierozdzielnie związane. Odebranie lub dodanie zasobów powinno automatycznie wpływać na konieczność wykonania nowych szacunków.

Obszar wiedzy:

Czas.

Techniki:

Osąd eksperta,

Szacunki przez analogię (ang. *Analogous estimating*) — estymacja na podstawie dotychczasowych, historycznych informacji,

Szacunki parametryczne (ang. *Parametric estimating*) — wyliczenie czasu trwania zadania na podstawie wzoru, który operuje na jednej lub kilku wagach (np. metoda punktów funkcyjnych),

Szacunki trójpunktowe (ang. *Three-point estimating*) — na bazie estymacji optymistycznej i pesymistycznej wyliczana jest estymacja optymalna za pomocą kalkulacji Pert (Optymistyczna + 4*Najprawdopodobniejsza + Pesymistyczna / 6),

Analiza rezerw jest procesem wyliczania rezerw w rezultatach estymacji. Te bufor rezerwowe oraz metoda ich wyliczenia są udokumentowane i jawne (6.3, 6.4).

Wyjście:

Szacunki czasu wykonania (ang. *Activity Duration Estimates*),

Zaktualizowane dokumenty projektowe, takie jak założenia projektowe.

♦ Nazwa polska: 6.5. Opracowanie harmonogramu

Nazwa angielska: 6.5. *Develop Schedule*

Wejście:

Lista czynności (ang. *Activity List*) wraz z atrybutami,

Diagram zależności czynności (ang. *Project Schedule Network Diagrams*),

Wymagania zasobów do czynności (ang. *Activity Resource Requirements*),

Kalendarze zasobów (ang. *Resource Calendars*),

Szacunki czasu wykonania (ang. *Activity Duration Estimates*),

Zakres projektu (ang. *Project Scope Statement*),

Inna dokumentacja, która może mieć wpływ na proces.

W procesie wszystkie elementy cząstkowe, takie jak analiza sekwencji zdarzeń czy dostępność zasobów, składane są w całość jako harmonogram projektu. Następuje odpowiedź na pytanie „kiedy”. Na końcu tego procesu widać jasno,

czy projekt ma szansę spełnienia narzuconych ograniczeń, takich jak koszt, czas czy jakość. Jeżeli wyraźnie widać, że plan i ograniczenia nijak mają się do siebie, należy unikać najłatwiejszego podejścia typu „to musi trwać krócej” lub „ograniczymy czas testów”. Takie „naginanie” rzeczywistości zemści się w sposób zwielokrotniony. W takich wypadkach jedynym rozsądnym podejściem jest zmiana sposobu realizacji projektu, ograniczenie zakresu projektu lub poszerzenie puli zasobów. Każde wyjście powoduje konieczność powtórzenia prac planistycznych.

Po wygenerowaniu szczegółowego planu projektu wskazane jest przygotowanie jego ogólnej wersji, czyli planu bazowego, który będzie zawierał pewne dodatkowe „bufory bezpieczeństwa”. Plan taki powinien zostać zaprezentowany do akceptacji interesariuszom (ang. *Stakeholders*).

Obszar wiedzy:

Czas.

Techniki:

Analiza sieciowa harmonogramu,

Metoda ścieżki krytycznej (ang. *Critical Path Method — CPM*),

Metoda łańcucha krytycznego (ang. *Critical chain method*),

Równoważenie (bilansowanie) zasobów (ang. *Resource Leveling*),

Analiza scenariuszy „co, jeśli”,

Zastosowanie przyspieszeń i opóźnień (ang. *Applying Leads and Lags*),

Optymalizacja harmonogramu,

Narzędzia do zarządzania harmonogramami (np. MS Project).

Wyjście:

Harmonogram projektu (ang. *Project Schedule*),

Plan bazowy (ang. *Schedule Baseline*),

Zaktualizowane dane związane z harmonogramem (ang. *Schedule Data*),
takie jak kamienie milowe, dokumentacja założeń i ograniczeń, alternatywne harmonogramy, harmonogramowanie ewentualnych rezerw,

Zaktualizowane dokumenty projektowe, takie jak kalendarze lub rejestr ryzyk.

♦ **Nazwa polska: 7.1. Szacowanie kosztów**

Nazwa angielska: 7.1. *Estimate Costs*

Wejście:

Zakres projektu (ang. *Scope Baseline*) włącznie ze strukturą produktową (WBS),

Harmonogram projektu (ang. *Project Schedule*),

Plan zasobów ludzkich (ang. *Human Resource Plan*),

Rejestr ryzyk (ang. *Risk Register*),

Inna dokumentacja, która może mieć wpływ na proces.

Na tym etapie szacowane są sumaryczne koszty wykonania projektu. W ramach tego procesu przeprowadzana jest analiza wykonania względem zakupu. Wraz z kolejnymi iteracjami procesu planistycznego dokładność estymacji powinna być coraz bardziej trafna i w związku z tym ryzyko jej niedokładności powinno sukcesywnie spadać. Dlatego też w początkowej fazie projektu należy zakładać szeroki margines błędu, który jednak powinien się zmniejszać.

Obszar wiedzy:

Koszt.

Techniki:

Osąd eksperta,

Szacunki przez analogię,

Szacunki parametryczne,

Szacunki od ogółu do szczegółu,

Szacunki trójpunktowe na bazie szacunku optymistycznego, pesymistycznego i najbardziej prawdopodobnego; wykonywana jest kalkulacja PERT,

Analiza rezerw — przygotowanie zapasowej puli funduszy,

Koszt jakości (ang. *Cost Of Quality*) — zapewnienie jakości jest zazwyczaj dodatkowym centrum kosztów,

Analiza ofert dostawców (ang. *Vendor Bid Analysis*).

Wyjście:

Szacunki kosztów czynności (ang. *Activity Cost Estimates*),

Założenia do szacunków (ang. *Basis of Estimates*),

Zaktualizowane dokumenty projektowe, takie jak rejestr ryzyk.

◆ Nazwa polska: 7.2. Zdefiniowanie budżetu

Nazwa angielska: 7.2. *Budget Determine*

Wejście:

Szacunki kosztów czynności (ang. *Activity Cost Estimates*),

Założenia do szacunków (ang. *Basi of Estimates*),

Zakres projektu (ang. *Scope Baseline*) włącznie ze strukturą produktową (WBS),

Kalendarze zasobów (ang. *Resorce Calendars*),

Kontrakty,

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Czynności po oszacowaniu kosztów są agregowane i tworzą bazowe budżety. Zawierają one wszystkie zaakceptowane budżety, ale nie ma w nich rezerw zarządczych. Od tej pory koszty projektowe będą mierzone względem bazowego, autoryzowanego budżetu.

Obszar wiedzy:

Koszt.

Techniki:

Kumulacja kosztów — zgodnie ze strukturą produktową (WBS) wszystkie poddrzewa są sumowane w górę,

Analiza rezerw,

Osąd eksperta,

Szacunki z uwzględnieniem danych historycznych — sposób odgórnego wyliczenia kosztów na bazie danych historycznych, np. koszt budowy wieżowca można liczyć, biorąc pod uwagę koszt jednego metra kwadratowego,

Ustalenie limitu funduszy — etap uzgodnienia możliwości finansowania projektu i oszacowanych kosztów względem zdefiniowanego zakresu projektu. Może to przykładowo powodować przełożenie części prac projektowych na okres późniejszy lub wręcz odwrotnie — zwiększenie funduszy projektowych (rzadszy przypadek ☺).

Wyjście:

Plan bazowy kosztów (ang. *Cost Performance Baseline*),

Wymagania finansowe projektu (ang. *Project Funding Requirements*) — przykładowo projekt może nie być finansowy w sposób ciągły, tylko w ramach rund finansowania (duże przelewy na określony okres prowadzenia działalności),

Zaktualizowane dokumenty projektowe, takie jak rejestr ryzyk.

♦ **Nazwa polska: 8.1. Planowanie jakości**

Nazwa angielska: 8.1. *Plan Quality*

Wejście:

Zakres projektu (ang. *Scope Baseline*) włącznie ze strukturą produktową (WBS),

Rejestr interesariuszy (ang. *Stakeholder Register*),

Plan bazowy kosztów (ang. *Cost Performance Baseline*),

Plan bazowy (ang. *Schedule Baseline*),

Rejestr ryzyk (ang. *Risk Register*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces, takie jak regulacje prawne lub historyczne bazy danych.

Planowanie jakości projektu to zdefiniowanie standardów jakościowych i sposobu ich zapewnienia w projekcie (plan zarządzania jakością), co w efekcie powinno zapewnić odpowiednią jakość procesów i wytworzonych produktów.

Obszar wiedzy:

Jakość.

Techniki:

Analiza koszt-zysk — weryfikacja, czy wskazane standardy jakościowe mogą lub powinny zostać sfinansowane,

Koszt jakości (ang. *Cost Of Quality*) — przeciwstawia koszty zapewnienia jakości kosztom niskiej jakości,

Wykresy kontrolne — definiują, czy proces zachowuje się stabilnie, tzn. w przewidzianych ramach wydajnościowych,

Benchmarking — porównywanie wytworzonych produktów do innych podobnych, aby odnaleźć nowe możliwości udoskonaleń i udostępnić bazę do porównań wydajności oraz funkcjonalności,

Projektowanie eksperymentów (ang. *Design of Experiments*) — badanie wpływu poszczególnych parametrów na poziom jakości całego produktu (procesu),

Próby statystyczne (ang. *Statistical Sampling*) — sprawdzenie części rozwiązania. Wielkość i częstotliwość prób muszą zostać zdefiniowane w planie jakości,

Diagramy przepływów (ang. *Flowcharts*),

Dodatkowe techniki planistyczne takie jak:

„Burza mózgów” (ang. *brainstorming*),

Diagramy pokrewieństwa (ang. *affinity diagrams*),

Analiza pola siły (ang. *force field analysis*),

Technika nominowanych grup (ang. *nominal group techniques*).

Wyjście:

Plan zarządzania jakością (ang. *Quality Management Plan*),

Metryki jakościowe (ang. *Quality Metrics*),

Listy weryfikujące jakość (ang. *Quality Checklists*),

Plan udoskonalania procesów (ang. *Process Improvement Plan*),

Zaktualizowane dokumenty projektowe, takie jak rejestr interesariuszy.

◆ **Nazwa polska: 9.1. Planowanie zasobów ludzkich**

Nazwa angielska: 9.1. *Develop Human Resource Plan*

Wejście:

Wymagania zasobów do czynności (ang. *Activity Resource Requirements*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Na tym etapie definiowane są odpowiedzialności poszczególnych członków zespołu, ich role oraz sposób i kierunki raportowania. Plan zarządzania zasobami ludzkimi powinien uwzględniać:

Sposób rekrutacji (zewnętrznej bądź wewnętrznej) do projektu,

Kto i kiedy będzie wymagany w projekcie,

Kryteria „zwolnienia” zasobów z projektu,

Identyfikację potrzeb szkoleniowych,

Plan premiowy,

Kwestie organizacyjne (np. zaświadczenia, przepustki itp.),

Kwestie bezpieczeństwa (kto i jak może ograniczyć zagrożenia projektu),
Oddziaływanie planów projektowych na organizację (np. poziom konsumpcji
zasobów, przewidywane koszty „ludzkie” utrzymania projektu).

Obszar wiedzy:

Zasoby ludzkie.

Techniki:

Diagramy struktur organizacyjnych i pozycji, które mogą być:

- ♦ hierarchiczne — klasyczna struktura drzewiasta,
- ♦ macierzowe, czyli takie, gdzie osoby są przyporządkowane do zadań poprzez takie słowa jak „wykonawca”, „odpowiedzialny”, „konsultacja”, „informowany” (analogicznie do tabeli RACI z metodyki COBIT),
- ♦ tekstowe — opis ról w organizacji;

Sieci społecznościowe — formalna i nieformalna interakcja w organizacji
(np. *lobbying*, *road shows* itd.),

Teorie organizacji — opisujące sposób zachowania ludzi w większych zespołach:

- ♦ pięciopoziomową hierarchię potrzeb Masłowa — obejmuje ona potrzeby fizjologiczne (musisz żyć), bezpieczeństwa (czuć się bezpiecznie), afiliacji (przynależć do zespołu), szacunek i uznanie, samorealizacja (wszystko jest OK.),
- ♦ teorię motywacji Herzberga, która mówi, że zazwyczaj bardziej niż pieniądze motywują lepsze warunki pracy,
- ♦ teorię X i Y McGregorego, opisującą osoby, które chcą być prowadzone (X), i te, które chcą realizować cele (Y).

Wyjście:

Plan zasobów ludzkich (ang. *Human Resource Plan*), który definiuje:

Listę ról i odpowiedzialności (rola, autoryzacje, odpowiedzialności, kompetencje) (ang. *Roles and responsibilities*),

Hierarchię organizacyjną projektu (ang. *Project organization charts*),

Plan zarządzania personelem (ang. *Staffing Management Plan*).

♦ Nazwa polska: 10.2. *Planowanie komunikacji*

Nazwa angielska: 10.2. *Plan Communications*

Wejście:

Rejestr interesariuszy (ang. *Stakeholder Register*),

Strategia zarządzania interesariuszami (ang. *Stakeholder Management Strategy*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

W procesie tym definiowane są potrzeby informacyjne interesariuszy projektu (ang. *stakeholders*) i na tej podstawie tworzony jest plan komunikacyjny, który określa sposób i zakres informowania interesariuszy o postępach w projekcie.

Dobry plan komunikacyjny istotnie zwiększa szanse powodzenia projektu i pozwala uniknąć nieporozumień na różnych szczeblach decyzyjnych.

Obszar wiedzy:

Komunikacja.

Techniki:

Analiza wymagań komunikacyjnych — metoda zdefiniowania kto, co i kiedy wiedzieć powinien na temat projektu, ustanowienie kanałów komunikacyjnych, których ilość powinna wynosić około $n*(n-1)/2$, gdzie n to liczba interesariuszy (dla 10 interesariuszy jest to 45 potencjalnych kanałów),

Techniki komunikacji — metody komunikacyjne mogą być bardzo różne i dlatego należy wziąć pod uwagę takie elementy jak pilność potrzeb informacyjnych, dostępność technologii, oczekiwany personel projektowy, długość projektu, środowisko projektowe,

Modele komunikacji — które definiują, że w momencie komunikacji wytwarza się pewien szum komunikacyjny i dlatego istotne jest osiągnięcie dwukierunkowego zrozumienia tematu na różnych poziomach abstrakcji,

Metody komunikacji — opisują trzy główne mechanizmy: interakcyjny (np. spotkania, telekonferencje), popchnij (np. e-maile, raporty), pociągnij (np. intranet, e-learning).

Wyjście:

Plan zarządzania komunikacją (ang. *Communications Management Plan*).

♦ **Nazwa polska: 11.1. Planowanie zarządzania ryzykiem**

Nazwa angielska: 11.1. *Plan Risk Management*

Wejście:

Zakres projektu (ang. *Project Scope Statement*),

Plan zarządzania kosztami (ang. *Cost Management Plan*),

Plan zarządzania harmonogramem (ang. *Schedule Management Plan*),

Plan zarządzania komunikacją (ang. *Communications Management Plan*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

W procesie definiowane są podejście, plan i wykonanie czynności związanych z zarządzaniem ryzykami projektowymi. W praktyce oznacza to zdefiniowanie procedury zarządzania ryzykami.

Obszar wiedzy:

Ryzyko.

Techniki:

Planowanie spotkań i analiza — technika grupowego zarządzania ryzykami.

Wyjście:

Plan zarządzania ryzykami (ang. *Risk Management Plan*) — zawiera opis ilościowej analizy ryzyk, która umożliwia wyliczenie siły ryzyka na bazie wpływu na koszt, czas, zakres i jakość.

♦ **Nazwa polska: 11.2. Identyfikacja ryzyka**

Nazwa angielska: 11.2. *Identify Risks*

Wejście:

Plan zarządzania ryzykami (ang. *Risk Management Plan*),
Szacunki kosztów czynności (ang. *Activity Cost Estimates*),
Szacunki czasu wykonania (ang. *Activity Duration Estimates*),
Zakres projektu (ang. *Scope Baseline*) włącznie ze strukturą produktową (WBS),
Rejestr interesariuszy (ang. *Stakeholder Register*),
Plan zarządzania kosztami (ang. *Cost Management Plan*),
Plan zarządzania harmonogramem (ang. *Schedule Management Plan*),
Plan zarządzania jakością (ang. *Quality Management Plan*),
Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

W ramach projektu można zarządzać jedynie znanymi ryzykami projektowymi i dlatego tak istotne jest zidentyfikowanie maksymalnie dużej ilości możliwych zagrożeń. Proces ten powinien być iteracyjny, gdyż wiedza zespołu o projekcie ulega stałej ewolucji w trakcie jego wykonania.

Obszar wiedzy:

Ryzyka.

Techniki:

Przeglądy dokumentacji — polegające na weryfikacji jakości planów oraz spójność planów z wymaganiami i założeniami projektowymi,
Technika zbierania informacji — burze mózgów, technika delficka, wywiady, identyfikacja źródła problemów,
Lista kontrolna — sprawdzanie ryzyk projektowych na bazie wiedzy z dotychczas zrealizowanych projektów,
Analiza założeń,
Techniki diagramowania — diagramy przyczynowo-skutkowe, przepływy procesowe i systemowe, diagramy wpływów wymienione w „8.1. Planowanie jakości”,
Analiza SWOT polegająca na analizie silnych i słabych stron, możliwości oraz zagrożeń,
Osąd eksperta.

Wyjście:

Rejestr ryzyk (ang. *Risk Register*).

◆ Nazwa polska: 11.3. Jakościowa analiza ryzyka

Nazwa angielska: 11.3. *Perform Qualitative Risk Analysis*

Wejście:

Rejestr ryzyk (ang. *Risk Register*),

Plan zarządzania ryzykami (ang. *Risk Management Plan*),

Zakres projektu (ang. *Project Scope Statement*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Wiele ze zidentyfikowanych ryzyk może wymagać dodatkowej analizy w celu stwierdzenia prawdopodobieństwa ich wystąpienia oraz wpływu na projekt.

Na bazie tych informacji definiowany jest priorytet ryzyka.

Obszar wiedzy:

Ryzyka.

Techniki:

Oszacowanie prawdopodobieństwa i wpływu ryzyka,

Tabela prawdopodobieństwa i wpływu ryzyka — metoda mierzenia siły ryzyka,

Oszacowanie jakości danych,

Kategoryzacja ryzyk,

Oszacowanie pilności ryzyk,

Osąd eksperta.

Wyjście:

Zaktualizowany rejestr ryzyk (ang. *Risk Register Updates*).

◆ Nazwa polska: 11.4. Ilościowa analiza ryzyka

Nazwa angielska: 11.4. *Perform Quantitative Risk Analysis*

Wejście:

Rejestr ryzyk (ang. *Risk Register*),

Plan zarządzania ryzykami (ang. *Risk Management Plan*),

Plan zarządzania kosztami (ang. *Cost Management Plan*),

Plan zarządzania harmonogramem (ang. *Schedule Management Plan*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Takie parametry jak prawdopodobieństwo wystąpienia czy wpływ na projekt są transformowane z wartości opisowych do wartości liczbowych. Następnie szacowany jest sumaryczny wpływ wszystkich ryzyk na cały projekt. Tego typu podejście umożliwia utworzenie jednej, mierzalnej metryki, która zdefiniuje łączne ryzyko całego projektu.

Obszar wiedzy:

Ryzyka.

Techniki:

Zbieranie danych i techniki reprezentacyjne, takie jak wywiady i wyliczanie dystrybucji prawdopodobieństwa na bazie osądów eksperta,

Ilościowa analiza ryzyka i techniki modelowania:

analiza czułości, która wskazuje najbardziej niepewne warianty sytuacji, w przypadku których niepewność przyszłej sytuacji jest największa (np. największy zakres możliwych kosztów realizacji lub czasu realizacji zadania),

analiza za pomocą techniki planowanej wartości finansowej (ang. *Expected Monetary Value*) — polega na rozpisaniu możliwych scenariuszy finansowych w drzewo decyzyjne i oszacowanie prawdopodobieństwa wystąpienia każdego z nich; węzły drzewa opisują sumaryczną wartość finansową i dzięki temu możliwe jest podjęcie obiektywnej decyzji strategicznej,

modelowanie i symulacje;

Osąd eksperta.

Wyjście:

Zaktualizowany rejestr ryzyk (ang. *Risk Register Updates*).

♦ **Nazwa polska: 11.5. Planowanie reakcji na ryzyko**

Nazwa angielska: 11.5. *Plan Risk Response*

Wejście:

Rejestr ryzyk (ang. *Risk Register*),

Plan zarządzania ryzykami (ang. *Risk Management Plan*).

Istnieje wiele różnych sposobów zarządzania ryzykiem i w każdym projekcie możliwe są różne opcje redukcji zagrożeń dotyczących celów projektu. Odpowiedź na ryzyko powinna być:

- ♦ adekwatna do ważności ryzyka,
- ♦ efektywna kosztowo i czasowo,
- ♦ realistyczna w kontekście danego projektu,
- ♦ uzgodniona ze wszystkimi zainteresowanymi stronami,
- ♦ przynależna jednej osobie.

Obszar wiedzy:

Ryzyka.

Techniki:

Strategie dla ryzyk negatywnych i zagrożeń: unikanie, transferowanie (na strony trzecie), redukcja, akceptacja,

Strategie dla ryzyk pozytywnych i możliwości: eksploatacja (uzyskanie pozytywnych efektów), współdzielenie (przekazanie pozytywów stronie trzeciej), rozszerzenie (uzyskanie większej ilości pozytywów), akceptacja,

Strategia pakietu odpowiedzi — różne odpowiedzi, zależne od warunków pojawienia się ryzyka,

Osąd eksperta.

Wyjście:

Zaktualizowany rejestr ryzyk (ang. *Risk Register Updates*),

Decyzje kontraktowe spowodowane ryzykiem (ang. *Risk-Related Contract Decisions*),

Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*),

Zaktualizowane dokumenty projektowe.

♦ **Nazwa polska: 12.1. Planowanie dostaw**

Nazwa angielska: 12.1. *Plan Procurements*

Wejście:

Zakres projektu (ang. *Scope Baseline*) włącznie ze strukturą produktową (WBS),

Dokument wymagań (ang. *Requirements Documentation*),

Porozumienia (ang. *Teaming Agreement*) — porozumienia o partnerstwie lub joint venture,

Rejestr ryzyk (ang. *Risk Register*),

Decyzje kontraktowe spowodowane ryzykiem (ang. *Risk-Related Contract Decisions*),

Wymagania zasobów do czynności (ang. *Activity Resource Requirements*),

Harmonogram projektu (ang. *Project Schedule*),

Szacunki kosztów czynności (ang. *Activity Cost Estimates*),

Plan bazowy kosztów (ang. *Cost Performance Baseline*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

W trakcie procesu definiowane jest, co, kiedy i jak powinno zostać zakupione lub uzyskane. Na tym etapie podjęte zostają również decyzje, które produkty, usługi lub rezultaty mogą zostać uzyskane z zewnątrz organizacji, a które opłaca się wykonać wewnątrz organizacji (decyzja: zrobić czy kupić). Uwzględnia się tu kwestie ryzyk związanych z obiema opcjami. Często wiele ryzyk można wydelegować na zewnątrz poprzez zakup gotowej usługi.

Obszar wiedzy:

Zaopatrzenie.

Techniki:

Analiza „zrobić czy kupić”,

Ocena eksperta,

Typy kontraktów:

- ♦ Kontrakt o ustalonej cenie (ang. *Fixed price*):
 - ♦ standardowy kontrakt o ustalonej cenie (ang. *Firm Fixed Price Contracts FFP*),
 - ♦ kontrakt o ustalonej cenie z marginesem (ang. *Fixed Price Incentive Fee Contracts FPIF*),
 - ♦ kontrakt o ustalonej cenie z opcją rewaloryzacji (ang. *Fixed Price with Economic Price Adjustment Contracts FP-EPA*);
- ♦ Kontrakty zwrotu kosztów (ang. *Cost-reimbursable*):
 - ♦ koszt z ustaloną opłatą (ang. *Cost-Plus-Fixed-Fee CPFF*),
 - ♦ koszt z marginesem (ang. *Cost-Plus-Incentive-Fee CPIF*),
 - ♦ koszt z premią (ang. *Cost-Plus-Award-Fee CPAF*);
- ♦ Kontrakt typu czas i materiał (ang. *Time and Material*).

Wyjście:

Plan zarządzania zaopatrzeniem (ang. *Procurement Management Plan*),
Zdefiniowany zakres prac kontraktowych (ang. *Contract Statement of Work*),
Udokumentowane decyzje „zrobić czy kupić”,
Dokumenty zaopatrzeniowe (ang. *Procurement Documents*),
Kryteria selekcji źródła zaopatrzenia,
Żądania zmian projektowych.

Procesy realizacji to wykonanie projektu.

♦ **Nazwa polska: 4.3. Kierowanie i zarządzanie realizacją projektu**

Nazwa angielska: 4.3. *Direct and Manage Project Execution*

Wejście:

Plan zarządzania projektem (ang. *Project Management Plan*),
Zaakceptowane propozycje zmian (ang. *Approved Change Request*),
Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

W ramach tego procesu wykonywane są wszelkie operacyjne czynności zarządcze. Czynności te wynikają z przygotowanego planu zarządzania projektem. W ramach tego procesu zbierane są również informacje niezbędne do oszacowania wydajności wykonywanej pracy.

Bardzo często w trakcie realizacji projektu pojawia się konieczność wykonania zmian. Jeżeli zostaną zaakceptowane, mogą powodować:

- ♦ Działania naprawcze — uruchamiane w przypadku rozbieżności realizacji względem planów projektu; mają na celu zmianę sposobu realizacji prac projektowych,
- ♦ Działania prewencyjne — udokumentowane działania mające na celu redukcję prawdopodobieństwa potencjalnych negatywnych konsekwencji związanych z ryzykami projektowymi,

- ◆ Naprawę defektów — udokumentowana identyfikacja defektu może powodować jego naprawę lub kompletne zastąpienie komponentu rozwiązania.

Obszar wiedzy:

Integracja.

Techniki:

Osąd eksperta,

System informatyczny do zarządzania projektem, np. systemy typu EPM lub PPM (*Enterprise Project Management* lub *Project Portfolio Management*).

Wyjście:

Produkty (ang. *Deliverables*),

Informacje o wydajności pracy (ang. *Work Performance Information*),

Żądania zmian (ang. *Change Requests*),

Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*),

Zaktualizowane dokumenty projektowe.

◆ **Nazwa polska: 8.2. Zapewnienie jakości**

Nazwa angielska: 8.2. *Perform Quality Assurance*

Wejście:

Plan zarządzania projektem (ang. *Project Management Plan*),

Metryki jakościowe (ang. *Quality Metrics*),

Informacje o wydajności pracy (ang. *Work Performance Information*),

Miary kontroli jakości (ang. *Quality Control Measurements*).

To proces systematycznego wykonania zaplanowanych czynności, mających na celu zapewnienie zdefiniowanej jakości projektu. W ramach tego procesu zawarta jest również pętla stałego doskonalenia procesów, co może powodować wygenerowanie nowych żądań zmian. Żądania te mogą wynikać również z bieżącej kontroli zgodności wykonanych prac z przyjętym planem.

Obszar wiedzy:

Jakość.

Techniki:

Techniki planowania i zapewnienia jakości wymienione w pozostałych procesach jakościowych,

Audyty jakości — czynności weryfikacyjne zgodne z politykami, procesami i procedurami projektowymi oraz organizacyjnymi; służą do sprawdzenia, czy zaakceptowane żądania zmian, działania naprawcze i prewencyjne oraz naprawy defektów zostały wykonane poprawnie,

Analiza procesu — weryfikuje zgodność procesu z planem udoskonalenia procesu.

Wyjście:

Aktualizacja standardów organizacyjnych, takich jak standardy jakościowe,
Nowe lub uaktualnione żądania zmian (ang. *Change Requests*),
Aktualizacja planu zarządzania projektem (ang. *Project Management Plan Updates*),
Aktualizacja dokumentacji projektowej.

♦ **Nazwa polska: 9.2. Organizacja zespołu projektowego**

Nazwa angielska: 9.2. *Acquire Project Team*

Wejście:

Lista ról i odpowiedzialności (ang. *Roles and Responsibilities*),
Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

W tym procesie zapewniana jest dostępność osób dla projektu. Kierownik projektu powinien skutecznie wpłynąć na osoby, które mogą zapewnić takich pracowników. Niezapewnienie tych zasobów może mieć negatywny wpływ na harmonogram projektu, budżet, satysfakcję klienta, jakość i ryzyka. W skrajnym przypadku może to wręcz spowodować rezygnację z projektu. W związku tym, wszelkie ewentualne sytuacje braku zasobów powinny zostać przewidziane w planie, który musi zawierać „warianty awaryjne”, takie jak wynajęcie zasobów zewnętrznych.

Obszar wiedzy:

Zasoby ludzkie.

Techniki:

Rezerwacja — zaplanowanie do przodu czasu konkretnych osób na potrzeby projektu,
Negocjacje — rozmowy w obrębie organizacji oraz poza nią w celu uzyskania optymalnego rozwiązania na przestrzeni wielu realizowanych projektów,
Nabycie — pozyskanie dla projektu dodatkowych osób spoza organizacji,
Zespoły wirtualne — metoda organizowania zespołów, które nie znajdują się fizycznie w jednym miejscu; tego typu rozwiązanie bardzo dobrze sprawdza się przy projektach typu open source,
Uruchomienie osób — integracja nowej osoby przeznaczonej do istniejącego projektu.

Wyjście:

Przyporządkowanie personelu projektowi (ang. *Project Staff Assignments*),
Kalendarze zasobów (ang. *Resource Calendars*),
Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*).

♦ **Nazwa polska: 9.3. Rozwój zespołu projektowego**

Nazwa angielska: 9.3. *Develop Project Team*

Wejście:

Przyporządkowanie personelu projektowi (ang. *Project Staff Assignments*),
Plan zarządzania projektem (ang. *Project Management Plan*),
Kalendarze zasobów (ang. *Resource Calendars*).

Proces organizowania zespołu polega na systematycznym budowaniu kompetencji i umiejętności każdej osoby oddzielnie i wszystkich razem. Z jednej strony, oznacza to właściwe motywowanie do rozwoju poszczególnych osób (m.in. realizacja planu szkoleń), a z drugiej strony, właściwe moderowanie komunikacji pomiędzy członkami zespołu. Dobrze działający zespół może więcej niż suma możliwości jednostek. Jeżeli w zespole panuje atmosfera wzajemnego zaufania i wspólnego dążenia do wytyczonego celu, znacząco rosną szanse powodzenia projektu. W praktyce proces ten oznacza również rozwiązywanie konfliktów. Bieżący stan zespołu powinien być regularnie badany i dokumentowany w ocenie wydajności zespołu.

Obszar wiedzy:

Zasoby ludzkie.

Techniki:

Umiejętności „miękkie” — umiejętności polegające na właściwym moderowaniu komunikacji w zespole; są „miksura” złożoną z empatii, umiejętności wywierania wpływu, kreatywności i pracy zespołowej; mogą znacząco zredukować liczbę problemów projektowych i zwiększyć wydajność zespołu,

Szkolenia,

Budowanie zespołu (ang. *Team-Building Activities*) — wszelkie działania, które zwiększają poczucie przynależności do grupy. Technika ta opisuje m.in. pięć etapów budowania zespołu:

Formowanie (ang. *Forming*) to faza przypisania osób do projektu i formalnego przypisania ról,

Uruchamianie (ang. *Storming*) to faza uruchomienia prac projektowych i pierwszych decyzji technologicznych; zespół nie jest jeszcze zintegrowany i mogą pojawić się destrukcyjne różnice zdań,

Normowanie (ang. *Norming*) to faza, w której wytwarzają się pewne nawyki i przyzwyczajenia; członkowie zespołu zaczynają sobie wzajemnie ufać,

Wykonywanie (ang. *Performing*) to faza, w której zespół zaczyna działać sprawnie i staje dobrze zorganizowaną jednostką,

Zawieszanie (ang. *Adjourning*) to faza, w której zespół kończy prace projektowe;

Podstawowe zasady (ang. *Ground rules*) — to swoisty kodeks etyczny, który definiuje oczekiwane zachowania członków zespołu; wczesne, wspólne ustalenie takich zasad przez cały zespół buduje poczucie zespołowości; jedną z takich zasad może być nieprzeklinanie: osoba, która złamie taką zasadę, może podlegać drobnej karze pieniężnej; jeżeli jednak całemu zespołowi to nie przeszkadza, nie ma konieczności ustalania takiej zasady ☺,

Lokalizacja „po sąsiedzku” (ang. *co-location*) — zasada umiejscowienia zespołu w jednej, fizycznej lokalizacji,

Wyróżnienia i nagrody — mechanizmy motywacyjne.

Wyjście:

Ocena wydajności zespołu (ang. *Team Performance Assessments*).

♦ **Nazwa polska: 9.4. Zarządzanie zespołem projektowym**

Nazwa angielska: 9.4. *Manage Project Team*

Wejście:

Przyporządkowanie personelu projektowi (ang. *Project Staff Assignments*):

Plan zarządzania projektem (ang. *Project Management Plan*),

Ocena wydajności zespołu (ang. *Team Performance Assessment*),

Raporty wydajnościowe (ang. *Performance Reports*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Proces ten koncentruje się na obserwowaniu wydajności członków zespołu i zapewnieniu informacji zwrotnej o funkcjonowaniu tego zespołu. W ramach procesu rozwiązywane są również konflikty personalne, które mogą powodować zmiany w planach zarządzania zasobami ludzkimi.

Proces wymaga specyficznych umiejętności miękkich, po to aby stworzyć efekt synergii i spowodować, że $2 + 2 = 5$, czyli aby suma wartości jednostek była mniejsza od wartości całego zespołu.

Obszar wiedzy:

Zasoby ludzkie.

Techniki:

- ♦ Obserwacja i konwersacja — obserwacja pracy operacyjnej i nastawienia członków zespołu,
- ♦ Ocena wydajności projektu — mierzenie wydajności poszczególnych członków zespołu z uwzględnieniem analizy rzeczywistych ról i odpowiedzialności poszczególnych osób w projekcie; w ramach tej techniki tworzona jest personalna, konstruktywna informacja zwrotna oraz ustalane są personalne cele rozwojowe; większy lub mniejszy stopień formalizacji zależy od uwarunkowań organizacyjnych i projektu,
- ♦ Zarządzanie konfliktami — konflikty są często nieuniknione w środowisku projektowym; ich przyczyny mogą być tak różne, jak sposoby ich rozwiązywania; osoby znajdujące się w konflikcie powinny najpierw same spróbować go rozwiązać, a dopiero w przypadku nieudanej próby zaangażowane mogą zostać osoby trzecie; wymienione są następujące techniki zarządzania konfliktami:
 - ♦ Unikanie — zapobieganie potencjalnym lub bieżącym obszarom konfliktowym,
 - ♦ Dostosowanie — wzmocnienie obszarów porozumienia, a nie obszarów różnic,

- ◆ Kompromis — poszukiwanie rozwiązań, które obu stronom dadzą pewien poziom satysfakcji,
- ◆ Wymuszenie — promowanie jednej sytuacji, a nie innej (sytuacja typu wygrana-przegrana),
- ◆ Współpraca — zaangażowanie innych osób w celu przedstawienia innego punktu widzenia i wypracowania uzgodnienia,
- ◆ Konfrontacja — potraktowanie konfliktu jako problemu do rozwiązania poprzez analizę alternatyw i zastosowanie metody „kija i marchewki”;
- ◆ Rejestr zdarzeń (ang. *Issue log*),
- ◆ Umiejętności miękkie, które są kluczowe dla właściwego prowadzenia zespołu, wpływania na cudze opinie oraz efektywnego podejmowania decyzji.

Wyjście:

Różne techniki organizacyjne, takie jak informacje historyczne, rejestr nauczek, szablony,

Żądanie zmiany (ang. *Change Request*),

Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*).

◆ Nazwa polska: 10.3. *Dystrybucja informacji*

Nazwa angielska: 10.2. *Distribute Information*

Wejście:

Plan zarządzania projektem (ang. *Project Management Plan*),

Raporty wydajnościowe (ang. *Performance Reports*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Proces ten to realizacja planu informowania interesariuszy (ang. *stakeholders*) o postępie projektu i odpowiadania na zapytania bieżące. Głównym celem jest tutaj udostępnienie informacji w przewidzianym zakresie i czasie. Efektywna wymiana informacji powoduje konieczność zastosowania różnych technik:

- ◆ Nadawca-odbiorca — system informacji zwrotnej,
- ◆ Wybór medium stosownego do potrzeb — kiedy lepiej porozmawiać, a kiedy lepiej przyjąć formalny tryb wymiany informacji,
- ◆ Styl pisarski — strona czynna czy bierna, jakie winny być struktury zdań i dobór słów,
- ◆ Zarządzanie spotkaniami — agenda spotkania i zarządzanie konfliktami,
- ◆ Techniki prezentacyjne,
- ◆ Techniki wspierające.

Obszar wiedzy:

Komunikacja.

Techniki:

Metody komunikacji — spotkania indywidualne i grupowe, wideo- i audiokonferencje oraz inne metody komunikacji zdalnej,

Narzędzia do dystrybucji informacji, takie jak dystrybucja dokumentów papierowych i elektronicznych.

Wyjście:

Zaktualizowana dokumentacja organizacyjna, taka jak rejestr nauczerek.

♦ **Nazwa polska: 10.4. Zarządzanie oczekiwaniami** interesariuszy

Nazwa angielska: 10.4. *Manage Stakeholders Expectations*

Wejście:

Rejestr interesariuszy (ang. *Stakeholder Register*),

Strategia zarządzania interesariuszami (ang. *Stakeholder Management Strategy*),

Plan zarządzania projektem (ang. *Project Management Plan*),

Rejestr zgłoszeń (ang. *Issue Log*),

Rejestr zmian (ang. *Change Log*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Proces ten ma za zadanie zapewnić komunikację i pracę zgodną z potrzebami interesariuszy w celu realizacji ich potrzeb. Jest to możliwe dzięki aktywnemu zarządzaniu ich oczekiwaniami oraz wyjaśnianiu ich wątpliwości i obaw. Aktywni interesariusze znacząco zwiększają szanse powodzenia projektu.

Obszar wiedzy:

Komunikacja.

Techniki:

Metody komunikacji,

Umiejętności interpersonalne, takie jak budowa zaufania, rozwiązywanie konfliktów, aktywne słuchanie i przezwyciężanie oporu przed zmianą,

Umiejętności zarządcze, takie jak umiejętności prezentacyjne, negocjacyjne i pisarskie.

Wyjście:

Zaktualizowana dokumentacja organizacyjna, taka jak rejestr nauczerek,

Żądanie zmiany (ang. *Change Requests*),

Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*),

Zaktualizowane dokumenty projektowe.

♦ **Nazwa polska: 12.2. Wybranie dostaw**

Nazwa angielska: 12.2. *Conduct Procurements*

Wejście:

Plan zarządzania projektem (ang. *Project Management Plan*),

Dokumenty zaopatrzeniowe (ang. *Procurement Documents*),

Kryteria selekcji źródła zaopatrzenia,

Lista wybranych sprzedawców — organizacyjna lista wybranych sprzedawców, na podstawie dotychczasowych doświadczeń,

Oferty sprzedaży,

Dokumentacja projektowa,

Udokumentowane decyzje „zrobić czy kupić”.

Po wygenerowaniu zapytań o ofertę zbierane są informacje od potencjalnych sprzedawców i wybierany jest jeden z nich, z którym podpisuje się kontrakt. Wybór dostawcy odbywa się na podstawie wcześniej zdefiniowanych kryteriów wyboru.

Możliwa jest więcej niż jedna iteracji komunikacji ze sprzedawcami poprzez np. wstępną selekcję krótkiej listy sprzedawców (ang. *short list*), którzy spełniają postawione wymagania.

Obszar wiedzy:

Zaopatrzenie.

Techniki:

Tryb przetargowy (ang. *Bidder Conferences*) — zapewnienie, że wszyscy sprzedawcy mają takie same informacje o zamówieniu i żaden z nich nie jest traktowany preferencyjnie; kwestia ta dotyczy również sesji pytań i odpowiedzi od poszczególnych sprzedawców (muszą być dostępne dla wszystkich),

Technika oceny propozycji — przypadek, w którym wybór oferty bazuje na odpowiedzi sprzedawcy na zdefiniowane kryteria wyboru, a siłę oferty można wyliczyć,

Niezależna ocena — kupujący przygotowuje swoje niezależne szacunki, które mają być miarą rynkowej wartości zakupu (ang. *benchmarking*); znaczące różnice w wycenach ofert sprzedawców oznaczają, że mamy do czynienia z nieporozumieniem; może źle sformułowano zapytania lub sprzedawcy ich nie zrozumieli,

Ogłoszenia w prasie,

Internet — wiele usług i komponentów obecnie można zakupić jako gotowy produkt,

Negocjacje zakupów z konkretnym sprzedawcą.

Wyjście:

Lista wybranych sprzedawców (ang. *Selected Sellers*) — sprzedawcy, którzy wygrali proces oceny ofert; w przypadku bardzo złożonych zakupów wymagana może być dodatkowa akceptacja na wyższych szczeblach,

Kontrakt zakupowy (ang. *Procurement Contract Award*),

Kalendarze zasobów (ang. *Resource Calendars*),

Żądanie zmiany (ang. *Change Requests*),

Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*),

Zaktualizowane dokumenty projektowe.

Procesy kontroli to odrębna grupa procesów odpowiedzialna za weryfikację przebiegu projektu i monitorowanie ewentualnych odstępstw od planów oraz uruchomienie w miarę potrzeb działań zapobiegawczych i (lub) naprawczych.

♦ **Nazwa polska: 4.4. Monitorowanie i nadzór nad pracami projektu**

Nazwa angielska: 4.4. *Monitor and Control Project Work*

Wejście:

Plan zarządzania projektem (ang. *Project Management Plan*),

Informacje o wydajności (ang. *Performance Reports*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Ten proces to proces kontrolingowy nadrzędny w stosunku do wszystkich pozostałych; obejmuje kwestie weryfikacyjne w trakcie inicjacji, wykonania oraz zamknięcia projektu zgodnie z planem zarządzania projektem. Monitorowanie projektu zawiera działania związane ze zbieraniem, mierzeniem i rozpowszechnianiem informacji na temat wydajności projektu oraz z oceną miar i trendów w celu ulepszania procesów projektowych. Krótko mówiąc, proces ten porównuje rzeczywisty stan projektu z planowanym (ang. *actual versus budget*), umożliwia wykrywanie trendów i planowanie działań korekcyjnych, które pozwolą na osiągnięcie celów projektowych.

Obszar wiedzy:

Integracja.

Techniki:

Osąd eksperta.

Wyjście:

Żądania zmian (ang. *Requested Changes*),

Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*),

Zaktualizowane dokumenty projektowe.

♦ **Nazwa polska: 4.5. Zintegrowane zarządzanie zmianami**

Nazwa angielska: 4.5. *Integrated Change Control*

Wejście:

Plan zarządzania projektem (ang. *Project Management Plan*),

Informacje o wydajności pracy (ang. *Work Performance Information*),

Żądania zmian (ang. *Requested Changes*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

W procesie tym przeglądane są wszystkie żądania zmian, które są akceptowane lub odrzucane. W przypadku pozytywnej decyzji analizowane są i nanoszone konieczne zmiany w różnych planach projektowych, takich jak harmonogram, budżet, specyfikacja techniczna lub plan jakości.

Większe zmiany mogą wymagać akceptacji interesariuszy (ang. *stakeholders*).

W obszarze tego procesu mieści się również część zarządzania konfiguracją, czyli:

- ◆ Identyfikacja — umożliwia odnalezienie konfiguracji i dokumentacji dla danego produktu oraz zapewnia, że zmiany są zarządzane w sposób wyłumaczalny,
- ◆ Monitorowanie stanu — udostępnia stosowne informacje na temat konfiguracji, takie jak status zaproponowanych zmian konfiguracji,
- ◆ Weryfikacja i audyt — zapewnienie, że konfiguracja jest właściwa w stosunku do dokumentacji, a jej zmiany konfiguracji są właściwie zarządzane.

Obszar wiedzy:

Integracja.

Techniki:

Osąd eksperta,

Spotkania kontrolujące zmianę — w trakcie tych spotkań stosowne gremium analizuje wpływające żądania zmian i akceptuje je lub odrzuca. Decyzje te mają charakter udokumentowany i wraz z informacjami o działaniach następczych są dostępne dla interesariuszy projektu.

Wyjście:

Zaktualizowany status żądań zmian (ang. *Change Request Status Updates*),

Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*),

Zaktualizowane dokumenty projektowe.

◆ **Nazwa polska: 5.4. Weryfikacja zakresu**

Nazwa angielska: 5.4. *Verify Scope*

Wejście:

Plan zarządzania projektem (ang. *Project Management Plan*) włącznie ze strukturą produktową (WBS),

Dokument wymagań (ang. *Requirements Documentation*),

Matryca śledzenia wymagań (ang. *Requirements Traceability Matrix*),

Sprawdzone produkty (ang. *Validated Deliverables*).

Weryfikacja zakresu jest sformalizowaną akceptacją wykonanego produktu. Polega ona m.in. na przeglądzie wykonanych prac dokonywanym wraz z interesariuszami. Proces ten powinien odbyć się po procesach kontroli jakościowej, które mają zapewnić, że projekt jest wykonany zgodnie z dokumentacją projektową, taką jak plan jakości.

Obszar wiedzy:

Zakres.

Techniki:

Inspekcja — weryfikacja, czy wykonane prace i ich rezultaty są zgodne z wymaganiami i kryteriami akceptacji (5.4).

Wyjście:

Zaakceptowane produkty (ang. *Accepted Deliverables*),

Żądania zmian (ang. *Change Requests*),

Zaktualizowane dokumenty projektowe.

♦ Nazwa polska: 5.5. Kontrola zakresu

Nazwa angielska: 5.5. *Scope Control*

Wejście:

Plan zarządzania projektem (ang. *Project Management Plan*),

Informacje o wydajności pracy (ang. *Work Performance Information*),

Dokument wymagań (ang. *Requirements Documentation*),

Matryca śledzenia wymagań (ang. *Requirements Traceability Matrix*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Kontrola zakresu to proces sprawdzający wprowadzanie zmian do projektu. Pewna jest tylko zmiana i dlatego plan bazowy może ulec modyfikacjom, ale muszą one przebiegać w sposób kontrolowany. Każda zmiana musi zostać zarejestrowana i przejść przez formalny proces analizy, akceptacji i realizacji. Tylko wtedy można mieć pewność, że modyfikacje zostaną wykonane we wszystkich powiązanych obszarach. Nieudokumentowane zmiany nie są „zgodne ze sztuką” i mogą powodować rozmycie zakresu projektu (ang. *project scope creep*).

Obszar wiedzy:

Zakres.

Techniki:

Analizy wariantowe — mają na celu zdefiniowanie, jak wielka jest zmiana w stosunku do pierwotnego planu bazowego, oraz podjęcie decyzji, czy planowane jest działanie korekcyjne, czy prewencyjne.

Wyjście:

Miary wydajności pracy (ang. *Work Performance Measurements*)

— przykładowo może to być miara weryfikująca rzeczywistą wydajność w stosunku do planowanej,

Zaktualizowane dane organizacyjne,

Żądania zmian (ang. *Change Requests*),

Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*),

Zaktualizowane dokumenty projektowe.

◆ **Nazwa polska: 6.6. Kontrola harmonogramu**

Nazwa angielska: 6.6. *Control Schedule*

Wejście:

Plan zarządzania projektem (ang. *Project Management Plan*),
Harmonogram projektu (ang. *Project Schedule*),
Informacje o wydajności pracy (ang. *Work Performance Information*),
Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Jest to proces monitorowania statusu projektu i aktualizacji jego postępu. Proces ten zawiera w sobie również zarządzanie zmianami w stosunku do bazowego harmonogramu. Proces ten wiąże się:

- ◆ Ze zdefiniowaniem obecnego stanu harmonogramu projektowego,
- ◆ Z wpływem na czynniki powodujące zmiany harmonogramu (np. uczestnictwo przy podejmowaniu decyzji),
- ◆ Z definiowaniem, czy harmonogram projektu może ulec zmianie,
- ◆ Z zarządzaniem zmianami, w momencie gdy się pojawiają.

Obszar wiedzy:

Czas.

Techniki:

Przeglądy wydajności — technika ta zajmuje się analizą wydajności realizacji harmonogramu, gdy weźmie się pod uwagę datę rozpoczęcia i zakończenia, procent wykonania i pozostałą część prac do wykonania. Można tutaj również wykorzystać elementy techniki uzyskanej wartości,

Analiza wariantowa,

Oprogramowanie do zarządzania projektami,

Równoważenie zasobów (ang. *Resource Leveling*),

Analiza scenariuszy „co, jeśli”,

Zastosowanie przyspieszeń i opóźnień (ang. *Applying Leads and Lags*),

Kompresja harmonogramu,

Narzędzia do zarządzania harmonogramami (np. MS Project).

Wyjście:

Miary wydajności pracy (ang. *Work Performance Measurements*),

Zaktualizowane dane organizacyjne,

Żądania zmian (ang. *Change Requests*),

Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*),

Zaktualizowane dokumenty projektowe.

◆ **Nazwa polska: 7.3. Nadzór nad kosztami**

Nazwa angielska: 7.3. *Control Costs*

Wejście:

- Plan zarządzania projektem (ang. *Project Management Plan*),
- Wymagania finansowania projektu (ang. *Project Funding Requirements*),
- Informacje o wydajności pracy (ang. *Work Performance Information*),
- Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Proces ten jest odpowiedzialny za monitorowanie bieżących wydatków projektowych w stosunku do planów. Kluczową kwestią jest powiązanie stopnia konsumpcji funduszy z fizycznie wykonaną pracą oraz weryfikacja wydajności wydatkowania tych funduszy. Dodatkowo proces ten weryfikuje sposób wprowadzania zmian do bazowego planu kosztów.

Obszar wiedzy:

- Koszt.

Techniki:

- Zarządzanie wartością wypracowaną (ang. *Earned Value Management — EVM*),
- Prognozowanie,
- Indeks wydajności niezbędnej do zakończenia projektu (ang. *To-Complete Performance Index — TCPI*), Przeglądy wydajności,
- Analiza wariantowa,
- Oprogramowanie do zarządzania projektami.

Wyjście:

- Miary wydajności pracy (ang. *Work Performance Measurements*),
- Prognozy budżetowe (ang. *Budget Forecast*),
- Zaktualizowane dane organizacyjne,
- Żądania zmian (ang. *Change Requests*),
- Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*),
- Zaktualizowane dokumenty projektowe.

♦ Nazwa polska: 8.3. Kontrola jakości

Nazwa angielska: 8.3. *Perform Quality Control*

Wejście:

- Plan zarządzania projektem (ang. *Project Management Plan*),
- Miary jakościowe (ang. *Quality Metrics*),
- Listy weryfikujące jakość (ang. *Quality Checklists*),
- Miary wydajności pracy (ang. *Work Performance Measurements*),
- Zaakceptowane żądania zmian (ang. *Approved Change Requests*),
- Produkty (ang. *Deliverables*),
- Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Kontrola jakości jest procesem odpowiedzialnym za monitorowanie i odnotowanie rezultatów projektu. Standardy jakościowe zawierają procesy projektowe oraz cele produktowe. Oznacza to, że proces ten zajmuje się kwestiami jakościowymi związanymi z procesem wytwórczym (np. wydajność kosztowa) oraz wytworzonymi produktami (np. wydajność produktu). Celem jest tutaj wykrycie przyczyn problemu oraz rekomendacja działań eliminujących te przyczyny.

Zadania jakościowe są często wykonywane przez dedykowany zespół jakościowy, który powinien posiadać wiedzę na temat statystycznych metod kontroli, takich jak próbkowanie.

Obszar wiedzy:

Jakość.

Techniki:

Diagramy przyczynowo-skutkowe,

Wykresy kontrolne,

Diagramy przepływów,

Histogramy,

Wykres Pareto,

Wykres próby (ang. *Run chart*) — pokazuje dane względem czasu; w przypadku zmian czy defektów umożliwia to dokładniejsze definiowanie trendów,

Diagram rozrzutu (ang. *Scatter diagram*) — umożliwia wskazanie związku pomiędzy dwoma zmiennymi (X,Y),

Próby statystyczne — wybór i badanie wybranych zbiorów danych (populacji),

Inspekcja — badanie rezultatów pracy przez pryzmat ich zgodności ze standardami,

Przegląd zaakceptowanych żądań zmian (ang. *Approved Change Requests Review*).

Wyjście:

Miary kontroli jakości (ang. *Quality Control Measurements*),

Sprawdzone zmiany (ang. *Validated Changes*),

Sprawdzone produkty (ang. *Validated Deliverables*),

Zaktualizowane dane organizacyjne,

Żądania zmian (ang. *Change Requests*),

Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*),

Zaktualizowane dokumenty projektowe.

Nazwa polska: 10.5. *Raportowanie wydajności*

Nazwa angielska: 10.5. *Report Performance*

Wejście:

- Plan zarządzania projektem (ang. *Project Management Plan*),
- Informacje o wydajności pracy (ang. *Work Performance Information*),
- Prognozy budżetowe (ang. *Budget Forecasts*),
- Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

Proces ten realizuje zebranie i dystrybucję danych o wydajności projektu, takich jak raporty stanu projektu, miary postępu i prognozy. Na bazie zebranych informacji uwzględniających zakres, harmonogram, koszt i jakość, proces ten generuje raporty, które są przekazywane interesariuszom. Przykładowe typy takich raportów to:

- ♦ Analiza dotychczasowej wydajności,
- ♦ Obecny stan ryzyk i zgłoszeń,
- ♦ Opis prac wykonanych w danym okresie czasu,
- ♦ Opis prac do wykonania w kolejnym okresie czasu,
- ♦ Podsumowanie zmian zaakceptowanych w danym okresie czasu.

Obszar wiedzy:

Komunikacja.

Techniki:

- Analiza wariantowa,
- Metody prognostyczne — przykładowo użycie danych historycznych do prognozowania trendu zmian w przyszłości lub ocena intuicyjna,
- Metody komunikacyjne,
- Systemy raportowe.

Wyjście:

- Raporty wydajnościowe (ang. *Performance Reports*),
- Zaktualizowane dane organizacyjne,
- Żądania zmian (ang. *Change Requests*).

♦ **Nazwa polska: 11.6. Monitorowanie ryzyk i nadzór nad nimi**

Nazwa angielska: 11.6. *Monitor and Control Risks*

Wejście:

- Rejestr ryzyk (ang. *Risk Register*),
- Plan zarządzania projektem (ang. *Project Management Plan*),
- Informacje o wydajności pracy (ang. *Work Performance Information*),
- Raporty wydajnościowe (ang. *Performance Reports*).

W trakcie tego procesu monitorowane są zarejestrowane ryzyka i identyfikowane nowe, wykonywany jest plan odpowiedzi na ryzyko i weryfikowana efektywność tych odpowiedzi. Ponadto proces ten monitoruje:

- ◆ Czy założenia projektowe są wciąż aktualne?
- ◆ Czy ryzyko zmieniło się od ostatniego przeglądu?
- ◆ Czy przestrzegane są właściwe polityki zarządzania ryzykiem?
- ◆ Czy rezerwy projektowe są adekwatne do ryzyk projektowych?

Obszar wiedzy:

Ryzyko.

Techniki:

Ponowny przegląd ryzyk (ang. *Risk Reassessment*) — w trakcie realizacji projektu należy zaplanować regularny przegląd rejestru ryzyk. W trakcie realizacji projektu ryzyka ulegają zmianie i należy identyfikować nowe ryzyka oraz aktualizować istniejące.

Audyty ryzyk — technika ta oznacza weryfikację efektywności odpowiedzi na ryzyko; audyty powinny być wykonywane regularnie w ramach spotkań przeglądowych lub jako osobne spotkania,

Analiza trendów i odchyleń — weryfikacja rzeczywistych rezultatów w stosunku do planowanych,

Techniczne miary wydajności,

Analiza rezerw,

Spotkania okresowe na temat stanu projektu.

Wyjście:

Zaktualizowany rejestr ryzyk (ang. *Risk Register Updates*),

Zaktualizowane dane organizacyjne,

Żądania zmian (ang. *Change Requests*),

Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*),

Zaktualizowane dokumenty projektowe.

◆ **Nazwa polska: 12.3. Administrowanie dostawami**

Nazwa angielska: 12.3. *Administer Procurements*

Wejście:

Dokumentacja na temat dostaw,

Plan zarządzania projektem (ang. *Project Management Plan*),

Kontrakty,

Raporty wydajnościowe (ang. *Performance Reports*),

Zaakceptowane żądania zmian (ang. *Approved Change Requests*),

Informacje o wydajności pracy (ang. *Work Performance Information*).

Proces ten zarządza kontraktami z dostawcami i relacjami pomiędzy kupującym a sprzedawcą. Oznacza to monitorowanie wydajności realizacji kontraktów i ewentualne zmiany oraz działania korekcyjne. Możliwa jest tutaj analiza, jak dany sprzedawca realizował swoje zamówienia w przeszłości.

Obszar wiedzy:

Zaopatrzenie.

Techniki:

System kontroli zmian kontraktów,

Przegląd wydajności dostawy — kupiec dokonuje formalnego przeglądu postępu prac u sprzedawcy z punktu widzenia zakresu, jakości, kosztów i harmonogramu w porównaniu do kontraktu,

Inspekcje i audyty — jeśli specyfikuje to kontrakt, kupiec może uruchomić audyt prac wykonywanych przez sprzedawcę,

Raporty wydajnościowe — raporty te są podstawą do weryfikacji zdolności wywiązania się z kontraktu,

Systemy płatności,

Administracja zażaleniami — rozwiązywanie sporów pomiędzy sprzedającym a kupującym, które dotyczą zmian i sposobu ich finansowania.

System zarządzania dokumentami na temat dostaw (ang. *Records Management System*).

Wyjście:

Dokumentacja na temat dostaw,

Zaktualizowane dane organizacyjne,

Żądania zmian (ang. *Change Requests*),

Zaktualizowany plan zarządzania projektem (ang. *Project Management Plan Updates*).

Procesy zamknięcia to procesy niezbędne do formalnego zamknięcia projektu lub fazy bez względu na to, czy projekt zmierza do negatywnego, czy pozytywnego zakończenia.

♦ Nazwa polska: 4.6. *Zamknięcie projektu lub fazy*

Nazwa angielska: 4.6. *Close Project or Phase*

Wejście:

Plan zarządzania projektem (ang. *Project Management Plan*),

Zaakceptowane produkty (ang. *Accepted Deliverables*),

Inna dokumentacja i narzędzia, które mogą mieć wpływ na proces.

W trakcie tego procesu formalnie zamykane są wszelkie pozostałe procesy związane z projektem. Wykonywane są m.in. przeglądy sprawdzające, czy faktycznie wykonano prace projektowe oraz osiągnięto założone cele.

Zamknięcie projektu, oprócz działań operacyjnych, wiąże się również z szeregiem działań administracyjnych, takich jak przekazanie wykonanych produktów lub usług do działów.

Obszar wiedzy:

Integracja.

Techniki:

Osąd eksperta.

Wyjście:

Finalny produkt, usługa lub rezultat przejścia (np. migracji (ang. *Final Product, Service, or Result Transition*),

Zaktualizowane dane organizacyjne.

♦ **Nazwa polska: 12.4. Zamknięcie dostaw**

Nazwa angielska: 12.4. *Close Procurements*

Wejście:

Plan zarządzania projektem (ang. *Project Management Plan*),

Dokumentacja dostaw.

W trakcie tego procesu kompletowana jest i finalizowana każda dostawa zrealizowana w ramach projektu lub jego fazy. Częścią procesu jest weryfikacja, czy wszystkie prace i dostarczone produkty zostały zaakceptowane.

Proces ten zawiera również działania administracyjne, które mają na celu finalizowanie otwartych zażeń oraz aktualizację i archiwizację danych związanych z dostawami.

Proces przewiduje sytuację awaryjną, w której kontrakt jest rozwiązywany przed czasem realizacji. Może to nastąpić w wyniku porozumienia obu stron, ze względu na zaniedbania jednej ze stron lub dla wygody kupującego (jeżeli umożliwia to kontrakt).

Obszar wiedzy:

Zaopatrzenie.

Techniki:

Audyty zaopatrzeniowe — przegląd wszystkich procesów zaopatrzeniowych w celu identyfikacji, czy kontrakty zrealizowane w ramach danego projektu są porażką, czy sukcesem,

Negocjacje,

System zarządzania dokumentami na temat dostaw (ang. *Records Management System*).

Wyjście:

Zamknięte kontrakty (ang. *Closed Contracts*),

Zaktualizowane dane organizacyjne.

Dodatek B

Specyfikacja dyscyplin RUP z Rational Method Composer (RMC)

♦ Dyscyplina RUP po polsku: *Modelowanie biznesowe*

Dyscyplina RUP po angielsku: *Business Modelling*

Cele:

- ♦ Zrozumienie bieżących problemów organizacji odbiorcy i zidentyfikowanie potencjalnych obszarów ulepszeń,
- ♦ Ocena wpływu zmian na organizację odbiorcy,
- ♦ Zapewnienie wspólnego języka i zrozumienia różnych grup interesariuszy (użytkownicy, dostawcy, odbiorcy itp.),
- ♦ Ustalenie wymagań biznesowych dla przygotowywanego oprogramowania,
- ♦ Ustalenie poziomu zgodności przygotowanego oprogramowania z wymaganiami.

Wybrane produkty (wyniki):

- ♦ Wizja biznesowa (ang. *Business Vision*),
- ♦ Architektura biznesowa (ang. *Business Architecture*),
- ♦ Dodatkowa specyfikacja biznesowa (ang. *Supplementary Business Specification*),
- ♦ Model biznesowy (ang. *Business Rules / Business Analysis Model*),
- ♦ Słownik biznesowy (ang. *Business Glossary*).

♦ Dyscyplina RUP po polsku: *Wymagania*

Dyscyplina RUP po angielsku: *Requirements*

Cele:

- ♦ Uzgodnienie zakresu funkcjonalności tworzonego systemu pomiędzy odbiorcą i dostawcą,
- ♦ Zapewnienie zrozumienia wymagań wśród projektantów i programistów,
- ♦ Ustalenie ograniczeń systemu,
- ♦ Przygotowanie podstaw do planowania iteracji,
- ♦ Przygotowanie danych niezbędnych do oceny czasu i kosztu wytworzenia systemu,
- ♦ Zdefiniowanie interfejsu użytkownika w oparciu o potrzeby użytkowników.

Wybrane produkty (wyniki):

- ♦ Model przypadków użycia (ang. *Use Case Model*),
- ♦ Baza atrybutów wymagań (ang. *Requirements Attributes*),
- ♦ Plan obsługi wymagań (ang. *Requirements Management Plan*),
- ♦ Specyfikacja wymagań dla oprogramowania (ang. *Software Requirements Specification*),
- ♦ Specyfikacja wymagań dodatkowych (ang. *Supplementary Specification*),
- ♦ Wizja systemu (ang. *Vision*).

♦ Dyscyplina RUP po polsku: *Analiza i projektowanie*

Dyscyplina RUP po angielsku: *Analysis and Design*

Cele:

- ♦ Transformacja wymagań na projekt systemu,
- ♦ Przygotowanie ogólnej architektury systemu,
- ♦ Uwzględnienie środowiska docelowego i wymagań pozafunkcjonalnych (np. wydajność).

Wybrane produkty (wyniki):

- ♦ Architektura komponentowa (ang. *Component Architecture*),
- ♦ Architektura oprogramowania (ang. *Software Architecture*),
- ♦ Architektura systemowa (ang. *System Architecture*).

♦ Dyscyplina RUP po polsku: *Implementacja*

Dyscyplina RUP po angielsku: *Implementation*

Cele:

- ♦ Zdefiniowana architektura techniczna, z uwzględnieniem podziału na warstwy,
- ♦ Implementacja projektu (programowanie),
- ♦ Stworzenie i przetestowanie zaprojektowanych komponentów,
- ♦ Integracja komponentów w finalne rozwiązanie.

Wybrane produkty (wyniki):

- ♦ Model implementacyjny (ang. *Implementation Model*),
- ♦ Plan integracji (ang. *Integration Build Plan*).

♦ Dyscyplina RUP po polsku: *Test*

Dyscyplina RUP po angielsku: *Test*

Cele:

- ♦ Znalezienie i udokumentowanie defektów jakościowych oprogramowania,
- ♦ Ocena i udokumentowanie zgodności systemu z wymaganiami.

Wybrane produkty (wyniki):

- ♦ Plan testów (ang. *Test Plan*),
- ♦ Przypadki do przetestowania (ang. *Test Cases*),
- ♦ Dziennik testów (ang. *Test Log*),
- ♦ Model analizowania obciążenia (ang. *Workload Analysis Model*).

♦ Dyscyplina RUP po polsku: *Wdrożenie*

Dyscyplina RUP po angielsku: *Deployment*

Cele:

- ♦ Zapewnienie, że stworzone rozwiązanie jest dostępne dla użytkowników, zgodnie z projektem.

Wybrane produkty (wyniki):

- ♦ Plan instalacji (ang. *Deployment Plan*),
- ♦ Jednostki instalacji (ang. *Deployment Units*),
- ♦ Informacje o wersji (ang. *Release Notes*),
- ♦ Materiały szkoleniowe (ang. *Training Materials*).

◆ **Dyscyplina RUP po polsku: Zarządzanie zmianą i konfiguracją**

Dyscyplina RUP po angielsku: *Change and configuration management*

Cele:

- ◆ Kontrola i synchronizacja rozwoju komponentów produkowanego systemu.

Wybrane produkty (wyniki):

- ◆ Żądanie zmiany (ang. *Change Request*),
- ◆ Plan zarządzania konfiguracją (ang. *Configuration Management Plan*).

◆ **Dyscyplina RUP po polsku: Zarządzanie projektem**

Dyscyplina RUP po angielsku: *Project Management*

Cele:

- ◆ Planowanie zasobów, kosztów i harmonogramów,
- ◆ Zarządzanie projektem wytwórczym,
- ◆ Zarządzanie ryzykiem.

Wybrane produkty (wyniki):

- ◆ Wskaźniki projektu (ang. *Project Measures*),
- ◆ Plan wytwórczy (ang. *Software Development Plan*),
- ◆ Plan iteracji (ang. *Iteration Plan*).

◆ **Dyscyplina RUP po polsku: Zarządzanie środowiskiem**

Dyscyplina RUP po angielsku: *Environment*

Cele:

- ◆ Zapewnienie środowiska pracy dla zespołu projektowego.

Wybrane produkty (wyniki):

- ◆ Infrastruktura wytwórcza (ang. *Development Infrastructure*),
- ◆ Proces wytwórczy (ang. *Development Process*),
- ◆ Wskazówki specyficzne dla projektu (ang. *Project-Specific Guidelins*).

Dodatek C

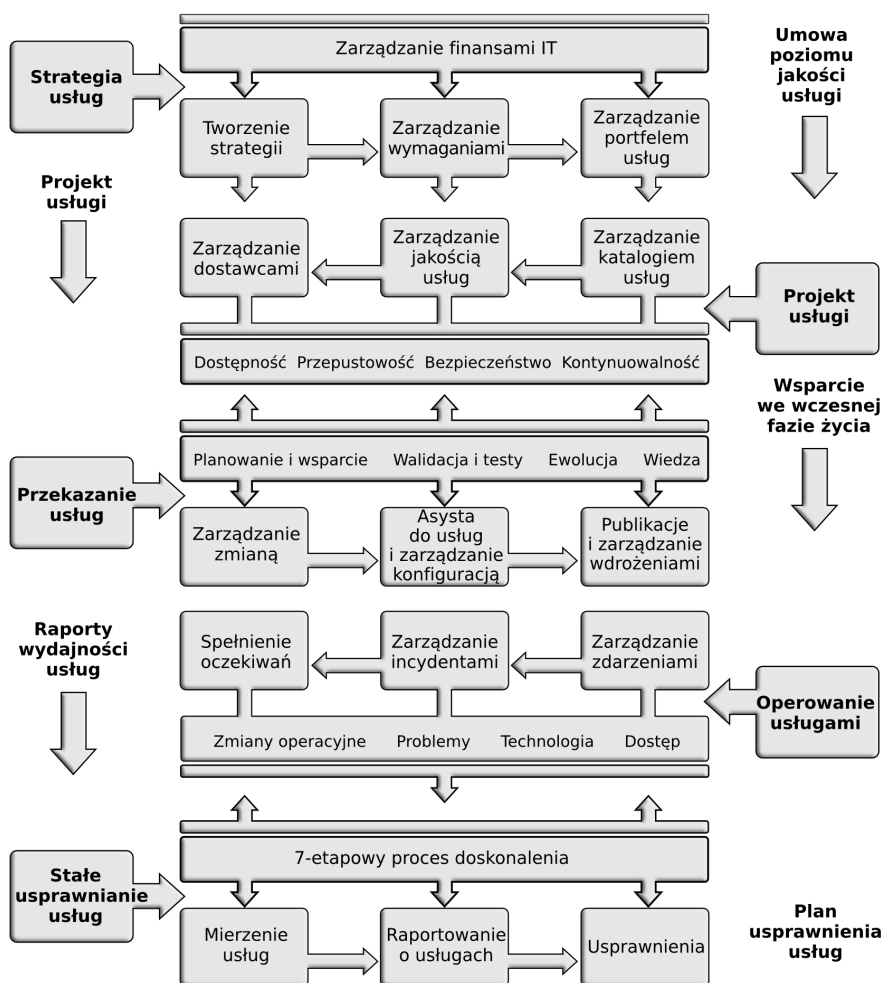
Lista wszystkich procesów ITIL

Strategia usług

- ◆ Tworzenie strategii (ang. *Strategy Generation*) obejmuje analizę takich elementów jak konkurencyjność, wartość usługi, sposób dostarczenia usługi, kluczowe czynniki sukcesu, organizacja projektowania i wytwarzania usługi, wybór sposobu zagwarantowania zasobów usłudze.
- ◆ Zarządzanie finansami (ang. *Financial Management*) definiuje sposób zarządzania budżetami, księgowaniem i rozliczaniem wymagań. Korzystając z narzędzi finansowych, usługa ta agreguje, współdzieli i utrzymuje dane finansowe konieczne do podjęcia krytycznych decyzji i działań.
- ◆ Zarządzanie wymaganiami (ang. *Demand Management*) precyzuje wpływ klienta na wymagania w stosunku do usług i zapotrzebowanie na przepustowość. Na poziomie strategicznym może to oznaczać analizę czynności biznesowych i profili użytkowników, a na poziomie taktycznym — właściwe balansowanie zasobami.
- ◆ Zarządzanie portfolio usług (ang. *Service Portfolio Management*) opisuje proaktywne mechanizmy zarządzania inwestycjami w ciągu całego cyklu życia usługi. Zawiera zadania związane z definiowaniem portfolio usług, jego analizą, akceptacją i sposobem podejmowania kluczowych decyzji.

Projektowanie usług

- ◆ Zarządzanie katalogiem usług (ang. *Service Catalogue Management*). Katalog usług jest jedynym, centralnym źródłem informacji o usługach IT dostarczanych biznesowi wraz z informacjami powiązanymi, takimi jak parametry dostępności i status usługi. Katalog usług pobiera te informacje głównie z portfolio usług za pomocą procesu zarządzania relacjami biznesowymi lub procesu zarządzania poziomem usług.



Rysunek 78. Powiązania procesów ITIL

- ♦ Zarządzanie poziomem usług (ang. *Service Level Management*) zajmuje się tematyką negocjacji i dokumentacji celów biznesowych usług IT. Następnie monitoruje i raportuje stan faktyczny względem uzgodnionego. Proces ten ma na celu zapewnienie operacyjności usług poprzez mierzenie ich wydajności. Głównie informacje te są zawarte w porozumieniach o poziomie świadczonych usług (ang. *Service Level Agreements* lub *Operational Level Agreements*) oraz planach udoskończeń usług i planie jakości usług.
- ♦ Zarządzanie przepustowością (ang. *Capacity Management*) zapewnia dostarczenie koniecznej przepustowości w trakcie całego cyklu życia usługi. Proces ten uwzględnia wymagania biznesowe i realne możliwości działu IT. Fundamentem procesu jest system zarządzania informacją o przepustowości, który zapewnia składowanie, analizę i zbiorcze raportowanie wszystkich podprocesów.

- ♦ Zarządzanie dostępnością (ang. *Availability Management*) obejmuje kwestie związane z dostępnością usług, komponentów i zasobów. Zawiera definicję wszystkich miar koniecznych do zapewnienia realizacji planów zapotrzebowań. Proces zawiera dwa niezależne mechanizmy:
 - ♦ reaktywny, czyli monitorowanie, mierzenie, analizę i zarządzanie zdarzeniami, incydentami oraz problemami związanymi z niedostępnością usług,
 - ♦ proaktywny, czyli planowanie, projektowanie, rekomendacje i zwiększanie dostępności.
- ♦ Zarządzanie ciągłością usług IT (ang. *IT Service Continuity Management*) definiuje mechanizmy bieżącego utrzymania, redukcji ryzyka i planów odtworzenia, w stosunku do usług kluczowych dla biznesu. W ramach tego procesu na bazie planów biznesowych tworzone są polityki i plany, które mają zapewnić stałą analizę stanu bieżącego oraz aktywne zarządzanie ryzykiem.
- ♦ Zarządzanie bezpieczeństwem informacji (ang. *Information Security Management*). Celem tego procesu jest zapewnienie, że właściwa informacja jest udostępniona właściwym osobom, zgodnie z politykami i strategiami bezpieczeństwa, czyli:
 - ♦ jest dostępna i używalna, gdy to konieczne (dostępność),
 - ♦ jest dostępna lub jawna tylko dla tych osób, które mają do tego uprawnienia (poufność),
 - ♦ jest kompletna, dokładna i zabezpieczona przed nieautoryzowanymi modyfikacjami (integralność),
 - ♦ transakcje biznesowe, takie jak wymiana informacji, są zaufane (autentyczność i niezaprzeczalność).
- ♦ Zarządzanie dostawcami (ang. *Supplier Management*) zapewnia uzyskanie wartości za płatności przekazane dostawcom. Weryfikuje, czy dostawcy wypełnią cele zawarte w kontraktach i porozumieniach, zgodnie z przyjętymi założeniami.

Przekazanie usług

- ♦ Zarządzanie zmianą (ang. *Change Management*) organizuje prace związaną ze zmianami w usłudze, włącznie z zapisem, analizą, autoryzacją, priorytetyzowaniem, planowaniem, testowaniem, implementacją, dokumentacją i przeglądem. Takie podejście ma zapewnić kontrolowany sposób wprowadzania zmiany i optymalizację z biznesowego punktu widzenia.
- ♦ Zarządzanie kapitałem i konfiguracją usług (ang. *Service Asset and Configuration Management*) zapewnia identyfikację, kontrolę i księgowanie kapitału usługi. Definiuje również kwestie związane z ochroną konfiguracji i zapewnieniem integralności usługi w trakcie całego cyklu życia.
- ♦ Zarządzanie wiedzą (ang. *Knowledge Management*) to zapewnienie, że właściwa osoba będzie miała dostęp do właściwej wiedzy, we właściwym czasie. Celem

jest tutaj dostarczenie i wsparcie usług wymaganych przez biznes, a kluczem jest baza wiedzy, składająca kluczowe, ustandaryzowane informacje opisujące kapitał i konfigurację usług.

- ◆ Planowanie i wsparcie przekazania (ang. *Transition Planning and Support*) obejmuje swoim zakresem wszelkie przygotowania niezbędne do wdrożenia zaprojektowanych usług, czyli takie działania jak planowanie niezbędnych zasobów, ich koordynację i zarządzanie ryzykiem.
- ◆ Zarządzanie publikacjami i wdrożeniami (ang. *Release and Deployment Management*) koncentruje się na zebraniu wszelkich kluczowych informacji o usłudze produkcyjnej i doprowadzeniu do efektywnego używania nowej lub zmienionej usługi.
- ◆ Walidacja i testowanie usług (ang. *Service Validation and Testing*). Sukces procesu testowego jest uzależniony od tego, czy sprawdzono szczegółowo, jak usługa będzie używana i jak jest skonstruowana. Wszystkie usługi, bez względu na sposób wytworzenia (wewnętrzny lub zewnętrzny), muszą być odpowiednio przetestowane. Proces ten sprawdza stopień realizacji założeń biznesowych w postaci raportów testowych.
- ◆ Ewaluacja (ang. *Evaluation*). Kluczowym czynnikiem sukcesu przekazania jest zapewnienie użyteczności biznesowej usługi w chwili wdrożenia oraz podczas wykorzystania w środowisku produkcyjnym. Właśnie dlatego proces ten udostępnia stosowne metryki i miary techniczne, które służą do stałego monitorowania, czy wykonane prace projektowe, wdrożeniowe i utrzymaniowe przynoszą spodziewane efekty.

Eksplotacja usług

- ◆ Zarządzanie zdarzeniami (ang. *Event Management*). Zdarzenie to zmiana stanu usługi, która może prowadzić do incydentu, problemu lub zmiany, jeżeli coś nie funkcjonuje właściwie. Zdarzenie może jednak wskazywać na rutynową czynność, taką jak konieczność zmiany taśmy. Odpowiedzią na zdarzenie może być czynność automatyczna lub ręczna interwencja zawiadomionego (np. SMS-em) personelu. Zarządzanie zdarzeniami jest ściśle powiązane z systemami monitorującymi, które sprawdzają statusy komponentów nawet wtedy, gdy nie wystąpiły zdarzenia.
- ◆ Zarządzanie incydentami (ang. *Incident Management*). Incydent to niezaplanowane przerwanie usługi IT lub redukcja jej jakości. Incydemtem jest również błąd konfiguracji, który jeszcze nie wpłynął na usługę. Zarządzanie incydentami ma na celu przywrócenie usługi do normalnego stanu tak szybko, jak to możliwe, przy minimalizacji ewentualnych skutków ubocznych mających wpływ na operacje biznesowe. Incydenty są najczęściej rejestrowane przez proces zarządzania zdarzeniami lub użytkowników kontaktujących się z „*service desk*”. Incydenty są kategoryzowane, aby łatwiej było zidentyfikować, kto powinien nad daną kwestią pracować. Incydenty są również priorytetyzowane, zgodnie z pilnością i wpływem na operacje biznesowe. Jeżeli w trakcie działań operacyjnych występuje problem związany z szybkością realizacji, decyzja o zwiększeniu priorytetu może zostać przekazana do osób postawionych wyżej

w hierarchii organizacyjnej (eskalacja). Po analizie, zdiagnozowaniu, implementacji i przetestowaniu rozwiązania należy upewnić się, że użytkownik jest usatysfakcjonowany; dopiero wtedy incydent może zostać zamknięty.

- ♦ Wypełnienie żądań (ang. *Request Fulfillment*). Żądanie użytkownika może dotyczyć informacji, rady, zmiany lub dostępu do usługi IT. Celem tego procesu jest utworzenie dla użytkowników końcowych dedykowanego kanału komunikacyjnego i zbioru procedur, które w standardowy sposób zapewnią obsługę żądania z listą wymaganych akceptacji.
- ♦ Zarządzanie dostępem (ang. *Access Management*). Celem tego procesu jest nadanie uprawnień użytkownikom, tak aby mieli dostęp do usługi lub grupy usług, przy równoczesnym zablokowaniu dostępu dla osób nieuprawnionych. Dlatego też uwzględnione są kwestie jednoznacznej identyfikacji tożsamości, weryfikacja uprawnień, zarządzanie uprawnieniami i logowanie wykonanych operacji. Proces ten wspiera również poufność, dostępność i integralność danych oraz zarządzanie własnością intelektualną.
- ♦ Zarządzanie problemami (ang. *Problem Management*). Problem jest inicjowany przez jeden lub więcej incydentów. Przyczyna problemu zazwyczaj nie jest znana w momencie rejestracji i proces zarządzania problemami jest odpowiedzialny właśnie za analizę zgłoszonego problemu. Kluczowym celem procesu jest zapobieganie problemom i rezultatom incydentów, eliminacja powtarzających się incydentów oraz minimalizacja wpływu incydentów na operacje biznesowe. Proces zawiera diagnozowanie przyczyn incydentów, opracowanie rozwiązania, jego implementację oraz utrzymanie informacji o problemach i ich obejściach lub rozwiązaniach.

Ustawiczne doskonalenie usług

- ♦ Siedmioetapowy proces doskonalenia (ang. *7-Step Improvement Process*):
 1. Zdefiniowanie, co powinno być mierzone,
 2. Zdefiniowanie, co można zmierzyć,
 3. Zebranie danych typu kto, jak, kiedy i określenie, czy dane są spójne,
 4. Obróbka danych (jak często, w jakim formacie, jakim systemem, jak dokładnie),
 5. Analiza danych (jakie są wzajemne relacje pomiędzy danymi i ich trendy, czy dane rzeczywiste są zgodne z planem i spełniają założone cele, czy konieczne są czynności naprawcze i jakie),
 6. Prezentacja informacji, podsumowania, plany działań itp.,
 7. Implementacja czynności naprawczych.
- ♦ Mierzenie usług (ang. *Service Measurement*) — cztery główne przyczyny monitorowania i mierzenia to:
 - ♦ walidacja poprzednio podjętych decyzji,
 - ♦ działania bezpośrednie nakierowane na uzyskanie uzgodnionych celów,

- ◆ uzasadnienie koniecznych działań poprzez udowodnienie stanu faktycznego,
- ◆ interwencja we właściwym miejscu i czasie — działania naprawcze.
- ◆ Wiele organizacji posiada miary na poziomie komponentów, ale dopiero miary na poziomie usług umożliwiają spojrzenie na całość z punktu widzenia klienta. Istnieją trzy kluczowe grupy metryk:
 - ◆ technologiczne — związane z komponentami i aplikacjami, takie jak wydajność i dostępność,
 - ◆ procesowe — zawarte w krytycznych parametrach sukcesu (ang. *Critical Success Factors* — CSF) lub kluczowych identyfikatorach wydajności (ang. *Key Performance Indicator* — KPI),
 - ◆ serwisowe — będące rezultatem usługi z punktu widzenia odbiorcy.

Raportowanie usług (ang. *Service Reporting*) — duża liczba danych jest konfrontowana i monitorowana codziennie przez IT w ramach procedur jakościowych. Z punktu widzenia biznesowego jedynie mały podzbiór tych danych jest istotny, np. historyczna reprezentacja wydajności lub stopień realizacji porozumień o poziomie świadczonych usług (*SLA*). IT buduje więc na potrzeby biznesowe zbiór raportów typu „co się zdarzyło”, „co zrobiło IT”, „jak IT zapewnia prewencję” lub „jak IT pracuje, aby ulepszyć sposób dostarczania usług”. Cel raportowania umiejscowiony jest w przyszłości, ale samo raportowanie koncentruje się na przeszłości i udostępnia rzeczywiste informacje o tym, jak IT wspiera biznes firmy pozytywnie lub negatywnie ☺.

Dodatek D

Lista wszystkich procesów COBIT

Poniższa lista procesów bazuje na publikacji COBIT 4.0.

Planowanie i organizacja

- ◆ PO1. Wyznaczanie planu strategicznego IT — określa kompromis między możliwościami IT a wymaganiami biznesowymi.
- ◆ PO2. Wyznaczanie architektury informatycznej — długoterminowy plan optymalizacji organizacji systemów informatycznych.
- ◆ PO3. Wyznaczanie kierunku technologicznego — wybór kluczowych technologii i rozwiązań, które będą służyły realizacji długoterminowych celów.
- ◆ PO4. Określanie procesów, organizacji i powiązań IT — reguluje sposób komunikacji IT z innymi jednostkami.
- ◆ PO5. Zarządzanie inwestycjami IT — reguluje sposób finansowania i kontroli kosztów.
- ◆ PO6. Komunikowanie celów i kierunków wyznaczonych przez kierownictwo — definiuje sposób wprowadzenia w życie polityk informacyjnych, włącznie z kwestiami jakości i bezpieczeństwa.
- ◆ PO7. Zarządzanie zasobami ludzkimi IT — to m.in. mechanizmy motywacyjne, zdobywania kompetencji oraz maksymalizacja udziału pracowników w procesach IT.
- ◆ PO8. Zarządzanie jakością — to m.in. wprowadzenie procesów jakościowych, zdefiniowanie standardów, mechanizmy stałego doskonalenia.
- ◆ PO9. Szacowanie ryzyka — mechanizmy wykrywania ryzyk i zarządzania nimi.
- ◆ PO10. Zarządzanie projektami — dostarczanie rozwiązań w uzgodnionym czasie i budżecie.

Nabycie i implementacja

- ◆ AI1. Identyfikowanie rozwiązań zautomatyzowanych — to m.in. zdobycie wiedzy na temat istniejących rozwiązań, rozpatrzenie opcji outsourcingu, zdefiniowanie kluczowych użytkowników.
- ◆ AI2. Nabywanie i utrzymanie oprogramowania aplikacyjnego — to m.in. kwestie konfiguracyjne, rozdział pomiędzy jednostką implementacyjną i testową, priorytetyzacja wymagań, uzgodnienie zakresu dokumentacji, monitorowanie zgodności z architekturą IT.
- ◆ AI3. Nabywanie i utrzymanie infrastruktury technologicznej — dostarczenie właściwych platform technologicznych dla aplikacji biznesowych.
- ◆ AI4. Zapewnienie dostępności i używalności — zapewnienie dostępu do aplikacji biznesowych i rozwiązań technologicznych.
- ◆ AI5. Pozyskiwanie zasobów IT — zapewnienie stosownych zasobów ludzkich, oprogramowania sprzętu do realizacji założonych celów (projektów).
- ◆ AI6. Zarządzanie zmianami — zapewnienie regulowanego kompromisu pomiędzy koniecznością zmian a warunkiem utrzymania realizacji projektu o zdefiniowanej jakości w czasie i budżecie.
- ◆ AI7. Instalacja i akredytowanie systemów — instalacja i weryfikacja, czy istniejące rozwiązania zapewniają realizację założonych celów.

Dostarczenie i wsparcie

- ◆ DS1. Wyznaczanie poziomów usług i zarządzanie nimi — zakontraktowanie umów typu SLA (ang. *Service Level Agreement*) definiujących warunki i sposób świadczenia usług (np. czas reakcji na różne typy zdarzeń).
- ◆ DS2. Zarządzanie usługami zewnętrznymi — jasna definicja odpowiedzialności strony trzeciej w przypadku korzystania z usług firmy zewnętrznej (np. *outsourcing*).
- ◆ DS3. Zarządzanie wydajnością i przepustowością — zapewnienie dostępności zasobów IT oraz tego, że są one najlepiej i najbardziej optymalnie wykorzystywane, tak aby zapewnić ich maksymalną wydajność.
- ◆ DS4. Zapewnienie ciągłości usług — zapewnienie uzgodnionej dostępności usług oraz prewencyjna minimalizacja potencjalnych skutków awarii i innych zakłóceń.
- ◆ DS5. Zapewnienie bezpieczeństwa systemów — ochrona informacji przed nieautoryzowanym użyciem, ujawnieniem, modyfikacją, uszkodzeniem lub utratą.
- ◆ DS6. Rozpoznanie i przypisanie kosztów — zapewnienie właściwej świadomości kosztów usług IT.
- ◆ DS7. Kształcenie i szkolenie użytkowników — stałe podwyższanie wiedzy o technologiach oraz ryzykach i odpowiedzialności z nimi związanej.

- ♦ DS8. Zarządzanie wsparciem i incydentami — zapewnienie, że każdy zgłoszony przez użytkowników problem zostanie odpowiednio obsłużony.
- ♦ DS9. Zarządzanie konfiguracją — to m.in. ewidencja i kontrola wszystkich komponentów IT, kontrola konfiguracji i wersji.
- ♦ DS10. Zarządzanie problemami i incydentami — zapewnienie, że problemy i incydenty zostaną rozwiązane, a przyczyny ich wystąpienia — przeanalizowane. Proces może, ale nie musi, zostać uruchomiony przez DS8.
- ♦ DS11. Zarządzanie danymi — zapewnienie kompletności, dokładności i prawdziwości danych w trakcie ich wprowadzania, aktualizacji i przechowywania w systemach informatycznych.
- ♦ DS12. Zarządzanie środowiskiem fizycznym — dostarczenie odpowiedniego środowiska fizycznego, które zapewnia ochronę ludziom i infrastrukturze IT przed zagrożeniami naturalnymi i wywołanymi przez człowieka, takimi jak np. włamanie.
- ♦ DS13. Zarządzanie operacjami — zapewnienie, że ważne funkcje dotyczące wsparcia są wykonywane w regularny i zorganizowany sposób.

Monitoring i ewaluacja

- ♦ ME1. Monitorowanie procesów — zapewnienie, że cele wydajnościowe ustalone dla procesów IT zostaną osiągnięte.
- ♦ ME2. Oceny kontroli wewnętrznej — zapewnienie, że cele kontroli wewnętrznej ustalone dla procesów IT zostaną osiągnięte.
- ♦ ME3. Zapewnienie zgodności z wymaganiami zewnętrznymi — stałe zapewnianie, że planowane i działające rozwiązania IT są zgodne z obowiązującymi regulacjami prawnymi i korporacyjnymi oraz wymaganiami zewnętrznymi, takimi jak umowy.
- ♦ ME4. Zapewnienie ładu informatycznego (ang. *IT Governance*) — to techniki organizacyjne, które mają na celu zapewnienie, że inwestycje w IT są zgodne ze strategią i celami firmy.

Spis rysunków

Rysunek 1.	Architektura procesów Prince2	18
Rysunek 2.	Przykładowa formatka ryzyka w MS Enterprise Project Management	20
Rysunek 3.	Przykładowa struktura prac (ang. Work Breakdown Structure)	22
Rysunek 4.	Model przywództwa zespołowego Kena Blancharda	37
Rysunek 5.	Wynik ankiety „Jakich systemów EPM używasz?” Pierra-Jeana Cherreta	38
Rysunek 6.	Architektura grup procesów PMBoK	40
Rysunek 7.	Współbieżność grup procesów PMBoK	40
Rysunek 8.	Relacja pomiędzy zadaniami Koniec-do-Początku	43
Rysunek 9.	Relacja pomiędzy zadaniami Koniec-do-Końca	43
Rysunek 10.	Relacja pomiędzy zadaniami Początek-do-Początku	43
Rysunek 11.	Relacja pomiędzy zadaniami Początek-do-Końca	44
Rysunek 12.	Przyspieszenia i opóźnienia zadań w MS Project	45
Rysunek 13.	Mechanizm równoważenia zasobów	46
Rysunek 14.	Przykład powstania diagramu pokrewieństwa	49
Rysunek 15.	Przykład analizy pola siły	50
Rysunek 16.	Przykład diagramu relacji: Dlaczego nie wykorzystujemy procesów rozwiązywanie problemów?	50
Rysunek 17.	Przykład diagramu macierzowego	51
Rysunek 18.	Planu informatyzacji państwa na lata 2007 – 2010	56
Rysunek 19.	Przykład zastosowania Zarządzania Wartością Wypracowaną (ang. Earned Value Management — EVM)	62
Rysunek 20.	Przykład wykresu kontrolnego	64
Rysunek 21.	Przykład histogramu	64
Rysunek 22.	Przykład wykresu Pareto	65
Rysunek 23.	Model kaskadowy	70
Rysunek 24.	Model spiralny	70
Rysunek 25.	Model iteracyjny	71
Rysunek 26.	Model adaptacyjny	71
Rysunek 27.	Architektura RUP	75
Rysunek 28.	Przeływ pracy w metodyce RUP	76
Rysunek 29.	Przykładowy ekran narzędzia Rational Method Composer (RMC)	81
Rysunek 30.	Diagram Zaktualizowanej Wartości Netto (ang. Net Present Value — NPV)	91
Rysunek 31.	Architektura metody MSF	107
Rysunek 32.	Architektura Fazy Wizji w MSF	108
Rysunek 33.	Architektura Fazy Planowania w MSF	110

Rysunek 34. Architektura Fazy Konstrukcji w MSF	112
Rysunek 35. Plan codziennych kompilacji w MSF	113
Rysunek 36. Codziennie kompilacje a fazy MSF	115
Rysunek 37. Architektura Fazy Stabilizacji w MSF	116
Rysunek 38. Przykładowy diagram stanów zgłoszeń	117
Rysunek 39. Punkt schodzenia się liczby błędów (ang. Bug Convergence)	118
Rysunek 40. Trampolina braku błędów (ang. Zero Bug Bogunce)	118
Rysunek 41. Architektura Fazy Wdrożenia w MSF	120
Rysunek 42. Współdziałanie MSF i MOF	122
Rysunek 43. Architektura MOF	122
Rysunek 44. Amerykańska reklama zgodna z podejściem trade-off	123
Rysunek 45. Trójkąt negocjacyjny MSF	124
Rysunek 46. Tetraedr negocjacyjny	124
Rysunek 47. Zarządzanie ryzykiem w MSF	125
Rysunek 48. Zarządzanie gotowością w MSF	126
Rysunek 49. RUP czy MSF?	127
Rysunek 50. Struktura produktowa (WBS) w przykładzie wdrożenia narzędzia BI	133
Rysunek 51. Struktura zespołu wdrożeniowego w przykładowym wdrożeniu narzędzia klasy BI	144
Rysunek 52. Przykład techniki EVM (1) w projekcie wdrożenia rozwiązania typu BI	164
Rysunek 53. Przykład techniki EVM (2) w projekcie wdrożenia rozwiązania typu BI	170
Rysunek 54. Kent Beck	179
Rysunek 55. Badanie wydajności prac różnych wariantów współpracy par programistycznych	183
Rysunek 56. Scrum na boisku	189
Rysunek 57. Bajka Scrum o kurczaku i świnie	191
Rysunek 58. Architektura metodyki Scrum	192
Rysunek 59. Przykładowy wykres spalania	220
Rysunek 60. Ewolucja modelu CMMI	227
Rysunek 61. Przykładowe porównanie rzeczywistego poziomu z minimalnie wymaganym poziomem procesów CMMI	234
Rysunek 62. Mechanizm „odpowiednika poziomu”	235
Rysunek 63. Krzywa Gaussa	239
Rysunek 64. Przykładowy diagram SIPOC sprzedaży aut	243
Rysunek 65. Zasada działania „Domu jakości”	243
Rysunek 66. Przykładowy „Dom jakości”	244
Rysunek 67. Architektura ITIL	251
Rysunek 68. Cykl życia usługi ITIL	253
Rysunek 69. System certyfikacji ITIL w 3. wersji	256
Rysunek 70. Wysokopoziomowe cele kontrolne procesu COBIT „PO1. Wyznaczanie planu strategicznego IT”	267
Rysunek 71. Przykładowa tabela RACI dla procesu COBIT „PO1. Wyznaczanie planu strategicznego IT”	269
Rysunek 72. Miary metryki procesu COBIT PO1	269
Rysunek 73. Poziomy dojrzałości COBIT	270
Rysunek 74. Prince2 a RUP	276
Rysunek 75. Prince2 a Scrum	277
Rysunek 76. PMBoK a Scrum	278
Rysunek 77. PMBoK a RUP	278
Rysunek 78. Powiązania procesów ITIL	326

Spis tabel

Tabela 1.	Przykład matrycy priorytetyzacyjnej	51
Tabela 2.	Tabela SLA w projekcie BlogSerwis	89
Tabela 4.	Przykładowy kosztorys zleceńbiorecy w karcie projektu BlogSerwis na 14 stycznia 2008 roku	94
Tabela 5.	Przykładowa realizacja kosztorysu zleceńbiorecy w karcie projektu BlogSerwis na 26 czerwca 2008 roku	100
Tabela 7.	Tabela wyboru usług audytorskich w przykładowym wdrożeniu narzędzia klasy BI	140
Tabela 8.	Tabela stopnia realizacji procesów PMBoK (1) w przykładowym wdrożeniu narzędzia klasy BI	141
Tabela 9.	Tablica porównawcza oferentów w przykładzie wdrożenia rozwiązania klasy BI	146
Tabela 10.	Plan budżetu projektu wdrożenia rozwiązania klasy BI	152
Tabela 11.	Tabela stopnia realizacji procesów PMBoK (2) w przykładowym wdrożeniu narzędzia klasy BI	153
Tabela 12.	Przykładowy rejestr ryzyk w projekcie wdrożenia narzędzia typu BI	157
Tabela 13.	Tabela stopnia realizacji procesów PMBoK (3) w przykładowym wdrożeniu narzędzia klasy BI	158
Tabela 14.	Realizacja budżetu projektu wdrożenia rozwiązania klasy BI na lipiec	169
Tabela 15.	Tabela stopnia realizacji procesów PMBoK (4) w przykładowym wdrożeniu narzędzia klasy BI	171
Tabela 16.	Końcowa realizacja budżetu projektu wdrożenia rozwiązania klasy BI	175
Tabela 17.	Przykładowy dziennik zadań produktu w projekcie Scrum BlogSerwis	210
Tabela 18.	Dziennik zadań pierwszego sprinta w projekcie Scrum BlogSerwis	212
Tabela 19.	Przykładowa realizacja pierwszego sprinta w projekcie Scrum BlogSerwis	216
Tabela 20.	Kategorie procesów w CMMI	234
Tabela 21.	Porównanie reprezentacji ciągłej i stopniowanej CMMI	236
Tabela 22.	Prawdopodobieństwo sukcesu w przypadku wieloetapowych procesów	240
Tabela 23.	Przykładowy harmonogram wdrożenia narzędzia wspierającego procesy ITIL	255
Tabela 24.	Tabela danych wejściowych procesu COBIT „PO1. Wyznaczanie planu strategicznego IT”	268
Tabela 25.	Tabela danych wyjściowych procesu COBIT „PO1. Wyznaczanie planu strategicznego IT”	268
Tabela 26.	Wzajemna korelacja pomiędzy metodami	279

Źródła

Harold Kerzner, *Advanced project management*, Wydawnictwo Helion, Gliwice 2005.

Praca zbiorowa, *Frameworks for IT Management*, Wydawnictwo Van Haren Publishing, Zaltbommel 2006.

Scott Berkun, *The art of project management*, O'Reilly, Sebastopol 2005.

Joseph Philips, *Zarządzanie projektami IT*, Wydawnictwo Helion, Gliwice 2005.

Robert Cialdini, *Wywieranie wpływu na ludzi*, Gdańskie Wydawnictwo Psychologiczne, Gdańsk 2001.

Stephen R. Covey, *7 nawyków skutecznego działania*, Dom Wydawniczy Rebis, Poznań 2003.

Prince2

Praca zbiorowa (OGC), *Managing Successful Projects with Prince2*, Norwich 2005 „The Stationary Office”

PMBok

Praca zbiorowa (PMI), A GUIDE TO THE PROJECT MANAGEMENT BODY OF KNOWLEDGE (PMBOK GUIDE) Fourth Edition, Pennsylvania 2008, PMI

Rita Mulcaby, PMP Exam Prep, Rita's Course in a Book for Passing the PMP Exam Fifth Edition, For the PMBOK Guide – Third Edition, USA 2005, RMC Publications

http://www.12manage.com/%5Cmethods_pmi_pmbok.html

http://books.google.pl/books?id=RV3jQ16F1_cC&pg=PA205&lpg=PA205&dq=%22American+National+Standards+Institute%22+PMBoK&source=web&ots=Z2FG2eNcD_&sig=eXNzgct-ar9Vx4UzC-sP5CqPDVvk&hl=pl#PPR12,M1

<http://www.di.unipi.it/~macri/pmbok.pdf>

<http://adminian.com/2006/12/26/book-1-pmbok-part-2/>

<http://akson.sgh.waw.pl/~darry/PM/Texts/introduction.html>

<http://www.stsc.hill.af.mil/Crosstalk/2006/01/0601McKeever.html>

<http://www.karubik.de/en/gnupmps/knowledge-areas/integration/direct-manage-project-execution.html>

http://www.pmiswic.org/pdfs/PMP_Preso092106bwr.pdf

PgMP details http://www.pmi.org/PDF/PDC_PgMPHandbook.pdf

RUP

Rational Method Composer http://www.ploug.org.pl/konf_07/materialy/pdf/12_rup.pdf

PRINCE2 and RUP: Loose Coupling Works Best

http://www.ibm.com/developerworks/rational/library/content/RationalEdge/apr03/Prince2RUP_TheRationalEdge_Apr2003.pdf

Software Project Management – A Mapping between RUP and the PMBOK

<http://www.ibm.com/developerworks/rational/library/4721.html>

RUP in the dialogue with Scrum <http://www-128.ibm.com/developerworks/rational/library/feb05/krebs/>

<http://www.ibm.com/developerworks/rational/library/1826.html>

MSF

Praca zbiorowa (Microsoft), *MCAD/MCSD Self-Paced Training Kit: Microsoft (2nd Edition) .NET Core Requirements*, 2003, Microsoft Press

<http://technet.microsoft.com/en-us/library/bb497060.aspx>

MSF resources <http://www.microsoft.com/technet/solutionaccelerators/msf/default.aspx>

<http://blog.brodzinski.com/2006/07/msf-version-40-versus-version-31.html>

<http://geekswithblogs.net/cyoung/articles/63354.aspx>

Metodyki lekkie

Scrum Primer <http://www.rallydev.com/documents/scrumprimer.pdf>

<http://www.extremeprogramming.org/>

http://en.wikipedia.org/wiki/Extreme_Programming_Practices

<http://jeffsutherland.com/oops/schwapub.pdf>

http://www.softhouse.se/Uploades/Scrum_eng_webb.pdf

http://www.inmost.org.pl/articles/Programowanie_Parami_-_razem_czy_osobno

<http://www.javart.com.pl/userfiles/file/Scrum%20-%20GIGACON.pdf>

James W. Newkirk, Alexei A. Vorontsov, Test-Driven Development in Microsoft .NET, 2004, Microsoft Press

Six Sigma

http://pl.wikipedia.org/wiki/Sze%C5%9B%C4%87_sigma

<http://www.polishsixsigmaacademy.pl/>

<http://www.statsoft.pl/sixsigma.html>

<http://www.polishsixsigmaacademy.pl/files/FilozofiaSixSigma.doc>

<http://www.pcp.com.pl/files/czytelnia/sixsigma.pdf>

http://www.4pm.pl/artykul/firma_na_poziomie_6_czyli_mity_i_prawdy_o_metodzie_six_sigma-26-95.html

<http://www.qfdonline.com>

ITIL

<http://centrum.bezpieczenstwa.pl/content/view/377/16/>

<http://centrum.bezpieczenstwa.pl/content/view/409/12/>

<http://manageengine.adventnet.com/products/service-desk/>

[ITIL-help-desk-smb-whitepaper.html?gclid=COSH05KwxJMCFQIMuwodeX7KZA](http://www.itil-help-desk-smb-whitepaper.html?gclid=COSH05KwxJMCFQIMuwodeX7KZA)

[CTPartners.pl](http://www.itil-help-desk-smb-whitepaper.html?gclid=COSH05KwxJMCFQIMuwodeX7KZA)

<http://ctpartners.pl/content/view/127/126/>

http://www.itsmf.si/index.php?option=com_docman&task=cat_view&gid=39&Itemid=56

<http://h41156.www4.hp.com/education/article.aspx?cc=pl&ll=pl&id=813>

http://www.pbsg.pl/images/stories/wiadomosci/kgm_case_study_iso_20000.pdf

<http://itsmf.org.pl/>

http://www.ploug.org.pl/konf_05/materialy/pdf/07.pdf

COBIT

<http://en.wikipedia.org/wiki/COBIT>

http://www.audyt.com/storage/pdf/metodyka/cobit_wytyczne_zarzadzania.pdf

<http://centrum.bezpieczenstwa.pl/content/view/365/89/>

<http://www.isaca.org/Template.cfm?Section=COBIT6&Template=/TaggedPage/TaggedPageDisplay.cfm&TPLID=55&ContentID=7981>

http://www.foi.hr/CMS_home/znan_strucni_rad/konferencije/IIS/2007/papers/T03_05.pdf

http://www.securitycn.net/img/uploadimg/20070831/cobit4.0_en.pdf

Skorowidz

A

AC, 60
ACC, 60
Actual Cost, 60
Actual Cost Cumulated, 60
ACWP, 63
Affinity diagram, 48
Agile MSF, 71
Agile Unified Process, 71, 178
analiza pola siły, 48
Analiza scenariuszy „co, jeśli”, 46
Analiza sieciowa harmonogramu, 45
anarchizacja projektu, 177
Applying Leads and Lags, 44
automatyzacja testów jednostkowych, 186

B

BAC, 60
BCWP, 63
BCWS, 63
Beck Kent, 179
BI, 129
biuro zarządzania projektami, 24
Black Belt, 245
Blanchard Ken, 36
BlogSerwis, 85
 diagram Zaktualizowanej Wartości Netto, 91
 dokument DIP, 91
 Sterowanie Etapem, 101
 tabela SLA, 89
 uruchamianie Fazy Podjęcia, 93
 zakończenie Fazy Przekazania, 100
 Zamykanie Projektu, 101
body shopping, 56
Boehm Barry, 70
Brainstorming, 47

BRM, 249
BTS, 78
Budget At Completion, 60
Budżet Całkowity, 60
Budżetowy Koszt Planowanej Pracy, 63
Budżetowy Koszt Wykonanej Pracy, 63
bufory czasu, 45
Bug Convergence, 118
Bug Tracking Systems, 78
Bug triage, 116
burndown chart, 193
Burza mózgów, 47
Business Case, 17

C

Capability Maturity Model, 226
Capability Maturity Model Integration, 225, 226
CAPM, 66
CAR, 232
CCR, 67
CCTA, 13, 14
Certified Associate in Project Management, 66
Certified Information Security Manager, 274
Certified Information Systems Auditor, 274
Champion, 245
Change Control, 20, 23
Change Request, 23
CISA, 274
CISM, 274
Closing a Project, 16
CM, 232
CMMI, 206, 225
 Analiza decyzji i rozwiązań, 232
 capability level, 233
 CAR, 232
 Causal Analysis and Resolution, 232
 cele dedykowane, 232

CMMI

- cele generyczne, 232
- CL, 233
- CM, 232, 233
- Configuration Management, 232
- continuous representation, 233
- DAR, 232
- Decision Analysis and Resolution, 232
- dedykowane praktyki, 232
- Definicja procesów organizacyjnych, 231
- equivalent staging, 233
- Generic Goals, 232
- Generic practices, 232
- generyczne praktyki, 232
- GG, 232
- GP, 232
- Ilościowe zarządzanie projektem, 230
- Integracja produktu, 231
- Integrated Project Management, 229
- Inżynieria, 231
- IPM, 229
- kategorie procesów, 234
- MA, 232, 233
- maturity level, 228, 233
- maturity model, 233
- Measurement and Analysis, 232
- Mierzenie i analiza, 232
- Modyfikacja procesów organizacyjnych, 230
- Monitorowanie i kontrola projektów, 229
- Nieformalna analiza i rozwiązania, 232
- odpowiednik poziomu, 233
- OID, 230
- OPD, 231
- OPF, 230
- OPP, 230
- Organizational Innovation and Deployment, 230
- Organizational Process Definition, 231
- Organizational Process Focus, 230
- Organizational Process Performance, 230
- Organizational Training, 230
- OT, 230
- PI, 231
- Planowanie projektu, 229
- PMC, 229, 233
- poziomy dojrzałości, 228
- PP, 229, 233
- PPQA, 232, 233
- Procesowe i produktowe zapewnienie jakości, 232
- Process and Product Quality Assurance, 232
- procesy, 229
- Product Integration, 231
- Project Monitoring and Control, 229
- Project Planning, 229
- QPM, 230
- Quantitative Project Management, 230
- RD, 231
- reprezentacja ciągła, 233
- reprezentacja stopniowana, 233
- reprezentacje, 232
- REQM, 231, 233
- Requirements Development, 231
- Requirements Management, 231
- Risk Management, 230
- Rozwiązania techniczne, 231
- RSKM, 230
- SAM, 229, 233
- SG, 232
- SP, 232
- Specific Goals, 232
- Specific practices, 232
- staged representation, 233
- Stworzenie wymagań, 231
- Supplier Agreement Management, 229
- Szkolenia, 230
- Technical Solution, 231
- TS, 231
- VAL, 231
- Validation, 231
- VER, 231
- Verification, 231
- Walidacja, 231
- Wdrożenie innowacji organizacyjnych, 230
- Weryfikacja, 231
- Wsparcie, 232
- Wydajność procesów organizacyjnych, 230
- Zarządzanie konfiguracją, 232
- Zarządzanie procesami, 230
- Zarządzanie projektem, 229
- Zarządzanie ryzykiem, 230
- Zarządzanie umowami z dostawcami, 229
- Zarządzanie wymaganiami, 231
- Zintegrowane zarządzanie projektem, 229
- CMMI 2000, 226
- CMMI dla akwizycji, 226
- CMMI dla programistów, 226
- CMMI dla usług, 226
- CMMI/SPI, 228
- CMMI-ACQ, 226
- CMMI-DEV, 226
- CMMI-SVC, 226
- COBIT, 263
- Acquire and Implement, 263
- AI, 263
- cele działań, 268
- certyfikacja, 273
- Deliver and Support, 263
- Dostarczenie i wsparcie, 263
- DS, 263

kluczowe miary wydajności, 268
ład korporacyjny, 265
ME, 263
miary metryki, 269
modele dojrzałości, 269
Monitor and Evaluate, 263
Monitoring i ewaluacja, 263
Nabycie i implementacja, 263
perspektywy, 271
Plan and Organize, 263
Planowanie i organizacja, 263
PO, 263
Początkujący proces, 270
Powtarzalny, ale intuicyjny proces, 270
poziomy dojrzałości, 270
procesy, 266, 331
produkty, 263
role, 265
szczegółowe cele kontrolne, 267
tabela RACI, 268, 269
wysokopoziomowe cele kontrolne, 266
wytyczne zarządcze, 268
Zarządzany i mierzalny proces, 270
Zdefiniowany proces, 270
Zoptymalizowany proces, 271
codzienne kompilacje, 115, 183
collaboration tools, 38
commercial off-the-shelf, 231
Configuration Management, 19
Continuing Certification Requirements, 67
Control Objectives for Information and related Technology, 263
Controlling a Stage, 15
Controls, 18
COSO, 265
Cost Performance Index, 60
Cost Variance, 60
Cost Variance Cumulated, 60
Cost-Plus-Award-Fee, 52
Cost-Plus-Fixed-Fee, 52
Cost-Plus-Incitive-Fee, 52
Cost-reimbursable, 52
COTS, 231
cowboy coding, 177
CP, 16
CPAF, 52
CPFF, 52
CPI, 60
CPI Cumulated, 61
CPIC, 61, 62
CPIF, 52
crashing, 46
Critical chain method, 45
Critical path method, 45

CS, 15
CSI, 253
CV, 60
CVC, 60, 61
Czarny Pas, 245
Czas, 35
czynniki powodzenia projektu, 74

D

daily build, 113, 183
DAR, 232
dead line, 195
death march, 185
delegowanie, 36
Dependency Determination, 44
Develop preliminary scope statement, 32
diagram relacji, 48
Diagram Rozrzutu, 65
diagram SIPOC, 243
diagram spalania, 193
diagram WBS, 22
diagram Zaktualizowanej Wartości Netto, 91
diagramy macierzowe, 51
diagramy pokrewieństwa, 48
diagramy przepływów, 51
DIP, 91
Directing a Project, 15
DMADV, 241
DMAIC, 240, 241
dokument DIP, 91
dokument otwarcia, 130
dokument żądania zmiany, 23
dokumentacja, 24, 197
Dostawa, 39
dowody wykonalności, 111
DP, 15
drill through, 145
dyscypliny RUP, 76
dystrybucja informacji, 161
dziennik zadań pierwszego sprinta, 212
dziennik zadań produktu, 191, 210
dziennik zadań sprinta, 192

E

EAC, 61, 63
Earned Value, 60
Earned Value Cumulated, 60
Earned Value Management, 32, 59, 144
Earned Value Technique, 32
Edytor szablonów procesowych, 106
Elementy sterowania, 18
eNBeTeP, 181

Enterprise Project Management, 38
 ERP, 162
 Estimate at completion, 61
 Estimate To Complete, 61
 ETC, 61, 63
 ETL, 162
 ETO, 265
 EV, 60
 EVC, 60
 EVM, 32, 59, 144
 EVT, 32
 ewidencjonowanie zgłoszeń i błędów, 78
 Extract, Transform and Load, 162
 eXtreme Prince, 71
 eXtreme Programming, 179
 Czterdziestogodzinny tydzień pracy, 184
 klienci, 180
 KO, 180, 184
 Kolektywne posiadanie kodu, 182
 Małe wydania, 180
 Metafora systemu, 181
 opowieści użytkownika, 180
 PL, 180
 Planowanie gry, 180
 PR, 180
 proces, 180
 programiści, 180
 Programowanie parami, 182
 Prosty projekt, 181
 Przebudowywanie, 182
 Refactoring, 182
 refactoring, 186
 Releases, 180
 role, 180
 Stała integracja, 183
 Stałe testowanie, 182
 Stały dostęp do klienta, 185
 standardy kodowania, 185
 TE, 180
 testy akceptacyjne, 182
 testy jednostkowe, 182
 wdrażanie, 186

F

Fast tracking, 47
 FCST, 63
 feedback, 179
 FFP, 52
 FindBugs, 185
 Firm Fixed Price Contracts, 52
 Fixed price, 52
 Fixed Price Incentive Fee Contracts, 52

Fixed Price with Economic Price
 Adjustment Contracts, 52
 Flowcharts, 51
 FMEA, 242
 FollowUp Actions, 16
 force field analysis, 48
 Formuła Realizacyjna, 14
 FP-EPA, 52
 FPIF, 52
 FTG, 63
 fxCop, 185

G

Green Belt, 245
 GUI, 107

H

histogram, 64

I

Idea/mind mapping, 48
 IDP-CMM, 226
 indeks wydajności niezbędnej
 do zakończenia projektu, 32, 63
 Information Technology Infrastructure Library, 247
 Inicjowanie Projektu, 15, 25
 Initiating a Project, 15
 Inspekcja, 65
 instruowanie, 36
 Integracja, 33
 Interactive Voice Response, 241
 interrelationship diagram, 48
 IP, 15
 IPM, 229
 ISACA, 264
 Issue log, 23
 IT Governance, 265
 ITGI, 264
 ITIL, 247, 248
 7-Step Improvement Process, 329
 Access Management, 329
 Architekt, 249
 architektura, 251
 Availability Management, 327
 BRM, 249
 Business Relationship Manager, 249
 Capacity Management, 326
 certyfikacja, 256
 Change Management, 327
 Chief Sourcing Officer, 249

- Configuration Item, 255
 - Configuration Management Database, 255
 - Continual Service Improvement, 253
 - CSI, 253
 - cykl życia usługi, 253
 - Demand Management, 325
 - Eksploracja usług, 249, 252, 328
 - Evaluation, 328
 - Event Management, 328
 - Ewaluacja, 328
 - Financial Management, 325
 - harmonogram wdrożenia, 255
 - Incident Management, 328
 - Information Security Management, 327
 - IT Service Continuity Management, 327
 - Kierownik projektów, 249
 - Kierownik zasobów, 249
 - Knowledge Management, 327
 - Menadżer bezpieczeństwa, 249
 - Menadżer ciągłości usług IT, 249
 - Menadżer dostępności, 249
 - Menadżer katalogu usług, 249
 - Menadżer poddostawców, 249
 - Menadżer projektowania usług, 249
 - Menadżer przepustowości, 249
 - Menadżer relacji biznesowych, 249
 - Mierzenie usług, 329
 - Planowanie i wsparcie przekazania, 328
 - PM, 249
 - Problem Management, 329
 - procesy, 251, 325
 - Product Manager, 249
 - projektant IT, 249
 - Projektowanie usług, 249, 252, 325
 - Przekazanie usług, 252, 327
 - punkt kontaktu, 255
 - Raportowanie usług, 330
 - Release and Deployment Management, 328
 - Request Fulfillment, 329
 - role, 249
 - SD, 252
 - Service Asset and Configuration Management, 327
 - Service Catalogue Management, 325
 - Service Design, 252
 - Service desk, 249
 - Service Level Management, 257, 326
 - Service Measurement, 329
 - Service Operation, 252
 - Service Portfolio Management, 325
 - Service Reporting, 330
 - Service Strategy, 251
 - Service Transition, 252
 - Service Validation and Testing, 328
 - Siedmioetapowy proces doskonalenia, 329
 - SKMS, 252
 - SO, 252
 - SS, 251
 - ST, 252
 - Strategia usług, 249, 251, 325
 - Strategy Generation, 325
 - Supplier Management, 327
 - systemowa usługa zarządzania wiedzą, 252
 - Transition Planning and Support, 328
 - Tworzenie strategii, 325
 - Ustawiczne doskonalenie usług, 253, 329
 - Walidacja i testowanie usług, 328
 - wdrażanie, 254
 - Wypełnienie żądań, 329
 - zarządzanie aplikacją, 250
 - Zarządzanie bezpieczeństwem informacji, 327
 - Zarządzanie ciągłością usług IT, 327
 - Zarządzanie dostawcami, 327
 - Zarządzanie dostępnością, 327, 329
 - Zarządzanie finansami, 325
 - Zarządzanie incydentami, 328
 - zarządzanie IT, 250
 - Zarządzanie kapitałem i konfiguracją usług, 327
 - Zarządzanie katalogiem usług, 325
 - Zarządzanie portfolio usług, 325
 - Zarządzanie poziomem usług, 326
 - Zarządzanie problemami, 329
 - Zarządzanie przepustowością, 326
 - Zarządzanie publikacjami i wdrożeniami, 328
 - Zarządzanie techniczne, 250
 - Zarządzanie wiedzą, 327
 - Zarządzanie wymaganiami, 325
 - Zarządzanie zdarzeniami, 328
 - Zarządzanie zmianą, 327
 - ITIL® Expert, 257
 - ITIL® V3 Foundation, 257
 - itSMF, 247
 - IVR, 241
- J**
- jakość, 35, 196
 - Jakość w środowisku projektu, 19
 - język UML, 73
 - JUnit, 182
- K**
- Keep Informed, 41
 - Keep Satisfied, 41
 - Key Performance Indicators, 247
 - Kierownik Projektu, 18
 - kluczowe miary wydajności, 247

KO, 180, 184
 Kolektywne posiadanie kodu, 182
 koleżeńskie przeglądy, 16
 Komitet Sterujący, 14, 18
 Kompresja harmonogramu, 46
 Komunikacja, 37
 Koniec-do-Końca, 43
 konsultowanie, 36
 Kontrakt o ustalonej cenie, 52
 Kontrakt o ustalonej cenie z marginesem, 52
 Kontrakt o ustalonej cenie z opcją rewaloryzacji, 52
 Kontrakty zwrotu kosztów, 52
 kontrola harmonogramu, 161
 kontrola jakości, 161
 Koszt, 35
 Koszt Faktyczny, 60
 Koszt z marginesem, 52
 Koszt z premią, 52
 Koszt z ustaloną opłatą, 52
 kruszenie, 46
 krzywa Gaussa, 239

L

Lean Six Sigma, 239
 Lifecycle Objective Milestone, 74, 275

M

MA, 232
 Małe wydania, 180
 Manage Closely, 41
 Management of Risk, 19
 Managing Product Delivery, 16
 Managing Stage Boundaries, 16
 Mapa pomysłów, 48
 marsze śmierci, 185
 matrix diagrams, 51
 Matryca śledzenia wymagań, 32
 matryce priorytetyzacyjne, 51
 MCSD, 126
 MCSE, 126
 mechanizm równoważenia zasobów, 46
 Metafora systemu, 181
 Metoda Diagramów Następstw, 42
 Metoda łańcucha krytycznego, 45
 Metoda ścieżki krytycznej, 45
 metodyki, 7
 PMI, 31
 metodyki adaptacyjne, 177
 eXtreme Programming, 179
 Scrum, 189

metodyki organizacyjne, 223
 CMMI, 225
 COBIT, 263
 ITIL, 247
 Six Sigma, 239
 metodyki wytwórcze, 69
 MSF, 105
 RUP, 73
 metodyki zarządcze, 11
 PMBok, 31
 Prince2, 13
 Micro Independent Software Vendors, 205
 Microsoft Operation Framework, 121
 Microsoft Solution Framework, 105
 Microsoft Visual Studio Team System, 105
 milestones, 144
 model adaptacyjny, 71
 model dojrzałości organizacyjnej, 226
 model iteracyjny, 71
 model kaskadowy, 70
 model przywództwa zespołowego, 37
 model spiralny, 70
 MOF, 121
 Changing, 121
 Operating, 121
 Optimization, 122
 Optymalizacja, 122
 Support, 122
 Wsparcie, 122
 Wykorzystanie, 121
 Zmiana, 121
 Monitor, 41
 monitorowanie, 161
 monitorowanie stabilności rozwiązania, 117
 MP, 16
 MSF, 71, 105, 126
 architektura, 107
 architektura Fazy Konstrukcji, 112
 architektura Fazy Planowania, 110
 architektura Fazy Stabilizacji, 116
 architektura Fazy Wdrożenia, 120
 architektura Fazy Wizji, 108
 Bug Convergence, 117
 Bug Tracking System, 117
 cele biznesowe, 109
 certyfikacja, 126
 check list, 120
 codzienne kompilacje, 113, 115
 daily build, 113
 Deploying, 106
 Developing, 106
 Development, 107
 diagram stanów zgłoszeń, 117

dowody wykonalności, 111
dyscypliny zarządcze, 125
Edukator Użytkownika, 107
Envisioning, 106
Faza Dostarczenia, 106
Faza Konstrukcji, 106, 112
Faza Planowania, 106, 109
Faza Stabilizacji, 106, 116
Faza Wdrożenia, 120
Faza Wizji, 106, 108
fazy, 106
funkcjonalności, 123
Graphical User Interfaceser Experience, 107
harmonogram, 123
interoperability, 110
Kierownik Produktu, 108, 109
Kierownik Programu, 107, 109
Konstruktor, 107
Konstruowanie, 107
lista weryfikacyjna, 120
Logistics Management, 108
model zespołu, 107, 127
monitorowanie stabilności rozwiązania, 117
plan codziennych kompilacji, 113
Planning, 106
Product Management, 108
Program Management, 107
Projektant Interfejsów Użytkownika, 107
projektowanie fizyczne, 110
Projektowanie interfejsów użytkownika, 107
projektowanie koncepcyjne, 110
projektowanie logiczne, 110
Proof Of Concepts, 111
punkt schodzenia się liczby błędów, 117
Release Candidates, 116
Release Management, 108
sesje reanimacyjne, 116
Stabilizing, 106
Tester, 107
Testing, 107
Testowanie, 107
tetraedr negocjacyjny, 124
trampolina braku błędów, 118
trójkąt negocjacyjny, 123, 124
User Education, 107
User Experience, 107
Wdrażanie, 108
Wdrożeniowiec, 108
współdziałanie, 110
zarządzanie gotowością, 125
Zarządzanie logistyczne, 108
Zarządzanie produktem, 108
Zarządzanie programem, 107
zarządzanie projektem, 125

zarządzanie ryzykiem, 125
zasoby, 123
Zero Bug Bounce, 118
MSF dla metodyk adaptacyjnych, 106
MSF dla modelu CMMI, 106
MSF for Agile Software Development, 106
MSF for Capability Maturity Model Integration,
 Process Improvement, 106
MSF3, 106
MSF4, 106
MSF4 for Agile, 178

N

nadzór nad kosztami, 161
nadzór nad pracami projektu, 161
narzędzia współpracy zdalnej, 38
Net Present Value, 91
neverending story, 16
No Single Point Of Failure, 121
Nominal group technique, 47
Nowe Przewidywania, 63
NPV, 91, 92
NUnit, 182

O

obszary wiedzy, 33
Odchylenie Harmonogramowe, 60
Odchylenie Kosztowe, 60
Office of Government Commerce, 14, 247
OGC, 14, 247
OID, 230
OLAP, 161
OPD, 231
OPF, 230
opowieści użytkownika, 180
OPP, 230
Opracowanie harmonogramu, 45
Organizacja, 17
Organization, 17
OT, 230

P

pakiety prac, 16
P-CMM, 226
peer review, 16, 232
PgMP, 67
PI, 231
pierwsza wersja rozwiązania produkcyjnego, 89
PINO, 25
PL, 17, 180

- plan jakości, 143
- plan komunikacji, 143
- plan zarządzania projektem, 140
- plan zarządzania ryzykiem, 147
- Planned Value, 59
- Planning, 17
- Planowanie, 17
- Planowanie dostaw, 52
- Planowanie gry, 180
- Planowanie jakości, 47
- Planowanie oparte na produktach, 21
- planowanie struktury produktowej, 21
- Plans, 18
- Plany, 18
- PM, 249
- PMBok, 31, 161, 277, 278
 - Acquire Project Team, 305
 - Administer Procurements, 318
 - Administrowanie dostawami, 318
 - analiza pola siły, 48
 - Analiza scenariuszy „co, jeśli”, 46
 - Analiza sieciowa harmonogramu, 45
 - Applying Leads and Lags, 44
 - Budget Determine, 294
 - certyfikacja, 66
 - Close Procurements, 320
 - Close Project or Phase, 319
 - Collect Requirements, 287
 - Conduct Procurements, 309
 - Control Costs, 314
 - Control Schedule, 314
 - Create WBS, 289
 - Critical chain method, 45
 - Critical path method, 45
 - Czas, 35
 - Define Activities, 289
 - Define Scope, 288
 - Definiowanie zakresu projektu, 288
 - Develop Human Resource Plan, 296
 - Develop Project Charter, 285
 - Develop Project Management Plan, 286
 - Develop Project Team, 305
 - Develop Schedule, 292
 - diagramy pokrewieństwa, 48
 - Direct and Manage Project Execution, 303
 - Distribute Information, 308
 - Dostawa, 39
 - dostosowanie do potrzeb organizacji, 66
 - Dystrybucja informacji, 308
 - Estimate Activity Durations, 291
 - Estimate Activity Resources, 291
 - Estimate Costs, 293
 - EVM, 59
 - Fast tracking, 47
 - grupy procesów, 40
 - histogram, 64
 - Identify Risks, 299
 - Identyfikacja interesariuszy, 286
 - Identyfikacja ryzyka, 299
 - Ilościowa analiza ryzyka, 300
 - indeks wydajności niezbędnej do zakończenia projektu, 63
 - Identify Stakeholders, 286
 - initiating processes, 285
 - Integracja, 33
 - Integrated Change Control, 311
 - Jakościowa analiza ryzyka, 300
 - Jakość, 35
 - Kierowanie i zarządzanie realizacją projektu, 303
 - Kompresja harmonogramu, 46
 - Komunikacja, 37
 - Kontrakt typu czas i materiał, 52
 - Kontrakty zwrotu kosztów, 52
 - Kontrola harmonogramu, 314
 - Kontrola jakości, 63, 315
 - Kontrola zakresu, 313
 - Koszt, 35
 - Manage Project Team, 307
 - Manage Stakeholders Expectations, 309
 - Mapa pomysłów, 48
 - mechanizm równoważenia zasobów, 46
 - Metoda Diagramów Następców, 42
 - Metoda łańcucha krytycznego, 45
 - Metoda ścieżki krytycznej, 45
 - Monitor and Control Project Work, 311
 - Monitor and Control Risks, 317
 - Monitorowanie i nadzór nad pracami projektu, 311
 - Monitorowanie ryzyk i nadzór nad nimi, 317
 - Nadzór nad kosztami, 314
 - obszary wiedzy, 33
 - Opracowanie dokumentu otwarcia, 285
 - Opracowanie harmonogramu, 45, 292
 - Opracowanie planu zarządzania projektem, 286
 - Organizacja zespołu projektowego, 305
 - Perform Qualitative Risk Analysis, 300
 - Perform Quality Assurance, 304
 - Perform Quality Control, 315
 - Perform Quantitative Risk Analysis, 300
 - Plan Communications, 297
 - Plan Procurements, 302
 - Plan Quality, 295
 - Plan Risk Management, 298
 - Plan Risk Response, 301
 - planning processes, 286
 - Planowanie dostaw, 52, 302
 - Planowanie jakości, 47, 295
 - Planowanie komunikacji, 297

- Planowanie reakcji na ryzyko, 301
Planowanie zarządzania ryzykiem, 298
Planowanie zasobów ludzkich, 296
poziomy wsparcia, 53
Precedence Diagramming Method, 42
procesy, 39, 285
procesy inicjacji, 40, 285
procesy kontroli, 58, 311
procesy planistyczne, 41, 286
procesy realizacji, 57, 303
procesy zamknięcia, 65, 319
prognozowanie, 61
Raportowanie wydajności, 316
Report Performance, 316
Resource leveling, 45
Rozwój zespołu projektowego, 305
Równoważenie zasobów, 45
Ryzyko, 38
Schedule Network Templates, 44
Scope Control, 313
Sequence Activities, 290
Szablony harmonogramu sieciowego, 44
Szacowanie czasu trwania czynności, 291
Szacowanie kosztów, 293
Szacowanie zasobów czynności, 291
Szeregowanie czynności, 42, 290
szybkie śledzenie, 47
tabela SLA, 53
TCPI, 63
Technika analizy zależności, 44
Technika delficka, 47
Technika grupy nominalnej, 47
Technika przyspieszeń i opóźnień, 44
techniki, 39
Umowa typu czas i materiał, 56
uruchamianie projektu, 40
Utworzenie struktury pakietów roboczych, 289
Verify Scope, 312
Weryfikacja zakresu, 312
współbieżność grup procesów, 40
Wybranie dostaw, 309
wykresy kontrolne, 63
wykresy Pareto, 64
Zakres, 34
Zamknięcie dostaw, 320
Zamknięcie projektu lub fazy, 319
Zapewnienie jakości, 304
zarządzanie jakością, 35
Zarządzanie oczekiwaniami interesariuszy, 309
Zarządzanie Wartością Wypracowaną, 59, 61
Zarządzanie zespołem projektowym, 307
Zasoby ludzkie, 35
Zbieranie wymagań, 287
Zdefiniowanie budżetu, 294
Zdefiniowanie czynności, 289
Zintegrowane zarządzanie zmianami, 311
PMC, 229
PMD, 185
PMI, 31
PMP, 66
Początek-do-Końca, 43
Początek-do-Początku, 43
podejście kaskadowe, 69
Podstawowe Założenia Projektu, 14
post-mortem analysis, 16
Postmortem meeting, 195
poziomy gotowości pracowników, 36
PP, 14, 229, 275
PPQA, 232
PR, 180
prawo Parkinsona, 46
Precedence Diagramming Method, 42
prezentacja stanu projektu, 154
Prince, 13
Prince In Name Only, 25
Prince tylko z nazwy, 25
Prince2, 13, 85, 275, 276
architektura procesów, 18
Business Case, 17
certyfikacja, 25
Change Control, 20, 23
Change Request, 23
Closing a Project, 16
Configuration Management, 19
Controlling a Stage, 15
Controls, 18
CP, 16
CS, 15
Directing a Project, 15
dokument żądania zmiany, 23
dokumentacja, 24
dostosowywanie do potrzeb organizacji, 25
DP, 15
Elementy sterowania, 18
Inicjowanie Projektu, 15, 25
Initiating a Project, 15
IP, 15
Issue log, 23
Jakość w środowisku projektu, 19
kierownicy zespołów, 18
Kierownik Projektu, 18, 25
koleżeńskie przeglądy, 16
Komitet Sterujący, 18
komponenty, 17
Management of Risk, 19
Managing Product Delivery, 16
Managing Stage Boundaries, 16

Prince2

- MP, 16
- Organizacja, 17
- Organization, 17
- PINO, 25
- PL, 17
- Planning, 17
- Planowanie, 17
- Planowanie oparte na produktach, 21
- planowanie struktury produktowej, 21
- Plans, 18
- Plany, 18
- PP, 14
- procesy, 14
- Product Based Planning, 21
- Product Breakdown Structure, 21
- Przegląd jakości, 24
- Quality In Project Environment, 19
- Quality Reviews, 24
- SB, 16
- SE, 15
- Starting Up a Project, 14
- Sterowanie Etapem, 15, 25, 101
- Sterowanie zmianami, 20, 23
- Strategiczne Zarządzanie Projektem, 15
- struktura podziału prac, 23
- SU, 14
- szablon rejestru zgłoszeń, 23
- techniki, 21
- Uruchamianie Projektu/Przygotowanie
 - Założeń Projektu, 14
- Uzasadnienie biznesowe, 17
- WBS, 22, 23
- WP, 16
- Zamykanie Projektu, 16
- zarządzanie dokumentacją, 24
- Zarządzanie konfiguracją, 19
- Zarządzanie ryzykiem, 19
- Zarządzanie Wytwarzaniem Produktów, 16
- Zarządzanie Zakresem Etapu, 16
- ZE, 16
- ZP, 16
- ZS, 15
- żądanie zmiany, 23

Prince2 Foundation, 25

Prince2 Practitioner, 26

Prioritization matrices, 51

Process Template Editor, 106

Product backlog, 191

Product Based Planning, 21

Product Breakdown Structure, 21

Program Management Professional, 67

programowanie kierowane testami, 179, 182

Programowanie parami, 182

Project Management Body Of Knowledge, 31

Project Management Office, 24

Project Management Professional, 66

Project mandate, 14

Project Resource Organisation Management

- Plannig Technique, 13

Project Statement of Work, 41

PRojects IN Controlled Environments, 13

PROMPT, 13

PROMPT II, 13

Proof Of Concepts, 111

Prosty projekt, 181

Próby Statystyczne, 65

Przebudowywanie, 182

Przegląd jakości, 24

Przegląd Napraw Defektów, 65

przegląd stanu projektu, 151

przetargi publiczne, 54

Przewidywany Skumulowany Koszt, 63

przypadki użycia, 74

przywództwo, 36

punkt schodzenia się liczby błędów, 117

PV, 59

PZP, 14

Q

QFD, 242

QPM, 230

Quality Function Deployment, 242

Quality In Project Environment, 19

Quality Reviews, 24

R

RACI, 268

raport EVM, 163

raportowanie wydajności, 161

Rational Method Composer, 78, 80, 321

Rational Unified Process, 73

RD, 231

re-engineering, 74

refaktoring, 179, 182, 186

rejestr ryzyk, 147

rejestr zgłoszeń, 23

Release Candidate, 89

relevant stakeholders, 229

REQM, 231

Requirements Traceability Matrix, 32

Resource leveling, 45

Return On Investment, 109

revenue sharing, 52

Review meeting, 194

RMC, 78, 80, 321

- ROI, 109
 - rozwiązania kombinowane, 275
 - Równoważenie zasobów, 45
 - RPN, 244
 - RSKM, 230
 - Run Chart, 65
 - RUP, 70, 73, 85, 126, 275, 276, 278
 - Adapt the process, 79
 - adaptacja do potrzeb organizacji, 80
 - Adaptuj proces, 79
 - Analiza i projektowanie, 77, 322
 - Analysis and Design, 322
 - architektura, 75
 - Balance stakeholder priorities, 80
 - Balansuj pomiędzy priorytetami decydentów, 80
 - Business driven development, 79
 - Business Modelling, 321
 - Change and configuration management, 324
 - Collaborate across teams, 80
 - Construction, 75
 - czynniki powodzenia projektu, 74
 - Demonstrate value iteratively, 80
 - Demonstruj rozwiązania iteracyjnie, 80
 - Deployment, 323
 - dyscypliny, 76, 321
 - Elaboration, 74
 - Elastycznie zwiększaj poziom abstrakcji, 80
 - Elevate the level of abstraction, 80
 - fazy, 74
 - Focus continously on quality, 80
 - High Level Definition, 74
 - Implementacja, 77, 323
 - Implementation, 323
 - Inception, 74
 - Initial Operational Capability Milestone, 75
 - komunikacja, 80
 - Konstrukcja, 75
 - Lifecycle Objective Milestone, 74
 - Low Level Definition, 74
 - Modelowanie biznesowe, 76, 321
 - niski poziom abstrakcji, 74
 - odebranie rozwiązania, 76
 - Opracowanie, 74
 - Podjęcie, 74
 - poziomy, 74
 - prace wytwórcze, 74
 - Product Release Milestone, 76
 - Project Management, 324
 - Przekazanie, 75
 - przepływ pracy, 76
 - przypadki użycia, 74
 - Requirements, 322
 - RMC, 80
 - skupianie się stale na jakości, 80
 - success factors, 74
 - Test, 323
 - Testowanie, 77
 - Transition, 75
 - uruchamianie Fazy Podjęcia, 93
 - use cases, 74
 - Wdrożenie, 77, 323
 - Wymagania, 76, 322
 - wysoki poziom abstrakcji, 74
 - zakończenie Fazy Przekazania, 100
 - Zarządzanie projektem, 78, 324
 - Zarządzanie środowiskiem, 78, 324
 - Zarządzanie zmianą i konfiguracją, 77, 324
 - ryzyko, 38, 147
 - Rzeczywisty Koszt Wykonanej Pracy, 63
- ## S
- SA-CMM, 226
 - SAM, 229
 - SB, 16
 - Scatter Diagram, 65
 - Schedule Network Templates, 44
 - Schedule Performance Index, 60
 - Schedule Variance, 60
 - Schedule Variance Cumulated, 60
 - Schwaber Ken, 190
 - scope creep, 118
 - Scrum, 189, 277, 278
 - architektura, 192
 - back log, 191
 - burndown chart, 193
 - Członek zespołu, 191
 - diagram spalania, 193
 - dokumentacja, 197
 - dziennik zadań, 191
 - Dziennik zadań produktu, 191
 - Dziennik zadań sprinta, 192
 - game, 193
 - gra, 193
 - Interesanci, 191
 - jakość, 196
 - kompletność, 197
 - postgame, 194
 - Postmortem meeting, 195
 - pregame, 193
 - proces, 191
 - Product backlog, 191
 - Review meeting, 194
 - role, 190
 - ScrumMaster, 190
 - spotkanie przeglądowe, 194
 - spotkanie retrospekcyjne, 195
 - Sprint backlog, 192

- Scrum
 - sprinty, 191
 - technika szybkości zespołu, 195
 - ustanowienie gry, 193
 - velocity, 195
 - wady, 220
 - Właściciel produktu, 190
 - zamykanie gry, 194
 - zarządzanie ryzykiem, 197
- Scrum BlogSerwis, 209
 - dziennik zadań pierwszego sprinta, 212
 - dziennik zadań produktu, 210
 - wykres spalania, 220
- ScrumMaster, 209
- SE, 15
- SE-CMM, 226
- SEI, 226
- Service Knowledge Management System, 251, 252
- Service Level Agreement, 53, 89, 121
- Service Level Management, 257
- service provider, 247
- sesje reanimacyjne, 116
- SG, 232
- Single Point Of Contact, 255
- SIPOC, 242
- SIWZ, 54
- Six Sigma, 239
 - certyfikacja, 245
 - Design of Experiments, 244
 - diagram SIPOC, 242, 243
 - DMADV, 241
 - DMAIC, 241
 - Dom jakości, 242
 - Dotkliwość dla klienta, 244
 - Failure Mode Effect Analysis, 242
 - FMEA, 242
 - Głos klienta, 242
 - IVR, 241
 - Karty ocen, 245
 - liczbowa identyfikacja ryzyk, 244
 - odchudzona wersja, 239
 - Planowanie eksperymentów, 244
 - Prawdopodobieństwo wystąpienia, 242
 - procesy, 241
 - QFD, 242
 - Quality Function Deployment, 242
 - RPN, 244
 - Score Card, 245
 - techniki, 242
 - Trudność wykrycia, 242
 - VOC, 242
 - Voice of the Customer, 242
 - wdrażanie, 241
 - Wykresy kontrolne, 245
 - Zarządzanie ryzykiem, 242
 - SKMS, 251, 252
 - Skumulowana Rozbieżność Harmonogramu, 62
 - Skumulowane Koszty Faktyczne, 60
 - Skumulowane Odchylenia Harmonogramowe, 60
 - Skumulowane Odchylenia Kosztowe, 60, 61
 - Skumulowane Wartości Wypracowane, 60
 - Skumulowany Indeks Rozbieżności Harmonogramu, 63
 - Skumulowany Wskaźnik Efektywności Harmonogramowej, 61
 - Skumulowany Wskaźnik Efektywności Kosztowej, 61, 62
 - SLA, 53, 89, 247
 - SO, 252
 - Specyfikacja Istotnych Warunków Zamówienia, 54
 - SPI, 60
 - SPI Cumulated, 61
 - SPIC, 61, 63
 - SPICE, 226
 - spiralność, 70
 - Spolsky Joel, 205
 - spotkania organizacyjne „na stojaka”, 179
 - spotkanie planistyczne, 209
 - Sprint backlog, 192
 - sprinty, 191
 - SS, 251
 - ST, 252
 - Stała integracja, 183
 - stałe przebudowywanie kodu, 179
 - Stale testowanie, 182
 - Standardowy kontrakt o ustalonej cenie, 52
 - standardy kodowania, 185
 - Stand-up meeting, 179
 - Starting Up a Project, 14
 - Sterowanie Etapem, 15, 25
 - Sterowanie zmianami, 20, 23
 - Strategiczne Zarządzanie Projektem, 15
 - struktura pakietów roboczych, 131, 132
 - struktura podziału prac, 22
 - style przywództwa, 36
 - SU, 14
 - SV, 60
 - SVC, 60, 62
 - SW-CMM, 226
 - syndrom pęłających wymagań, 118
 - system BI, 129, 130
 - szablon rejestru zgłoszeń, 23
 - Szablony harmonogramu sieciowego, 44
 - szacowanie kosztów, 150
 - Szacunek Kosztu do Ukończenia, 61
 - Szacunkowy Budżet Całkowity, 61
 - szybkie śledzenie, 47

T

tabela SLA, 53, 89
TCO, 109
TCPI, 32, 63
TE, 180
Technical Solution, 231
Technika analizy zależności, 44
Technika delficka, 47
Technika grupy nominalnej, 47
Technika przyspieszeń i opóźnień, 44
Technika Uzyskanej Wartości, 32
technika wzajemnych testów, 232
Test Driven Development, 179, 182
testy akceptacyjne, 182, 196
testy jednostkowe, 182
tetraedr negocjacyjny, 124
The Delphi Technique, 47
the feeding buffer, 45
The IT Service Management Forum, 247
the project buffer, 45
To-Complete Performance Index, 32, 63
Total Cost of Ownership, 109
trampolina braku błędów, 118, 160
trójkąt negocjacyjny MSF, 124
TS, 231

U

UML, 73
Umowa typu czas i materiał, 56
Uruchamianie Projektu/Przygotowanie Założeń Projektu, 14
uruchomienie projektu, 131
user stories, 180
Uzasadnienie biznesowe, 17

V

VAL, 231
VER, 231
VOC, 242

W

Wartość Uzyskana, 60
Wartość Zaplanowana, 59
WBS, 22, 23, 41, 131
wdrażanie systemu BI, 129
analiza wskaźników, 163
decyzja o wyborze usługodawcy wdrożenia, 149
dokument otwarcia, 130
harmonogram, 156
odpowiedzi oferentów, 148

plan jakości, 143
plan komunikacji, 143
plan zarządzania projektem, 140
plan zarządzania ryzykiem, 147
platforma sprzętowa, 155
PMBok, 161
prezentacja stanu projektu, 154
prognozy wykonania budżetu, 164
przegląd stanu projektu, 151
raport EVM, 163, 168
rejestr ryzyk, 147
ryzyko, 147
struktura pakietów roboczych, 131
szacowanie kosztów, 150
tabela stopnia realizacji procesów PMBoK, 158
trendy realizacji harmonogramu, 163
uruchomienie projektu, 131
WBS, 131
wybór usługi audytorskiej, 139
wybór usługodawcy wdrożenia, 145
zamknięcie oferty, 144
zapytanie ofertowe na usługi audytorskie, 133
zapytanie ofertowe na usługi wdrożeniowe, 135
zarządzanie oczekiwaniami interesariuszy, 176
zintegrowane zarządzanie zmianami, 166
Work Breakdown Structure, 22, 34, 41
work packages, 16
WP, 16
Wskaźnik Efektywności Harmonogramowej, 60
Wskaźnik Efektywności Kosztowej, 60
wspieranie, 36
wybór usługi audytorskiej, 139
wybór usługodawcy wdrożenia, 145
Wykres Postępu, 65
wykresy kontrolne, 63
wykresy Pareto, 64
wytwarzanie odchudzone, 239

X

XP, 179
XPrince, 71

Y

YAGNI, 181
You Are Not Gonna Reed It, 181

Z

Zakres, 34
Zamykanie Projektu, 16
zapewnienie jakości, 161

- zapytanie ofertowe na usługi audytorskie, 133
- zapytanie ofertowe na usługi wdrożeniowe, 135
- zarządzanie dokumentacją, 24
- Zarządzanie konfiguracją, 19
- zarządzanie oczekiwaniami interesariuszy, 161, 176
- zarządzanie projektem, 161
- zarządzanie ryzykiem, 19, 197
- Zarządzanie Uzyskaną Wartością, 32
- Zarządzanie Wartością Wypracowaną, 59, 61, 144
- Zarządzanie Wytwarzaniem Produktów, 16
- Zarządzanie Zakresem Etapu, 16, 34
- zasada YAGNI, 181
- Zasoby ludzkie, 35
- ZE, 16
- Zero Bug Bounce, 118, 160
- Zielony Pas, 245
- zintegrowane zarządzanie zmianami, 166
- Zlecenie Przygotowania Założeń Projektu, 14
- ZP, 16
- ZPZP, 14
- ZS, 15, 275
- zwrotna pętla informacji, 179

PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ HELION



1. ZAREJESTRUJ SIĘ
2. PREZENTUJ KSIĄŻKI
3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW
w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

<http://program-partnerski.helion.pl>

GRUPA WYDAWNICZA

