

Piotr Wróblewski

ZWINNIE DO PRZODU

PORADNIK KIEROWNIKA PROJEKTÓW
INFORMATYCZNYCH



Helion 

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek 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 Helion SA 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 Helion SA nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktor prowadzący: Małgorzata Kulik

Projekt okładki: Maciej Grzegorek

Grafika na okładce została wykorzystana za zgodą Shutterstock.com

Helion SA

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/zwidop>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

ISBN: 978-83-283-6654-1

Copyright © Helion 2020

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

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

Spis treści

Wstęp	11
Część I. Niezbędnik kierownika projektu	17
Rozdział 1. Pojęcia podstawowe	19
Czym jest, a czym NIE JEST projekt?	21
Realia organizacyjne	23
W stronę organizacji zorientowanej projektowo	26
Portfele i programy	27
Ewolucja project managementu	28
Cykl życia projektu	28
Zróbmy to po swojemu	30
Narzędzia	30
Niełatwa ocena sukcesu projektu	31
Triada projektowa	31
Zgubne dziedzictwo błędów planowania	32
Był sukces czy nie?	33
Zarządzanie i zespół	34
Zastosowanie zarządzania projektami	35
Pytania kontrolne	37
Rozdział 2. Uruchamianie projektu	39
Proste trudnego początki	39
Karta projektu	40
Jak dobrze zainicjować projekt?	42
Plan projektu: budowa i utrzymanie	45
Zakres prac	46
Uczestnicy projektu	48
Struktury pozaprojektowe	49
Procedury	50
Szkolenia	52
Infrastruktura	52
Harmonogram prac	52
Budżet	53
Plan projektu zmienia się w czasie!	53
Pytania kontrolne	53

Rozdział 3. Zespół... mitów i zalet	55
Praktyki budowania zespołów projektowych	55
Pozyskiwanie zasobów osobowych	56
Razem czy osobno?	58
Utrzymanie zespołu	59
Budowanie autorytetu kierownika	63
Role w zespole, czyli optymalne ludzi dopasowanie	65
Klasyfikacja dr. Belbina	65
Typologia MTR-i™	66
Ludzie są różni, czyli model typów osobowości Myers-Briggs	67
Pojęcia podstawowe	68
Typy Myers-Briggs w pigułce	69
Role i typy osobowości — konkluzja	70
Zarządzanie konfliktami	71
Pytania kontrolne	72
Rozdział 4. Od WBS do harmonogramu	73
Dualizm projektowy	73
Struktura podziału prac (WBS)	74
Tworzenie WBS	78
Rola WBS w kontroli zakresu projektu	80
Pytania kontrolne	81
Rozdział 5. Zarządzanie zmianami	83
Proces zarządzania zmianami	83
Role w procesie zarządzania zmianami	85
Wpływ zmian na tzw. wydania produktu	86
Pytania kontrolne (bardzo trudne!)	88
Rozdział 6. Sztuka zarządzania wymaganiami użytkownika	89
Użytkownicy i udziałowcy	90
Niezrozumienie wymagań użytkownika	90
Środowisko „upolityczone”	91
Niestabilne wymagania użytkownika	92
Mapa polityczna projektu	93
Specyfika wymagań informatycznych	94
Typologia wymagań informatycznych	95
Jakość wymagań	99
Skuteczne zbieranie wymagań	100
Standardy i specyfikacje branżowe jako źródło wymagań	100
Identyfikacja uczestników procesu zbierania wymagań	101
Praca z klientem	104
Wizualizacja wymagań	104
Praca z przyszłymi użytkownikami	106
Wzorce WBS i rejestry produktów	107
Joint Application Design	107
Pytania kontrolne	109
Rozdział 7. Zarządzanie ryzykiem	111
Pojęcia podstawowe	112
Odkrywanie ryzyk projektowych	115
Rodzaje ryzyka	116
Podział według pochodzenia	117
Podział według natury ryzyka	117

Materializacja ryzyka i jego wpływ na projekt	119
Szablon dokumentowania ryzyka	120
Mapa ryzyka	121
Pytania kontrolne	121
Rozdział 8. Harmonogram doskonały, czyli jak planować, aby się udało	123
Planowanie kontra chaos	123
Zależności pomiędzy zadaniami	125
Związek „zakończ – rozpocznij” (ang. Finish-Start)	126
Związek „zakończ – zakończ” (ang. Finish-Finish)	126
Relacja „rozpocznij – rozpocznij” (ang. Start-Start)	127
Relacja „rozpocznij – zakończ” (ang. Start-Finish)	127
Wprowadzanie opóźnień lub przyspieszeń zadań	127
Ścieżka krytyczna	128
Szacowanie pracochłonności i kosztów prac	129
Przypisywanie zasobów zadaniom	131
Planowanie zadań w trybie „effort-driven”	131
Planowanie zadań z wyłączonym trybem „effort-driven”	136
Sztuka wymyślenia i układania zadań	136
Optymalizacja obciążenia zasobów projektowych	137
Przekazywanie zadań członkom zespołu	139
Kompresja harmonogramu	141
Pytania kontrolne	142
Rozdział 9. Zarządzanie budżetem w projekcie informatycznym	143
Opłacalność projektu w kontekście organizacji	144
Elementy analizy budżetowej w projekcie	145
Koszty własne w projekcie	147
Koszty zewnętrzne w projekcie	148
Planowanie wykorzystania zasobów	149
Kontrolowanie czy raportowanie czasu pracy?	150
Pytania kontrolne	151
Rozdział 10. Śledzenie postępów i metoda Earned Value	153
Pojęcie wersji bazowej harmonogramu	153
Rejestrowanie danych o postępie prac	155
Uprozczone odznaczanie realizacji zadań	155
Pełna rejestracja stanu realizacji prac	156
Metoda wartości wypracowanej (Earned Value)	158
BAC — budżet	159
BCWS — planowany koszt planowanej pracy	159
BCWP — planowany koszt wykonanej pracy (Earned Value)	159
ACWP — rzeczywisty koszt wykonanej pracy	159
EAC — przewidywany budżet całkowity	160
Wskaźniki wydajności i realizacji	160
Earned Value na przykładach	161
Porównanie metody klasycznej z Earned Value	161
Praca z harmonogramem	164
Symulowanie postępu prac	165
Pytania kontrolne	166

Rozdział 11. Dokumentacja projektowa w fazie wytwórczej	167
Raportowanie stanu projektu	167
Dziennik projektu	169
Zespół projektowy	170
Produkty	170
Środowisko sprzętowe i programowe	171
Nota wydania (ang. Release notes)	172
Baza wiedzy	172
Lista ryzyk projektowych	173
Sprawy bieżące, problemy	173
Zdarzenia projektowe	174
Historia zmian w wymaganiach	174
Dokumentowanie spotkań	175
Pytania kontrolne	176
Rozdział 12. Skuteczna komunikacja	177
Sztuka sprawnej komunikacji	177
Komunikacja elektroniczna	180
Poufność i ochrona informacji projektowej	180
Wymiana informacji w projektach	181
Spotkania, czyli jak się nie zagadać na śmierć	182
Pytania kontrolne	184
Rozdział 13. Zarządzanie jakością w projekcie informatycznym	185
Normy ISO serii 9001	186
Model CMM	189
Testowanie oprogramowania	190
Ogólne rodzaje testów	191
Testy нефunkcjonalne	192
Testy z zakresu bezpieczeństwa	192
Organizacja metod testowania	193
Ciągła integracja i kontrola konfiguracji	195
Pytania kontrolne	197
Rozdział 14. Dostawa i zamknięcie projektu	199
Dostawa produktu	199
Strategie wdrażania złożonych systemów informatycznych	201
Szkolenia użytkowników	201
Zamknięcie prac w projekcie	202
Gwarancja i konserwacja	204
Pytania kontrolne	205
Rozdział 15. Podwykonawstwo i używanie darmowego oprogramowania	207
Outsourcing, czyli kupujemy zamiast wytwarzać	207
Proces nabywania usług lub towarów	208
Modele OEM i ODM	212
Zabezpieczenia depozytowe (ang. escrow agreement)	214
Jak się odnaleźć w gąszczu open source	214
Prawa własności intelektualnej	214
Klasyfikacja licencji FOSS	215
Pytania kontrolne	217

Rozdział 16. Certyfikacja project management	219
Certyfikacje PMI	220
PMBOK (Project Management Body of Knowledge)	220
PMP (Project Management Professional)	220
PMI-ACP (PMI Agile Certified Practitioner)	223
Metodyka i certyfikacja PRINCE2	224
Co wyróżnia PRINCE2?	224
Szkolenia i certyfikacja PRINCE2	226
PRINCE2 Foundation	227
PRINCE2 Practitioner	227
PRINCE2 Agile Foundation	227
PRINCE2 Agile Practitioner	228
AgilePM (Agile Project Management)	228
AgilePM Foundation	228
AgilePM Practitioner	229
Pytania kontrolne	229
Część II. Zwinne rewolucje i praktyki PM	231
Rozdział 17. W kierunku zwinnych metod wytwórczych	233
Proces ex machina, czyli próby uporządkowania rzeczywistości	233
Radosna twórczość	234
Normy i standardy	234
Nowoczesne praktyki zarządzania	236
A może zrobimy to inaczej?	236
Crystal Clear	236
Programowanie ekstremalne (XP)	238
Feature-Driven Development (FDD)	240
Pytania kontrolne	242
Rozdział 18. Usprawnić produkcję!	243
Lean Manufacturing, czyli eliminacja strat	243
Lean Software Development	247
Odwzorowanie pojęć Lean Manufacturing	247
Zasady Lean czy zasady Agile?	249
Praktyczne zasady Lean Software Development	250
Cykl Deminga (PDCA)	253
Rozdział 19. Kanban, czyli zrozumieć i ujarzmić flow	255
Kanban w systemach produkcji klasycznej	255
Tablice Kanban	258
Struktura tablicy Kanban	259
Limity WIP	261
Kanban w Agile	261
Rozdział 20. Od iteracji do Agile	263
Zaplanujmy lepsze jutro!	263
(Nieco) lepszy rodzaj planowania	264
35 lat minęło jak jeden dzień... ..	266
Złożone systemy adaptacyjne	269
Manifest Agile, czyli „przykazania” ogólne	271
Zasady Agile, czyli „katechizm” na co dzień	272
Rodzina metodyk Agile	273

Rozdział 21. Agile w wydaniu DSDM	275
Geneza DSDM	275
Cykl życia projektu w DSDM	279
Zespół i role w DSDM	281
MoSCoW, czyli zarządzanie priorytetami	283
Demonstracja wykonanej pracy	284
Komunikacja	285
Umiejętności miękkie	285
Nienadużywanie komunikacji elektronicznej	285
Praktyka wymiany informacji	286
Sztuka definiowania wymagań	287
Rola Kierownika Projektu w DSDM	288
Planowanie i śledzenie postępu	288
Podsumowanie	289
Rozdział 22. Scrum w pigułce	291
Zasady gry Scrum	292
Flow sterowany wymaganiami	293
Organizacja pracy	294
Spotkania w Scrumie	295
Planowanie sprintu	295
Codzienny Scrum, czyli tzw. daily scrum	295
Backlog grooming	295
Przegląd sprintu	296
Retrospekcja sprintu	296
Postęp prac w Scrumie	296
Rozdział 23. Porównanie Agile z metodami tradycyjnymi	297
Triada projektowa raz jeszcze	297
Według planu czy na luzie?	299
Dokumentacja i zarządzanie wiedzą	302
Metody szacowania Agile	304
A może nie szacujemy?	304
Planning Poker	304
Rozdział 24. Jak raportować postęp projektu zwinnego?	307
Kanban w Agile	307
Rejestr Sprintu	308
Wykresy spalania	309
Kumulacyjny diagram przepływu	311
Earned Value w Agile	314
Rozdział 25. Wdrażanie Agile	317
Pierwsze koty za płoty	317
Wymagania użytkownika	319
Historyjki użytkownika, czyli eliminacja rzeczy zbędnych	319
Mapowanie historyjek użytkownika, czyli punkt widzenia użytkownika	321
Użycie WBS w metodach zwinnych?	322
Jaki Agile?	323
SAFe®, czyli skalowalny Agile	323
Large-scale Scrum (LeSS)	326
Disciplined Agile Delivery (DAD)	328
Nexus	330
Pytania kontrolne	330

Część III. Darmowe oprogramowanie do zarządzania projektami 331**Rozdział 26. Przegląd darmowego oprogramowania do zarządzania projektami .. 333**

Projekty pominięte w książce	334
dotproject	334
Maven	335
OdoO	336
Planner	336
RT: Request Tracker	336
Trac	337
Wekan	337
ZenTao	338

Rozdział 27. GanttProject, czyli proste harmonogramowanie 339

Rozpoczynamy nowy projekt	340
WBS	341
Szybkie wprowadzanie parametrów zadań	343
Praca przy użyciu myszy	343
Zależności i inne atrybuty zadań	344
Zaawansowane atrybuty zadań	345
Zasoby	346
Wygląd wykresu	347
Analiza projektu	348
Zaawansowane opcje programu	348

Rozdział 28. ProjectLibre, czyli prawie jak Microsoft Project 349

Rozpoczynamy nowy projekt	350
WBS	352
Tworzenie i edycja parametrów zadań	353
Zależności i inne atrybuty zadań	355
Zasoby	356
Wygląd wykresu	357
Widoki	359
Kalendarze niestandardowe	363
Analiza i raportowanie projektu	364

Literatura 367**Spis rysunków 369****Spis tabel 373****Skorowidz 375**

Rozdział 7.

Zarządzanie ryzykiem

W poprzednim rozdziale skoncentrowaliśmy się na kluczowym dla projektów informatycznych etapie zarządzania wymaganiami. Dobre wymagania oznaczają produkt finalny o wysokiej jakości, złe wymagania oznaczają kłopoty z akceptacją produktu i generalnie ryzyko dla sukcesu przedsięwzięcia projektowego. Zresztą, jakby tak popatrzeć z boku, projekty informatyczne są obciążone olbrzymim ryzykiem niepowodzenia z uwagi na możliwość popełnienia błędów, których skutki zaczynają się objawiać w momencie, gdy już zostały wydane olbrzymie pieniądze i minął cenny czas, nasz najdroższy zasób projektowy!

Celem tego rozdziału jest omówienie bardzo modnej (w pozytywnym znaczeniu tego słowa) dziedziny zarządzania ryzykiem w projektach informatycznych. Zdefiniuję samo pojęcie ryzyka i metody zarządzania nim, tak jak w poprzednich rozdziałach postaram się też uniknąć zbytniego teoretyzowania.

Zarządzanie ryzykiem stało się bardzo modnym pojęciem w czasie pisania tej książki, gdyż w ostatniej dekadzie nastąpiły spektakularne wydarzenia, których nikt nie potrafił przewidzieć, a które zrujnowały wiele renomowanych firm i zachwiały wiarę w stabilność ekonomii — poniżej przedstawiam najdobitniejsze przykłady.

- ◆ Upadek renomowanego banku inwestycyjnego Barings w 1995 roku z powodu spekulacji nieuczciwego pracownika. Powodem bankructwa był brak nadzoru nad „kreatywnym” pracownikiem, który obracał bez żadnej kontroli olbrzymimi kwotami i popełnił kilka kardynalnych błędów podczas procesu inwestowania.
- ◆ Zniszczenie w wyniku zamachu terrorystycznego siedziby World Trade Center w Nowym Jorku (11 września 2002 roku), miejsca pracy kilku tysięcy ludzi i jedynej siedziby wielu spółek, które utraciły w ciągu jednego dnia wszystkich swoich pracowników oraz znaczące zasoby materialne i bezcenne dane informatyczne, które nie były zarchiwizowane poza siedzibą firmy.
- ◆ Także w 2002 roku upadł koncern Enron, doprowadzając do ruiny wielu inwestorów, którzy ufali jego kłamliwej księgowości, polegającej głównie na traktowaniu przyszłych dochodów jako bieżących zysków.
- ◆ W 2003 roku doszło do spektakularnego załamania się sieci energetycznej w USA — tysiące ludzi zostało pozbawionych dostępu do energii elektrycznej w kraju, który jest kojarzony z najnowszymi osiągnięciami techniki. Powody tej awarii do dziś nie są do końca jasne, na pewno istotne tu były przestarzałe i źle zaprojektowane sieci przesyłowe, ale i zadecydowały zwykłe błędy ludzkie i czynniki trudne do kontrolowania, takie jak deregulacja rynku dostawców energii.
- ◆ W 2004 roku zaczęła już być powoli widoczna napędzana spekulacyjnie hossą na rynku cen ropy.

- ◆ W 2008 roku ujawniła się katastrofa na rynku poręczeń finansowych: okazało się, że czołowe amerykańskie instytucje finansowe pożyczły olbrzymie środki klientom, którzy nie byli w stanie spłacić kredytów. Papiery tych samych banków były często wykupywane przez światowe instytucje finansowe jako „bezpieczne” poręczenia, tymczasem okazały się praktycznie bez wartości. Pomimo szybkiego dokapitalizowania przez państwo zagrożonych upadkiem instytucji (np. Fannie Mea i Freddie Mac), przez giełdy całego świata przebiegła fala panicznej wyprzedaży, przybliżając widmo globalnej recesji, która w 2009 roku stała się faktem!
- ◆ W połowie 2008 roku kurs franka szwajcarskiego znajdował się na rekordowo niskim poziomie 1,96 złotego i był to najniższy kurs franka od 1995 roku. W tym czasie banki namawiały swoich klientów do zaciągania kredytów właśnie we frankach, wskazując na stabilność kursów tej waluty, i w efekcie wytworzyła się grupa kredytobiorców o nazwie „frankowicze”, liczących, że finansując zakup nieruchomości i kredytuując ją we frankach, zrobią interes życia. Jeżeli jednak uwzględnimy zmiany kursu franka, który po 2009 roku po *raz pierwszy* gwałtownie podrożał, wysokie „spready” i spadek wartości nabytych mieszkań oraz domów, to okaże się, że frankowicze dziś muszą spłacać koszty kredytu często dwukrotnie przekraczające wartość inwestycji na rynku mieszkaniowym (ewentualna sprzedaż mieszkania nie zrekompensowałaby straty, wypowiedzenie kredytu nie opłaca się z powodu konieczności zwrotu pieniędzy po *aktualnym* kursie). Interesujące jest, że dzisiaj specjaliści twierdzą, że wzrost kursu franka był do przewidzenia, ale ryzyka, że tak się stanie, nie uwzględniali niedoinformowani klienci, a banki, dysponujące sztabem prawników, znakomicie zabezpieczyły swoje interesy w umowach, które klienci podpisywali często bez czytania...
- ◆ Pandemia COVID-19 w 2020 roku doprowadziła do katastrofy na rynku usług turystycznych, wstrzymane zostały imprezy masowe, a linie lotnicze znalazły się na granicy bankructwa. Skutki tej epidemii nie są jeszcze możliwe do oszacowania w trakcie pisania książki, ale z pewnością będą fatalne dla wielu branż i dziedzin gospodarki.

Niektóre podane powyżej przykłady dotyczą głównie operacji finansowych, gdzie ryzyko jest szczególnie duże, jednak już przykład WTC może dotyczyć każdej firmy, która nie dokonała odpowiedniego zabezpieczenia swoich zasobów strategicznych. Przykład z eksploatującymi cenami ropy pokazuje, że nie zawsze można bezpiecznie zaplanować koszty (zazwyczaj wzrost cen surowców pierwotnych ekstrapoluje się wielkością inflacji).

Pojęcia podstawowe

Definicji ryzyka jest sporo, niektóre z nich mają charakter praktyczny, inne zbliżają się niebezpiecznie blisko do pojęć związanych z rachunkiem prawdopodobieństwa. Myślę, że na potrzeby tej książki wystarczy nam prosta definicja przedstawiona poniżej.



Definicja

Ryzyko w projekcie informatycznym jest niepomyślnym dla przebiegu projektu zdarzeniem, prowadzącym do niepożądanego rezultatu.

Ryzyko = potencjalny problem!

Zarządzanie ryzykiem = unikanie lub neutralizacja skutków niepomyślnych dla projektu wydarzeń.

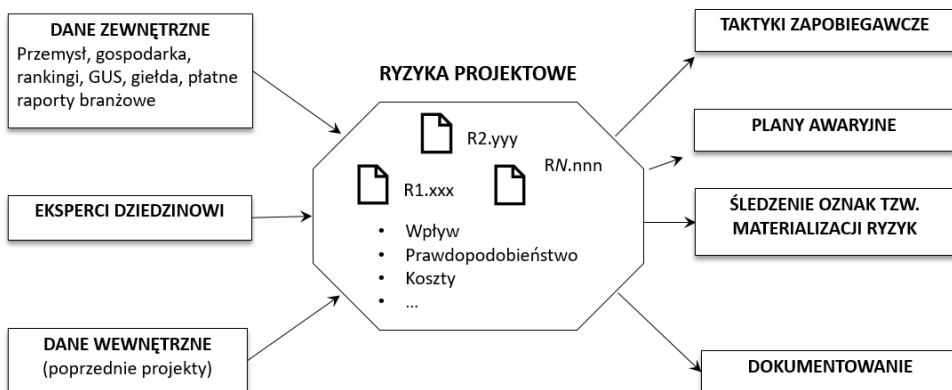
Cała dyskusja o zarządzaniu ryzykiem sprowadza się zatem w sumie do *metod przewidywania i unikania problemów*. Czy brzmi to trochę jak wróżenie z fusów? Owszem brzmi, ale wbrew

pozorem nie omawiamy tutaj sztuki dywagowania, tylko techniki bazujące na danych historycznych i logicznym rozumowaniu, którego nadmiar nigdy nie powinien nam zaszkodzić. Obserwacja znanych katastrof i analiza przyczyn ich wystąpienia pokazuje, że przesłanki wskazujące na katastrofę istniały od dawna, tylko nie zostały na czas dostrzeżone. Pomijając przypadki działań wrogich lub kryminalnych (np. atak na WTC, przekręty finansowe rządów firm), „zwyczajne” katastrofy są często kompozycją wielu drobnych zdarzeń o niskim prawdopodobieństwie wystąpienia, które perfidnie zaszły w tym samym momencie, prowadząc do znacznie groźniejszych zdarzeń!

Zastanawiając się nad pojęciem zarządzania ryzykiem, na pewno prędzej czy później dojdziemy do wniosku, że najtańszą opcją jest *unikanie ryzyka*. Podstawowym problemem takiej opcji jest jednak jej bezsens ekonomiczny, gdyż to, co potencjalnie przynosi zysk, jest z założenia ryzykowne, a coś, co nie niesie ze sobą ryzyka, nie jest interesujące z ekonomicznego punktu widzenia!

Inna strona medalu, ściśle powiązana z zarządzaniem ryzykiem, dotyczy *kosztu obsługi tego procesu*: jeśli chcemy się aktywnie przygotować na zmaterializowanie się ryzyk projektowych, to musimy rezerwować na ich obsługę czas i zasoby, na co zazwyczaj z trudem godzi się kierownictwo, ceniące aktywności wyłącznie „produkcyjne”. Niestety kosztów tych nie da się wyeliminować i lepiej zarezerwować nieco budżetu projektowego na aktywne zarządzanie ryzykiem (przewidywanie, plany awaryjne), niż uprawiać kosztowne „gaszenie pożarów”.

Proces zarządzania ryzykiem w projekcie został poglądowo przedstawiony na rysunku 7.1.



Rysunek 7.1. Proces zarządzania ryzykiem

Kluczowym elementem jest *lista ryzyk projektowych*, spisanych w uzgodnionej formie (np. formularze, opisy pełnym tekstem). Konstruuje ją zazwyczaj kierownik projektu (albo nawet dedykowana osoba odpowiedzialna za zarządzanie ryzykiem).

Oczywiście kierownik nie pracuje w tym krytycznym momencie sam, lista zagrożeń powinna być stworzona przy pomocy uczestników projektu i nawet ekspertów zewnętrznych. Podstawowym kryterium klasyfikacji zdarzenia (zjawiska) jako ryzyka jest doprowadzenie projektu do niepożądanego rezultatu, np. przekroczenia budżetu, konieczności przebudowy znacznej części systemu, opóźnienia instalacji u klienta.

Jeśli jest to możliwe, można ryzykom przypisać następujące atrybuty:

♦ **Prawdopodobieństwo** wystąpienia ryzyka

Zazwyczaj obok miar numerycznych (od 1% do 99%) stosuje się miary umowne (np. niskie, średnie, wysokie...). Oszacowanie prawdopodobieństwa wystąpienia

ryzyka powinno być racjonalne, metoda wrózenia z fusów może narazić cały proces na śmieszność.

◆ *Wpływ na projekt*

Podobnie jak wyżej stosujemy miary umowne, np. duży, mały, średni. Miara wpływu na projekt jest stosowana do filtrowania listy ryzyk tak, aby podczas przeglądów koncentrować się szczególnie na tych o najwyższym wpływie na projekt.

◆ *Koszty związane z obsługą ryzyka lub koszty samego wystąpienia* — w tym przypadku mówimy już o pieniądzach!

◆ *Czas ważności* — nie każde ryzyko jest istotne i ważne podczas całego „cyklu życia” projektu.

◆ *Wartość ryzyka*, czyli jego prawdopodobieństwo wystąpienia pomnożone przez koszty z nim związane, pokazuje w sposób unormowany faktyczną wartość danego ryzyka.



Uwaga

Utworzenie listy ryzyk projektowych jest jedną z kluczowych czynności inicjujących projekt i występuje zazwyczaj raz, na samym początku, choć w dużych projektach, zwłaszcza tych o długim czasie realizacji, nowe ryzyka mogą się pojawić podczas przeglądów. Monitorowanie projektu pod kątem ich wystąpienia powinno stać się stałym elementem zarządzania projektem informatycznym.

Lista ryzyk projektowych nie powstaje z kapelusza, najlepiej bazować tu na znajomości przebiegu poprzednich projektów, podobnych do obecnego, korzystać z powszechnie znanych danych zewnętrznych (prawo, otoczenie ekonomiczne), ewentualnie z pomocy zewnętrznych ekspertów, co niestety bywa kosztowne. Wbrew pozorom wynajdywanie ryzyk nie jest działalnością magiczną, ale umiejętnością znajdowania w naszej działalności biznesowej (przekładającej się na czynności technologiczne w projekcie) swoistych pięt achillesowych. Stare hasło „przezorny zawsze ubezpieczony”, wylansowane w swoim czasie przez PZU, staje się jak najbardziej aktualne!

W momencie gdy znamy już ryzyka projektowe i wstępnie je sklasyfikowaliśmy pod względem ważności czy też innych kryteriów, o których jeszcze powiemy w tym rozdziale, musimy przygotować się do ich obsługi.

Oto powszechnie stosowane praktyki:

◆ *Ignorujemy ryzyko (a nuż się nic nie stanie).*

Jeśli nic nie robimy, to istnieje szansa, że ryzyko się nie zmaterializuje (zerowy koszt obsługi). Jednak w przeciwnym razie koszt związany z obsługą może doprowadzić do znaczącego przekroczenia budżetu lub wręcz upadku projektu!

◆ *Zakładamy, że ryzyko się zmaterializuje.*

Przykładowo, projektujemy system, aby obsłużyć jego wystąpienie (np. poprzez sterowanie danymi referencyjnymi, dokładanie konfiguracji sprzętowych na użytek obsługi sytuacji awaryjnych itp.). Jeśli zakładamy, że system musi być na tyle uniwersalny, aby obsłużyć ewentualne wystąpienie ryzyka, to musimy mieć świadomość, że uniwersalizm ma swój koszt!

◆ *Przygotowujemy plany awaryjne na wypadek wystąpienia ryzyka.*

Co powiedzieć o planach awaryjnych — czy one są „darmowe”? Oczywiście, że nie — dobry plan awaryjny wymaga często pogłębionej analizy, zbliżonej w swojej złożoności do pracy innowacyjnej nad nowym systemem!

◆ *Nikomu nic nie mówimy o wykrytym zagrożeniu.*

Każda z powyższych opcji ma swoje zalety i wady. Ostatnia opcja wbrew pozorom nie jest taka nieprawdopodobna. Nie zawsze osoba odkrywająca ryzyko ma dobre intencje i zechce się pochwalić swoją wiedzą.

Aby dane ryzyko projektowe zostało zaakceptowane jako podstawa do rezerwacji budżetu na jego zminimalizowanie (uniknięcie, naprawę szkód post factum), musi zostać dobrze udokumentowane. Skąd jednak brać dane do oceny szkód związanych z wystąpieniem ryzyka w momencie, gdy cały proces przypomina wróżenie z fusów? Jak zwykle okazuje się, że najlepiej odnieść się do własnych lub cudzych doświadczeń z przeszłości i pokazać udokumentowany przykład kosztów związanych z wystąpieniem danego ryzyka. Aby jednak było do czego sięgać, musimy dysponować danymi historycznymi, a ich dostępność silnie zależy od kultury organizacji, w której są realizowane projekty, i od jej dojrzałości w procesie zarządzania projektami. Bywa i tak, że to nasz projekt będzie tym pierwszym, który dostarczy danych historycznych w naszej organizacji!

Zidentyfikowana lista ryzyk stanowi ważny dokument projektowy i powinna być od czasu do czasu (w sposób zaplanowany) weryfikowana w trakcie realizacji projektu. Oto typowe działania:

- ◆ śledzenie oznak materializacji ryzyk,
- ◆ sprawdzanie wszelkich oznak świadczących o przyszłym lub już aktualnym wystąpieniu ryzyka,
- ◆ podejmowanie decyzji o uruchomieniu działań (tatykt) zapobiegawczych,
- ◆ podejmowanie decyzji o uruchomieniu planów awaryjnych,
- ◆ aktualizacja listy (statusy, zawartość).

Kierownik projektu powinien dbać o szczegółowe archiwizowanie danych o ryzykach projektowych, zarówno dokumentacji ryzyk, jak i notatek z ich obsługi w przypadku wystąpienia zagrożenia. Dane te mogą się okazać bezcenne dla organizacji (firmy) w przyszłych projektach, ponieważ pewne ryzyka „branżowe” mają dość typowy charakter.

W dalszych częściach rozdziału omówimy szczegółowo opisany tutaj proces, podając sporo praktycznych przykładów, które być może lepiej wytłumaczą zagadnienie zarządzania ryzykiem.

Odkrywanie ryzyk projektowych

Nie ma chyba lepszej metody na wykrycie (odkrycie, wymyślenie?) ryzyk projektowych niż ukierunkowana dyskusja na ten temat, angażująca zespół projektowy i udziałowców, którzy mogą mieć coś do powiedzenia w tej sprawie.

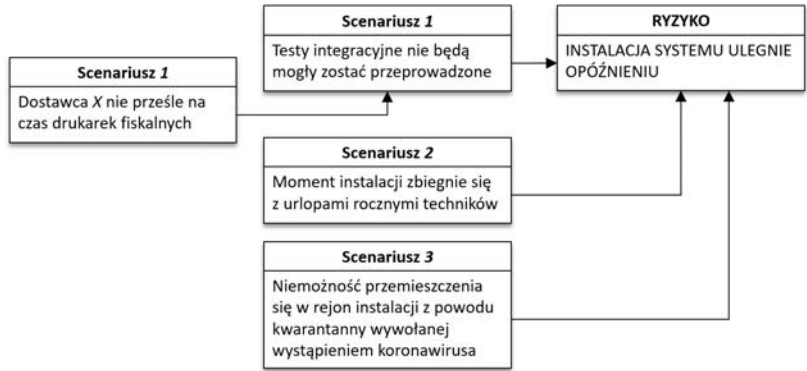
W kwestii samej metody dotyczącej wyszukiwania ryzyk warto stosować następujące podejścia:

- ◆ Burzę mózgow — uczestnicy wymyślają wszelkie katastrofy i problemy, z którymi teoretycznie można będzie się spotkać.
- ◆ Wymyślanie scenariuszy katastroficznych, które mogą spowodować materializację ryzyk projektowych (patrz przykład na rysunku 7.2).
- ◆ Analizę przyczyn pierwotnych, które mogą doprowadzić do uruchomienia niekorzystnego scenariusza.

Prowadzący spotkanie powinien uświadomić jego uczestnikom, że celem jest wyłącznie wykrycie jak największej liczby zagrożeń, a nie wymyślenie sposobu zaradzenia im. Ten etap oczywiście nastąpi, ale już po sklasyfikowaniu ryzyk. Osoba prowadząca dyskusję

Rysunek 7.2.

Poszukiwanie scenariuszy katastroficznych



powinna być wytrenowana w technikach moderacji i elementach manipulacji treścią dyskusji, np. poprzez prowokacje, zmiany punktów widzenia. Do notowania rezultatów warto użyć wykwalifikowanego asystenta, gdyż jednocześnie prowadzenie spotkania i notowanie jego przebiegu bywa kłopotliwe.

Analiza *przyczyn pierwotnych* zakłada zaawansowaną pracę analityczną mającą na celu wykrycie rzeczywistych przyczyn problemów, często głęboko ukrytych w procesie i użytych technologiach projektowych. Metodę tę można porównać do odszukiwania przyczyn choroby, w przeciwieństwie do koncentrowania się na jej objawach.



Uwaga

Ryzyko może zgłosić w zasadzie każdy, kto uważa, że powinien to zrobić, jednak analizę ryzyka (atrybuty, klasyfikacja) warto wykonać już wspólnie.

Rodzaje ryzyka

Pojęcie ryzyka w projekcie informatycznym jest szczególnie złożonym zagadnieniem z uwagi na jego potencjalną wielopłaszczyznowość:

- ◆ technologia,
- ◆ skomplikowane procesy wytwórcze,
- ◆ wpływ zagadnień natury prawnej na aspekty projektowe,
- ◆ silne otoczenie konkurencyjne występujące w większości projektów informatycznych (jakkolwiek by było, informatyka i jej narzędzia strywalizowały się w dużym stopniu),
- ◆ zastosowanie komponentów firm trzecich (więcej na ten temat w rozdziale 15. „Podwykonawstwo i użytkowanie darmowego oprogramowania”),
- ◆ silny wpływ czynnika ludzkiego na proces zbierania wymagań i sam proces wytwórczy.

Wyszukując ryzyka projektowe, warto zastosować przedstawioną dalej typologię, która, podobnie do typologii wymagań, systematyzuje katalogowanie ryzyk i zasadniczo ułatwia ich identyfikację. Aplikując poniższe sugestie, warto jednak nie stosować ich zbyt mechanicznie, gdyż dokument oceny ryzyka projektowego może zamienić się w pozbawiony konkretnych bełkot, który na pewno nie wyda się przekonujący naszemu kierownictwu, nie mówiąc już o najbliższych współpracownikach.

Podział według pochodzenia

Analizując projekt pod kątem ryzyka, można dokonać jego *personifikacji jako bytu podległego potencjalnym zagrożeniom znajdującym się w najbliższym i coraz dalszym otoczeniu*, np.:

- ◆ Projekt sam w sobie jest zbudowany na silnie ryzykownych założeniach, np. klient zażyczył sobie użycia technologii IBM WebSphere, a my mamy dość nikłe pojęcie o możliwościach jej wykorzystania, o czym klient może nie wiedzieć! (BLISKO).
- ◆ Realizacja projektu może być kłopotliwa z uwagi na długi urlop chorobowy kluczowego pracownika, eksperta od nietypowych technologii (BLISKO).
- ◆ Wewnątrz naszej własnej firmy mamy słabe wsparcie od kierownictwa (NIECO DALEJ).
- ◆ Firma przeżywa kłopoty finansowe lub nastroje w niej panujące są złe, co zakłóca procesy produkcyjne (NIECO DALEJ).
- ◆ Nasz klient sprawia niezrozumiałe dla nas trudności podczas procesu zbierania wymagań, mamy słabe wsparcie ze strony jego pracowników oddelegowanych do obsługi projektu (DALEJ).
- ◆ Klient być może okaże się niewypłacalny (DALEJ).
- ◆ Wymogi prawne, na których jest oparty projekt, być może niedługo ulegną zmianie przez ustawodawcę (ZEWNĘTRZNE OTOCZENIE EKONOMICZNO-PRAWNE).
- ◆ Bank nie chce udzielić nam kredytu na realizację projektu (JW.).
- ◆ Grozi nam stan wyjątkowy (mało obecnie realne, ale...) (DALEKA ABSTRAKCJA).
- ◆ Szykuje się wojna z państwem X, która może zakłócić procesy gospodarcze, na których oparte jest bezpieczeństwo naszego projektu (np. kursy walut) (DALEKIE, CHOĆ REALNE ZAGROŻENIE).

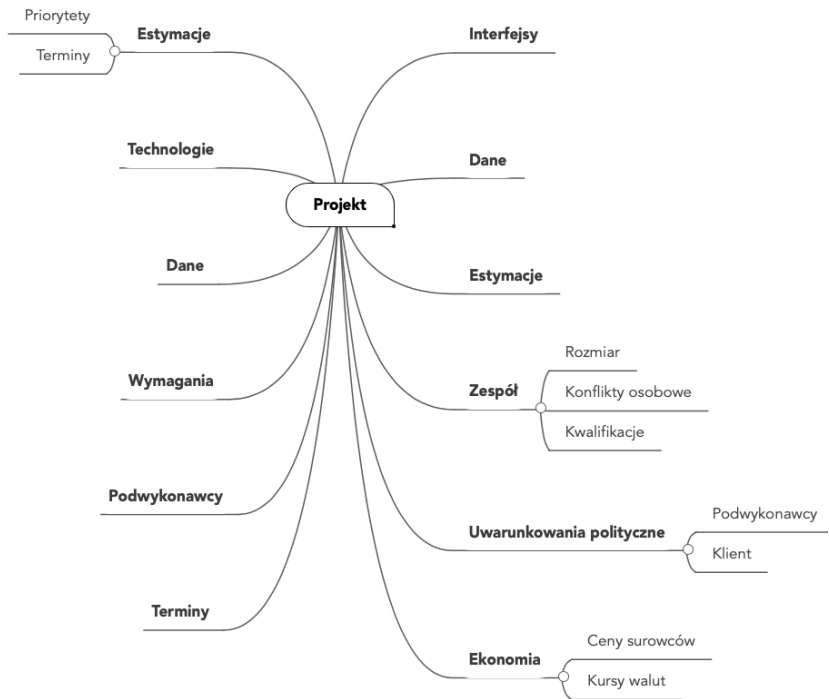
Podobnych przykładów można wymienić sporo (patrz rysunek 7.3) i dość pochopnie można ulec złudzeniu, że względnie łatwo się je konstruuje. Warto jednak nie uprawiać fantastyki naukowej lub politycznej i skoncentrować się wyłącznie na zagadnieniach mających faktycznie wpływ na nasz projekt i których wystąpienie może prowadzić do niepożądanego dlań rezultatu.

Podział według natury ryzyka

Natura ryzyka to inaczej mówiąc jego istota, cechy charakterystyczne. W innowacyjnych projektach informatycznych źródłem ryzyka jest *technologia*, np.:

- ◆ konieczność użycia pewnych nowinek technicznych, nowego języka programowania;
- ◆ Pojawienie się nowej platformy sprzętowej lub programistycznej;
- ◆ złożone przepływy danych pomiędzy wieloma systemami — w dużych projektach integracyjnych ryzyko to eksploduje w momencie prób wymiany informacji pomiędzy systemami zbudowanymi na odrębnych zasadach technicznych (np. różne systemy operacyjne, różne protokoły, technologia plików płaskich przeciwstawiona bazie danych, odmienne protokoły komunikacyjne);
- ◆ wyzwania związane z bazami danych (wielkość, szybkość dostępu, bezpieczeństwo danych, spójność informacji, odporność na awarie);
- ◆ cykl życia produktów firm trzecich, np. długość okresu wsparcia w zakresie awarii lub części zamiennych.

Rysunek 7.3.
Przykładowe obszary wyszukiwania ryzyka w projektach informatycznych



Bazując na poprzednim przykładzie, można odszukać pozostałe *wymiary*, w których można wyszukiwać ryzyka:

- ◆ *czynniki ludzkie* (choroby, nieobecności, sabotaż, ciągłość zatrudnienia personelu, konflikty odpowiedzialności na styku kierownik projektu i kierownik funkcyjny, wzajemne dopasowanie ludzi w zespole itd.);
- ◆ *kwestie finansowe* (budżet, wypłacalność, płynność finansowania);
- ◆ wewnętrzne lub zewnętrzne *problemy organizacyjne*, np. procesy komunikacyjne, słaba infrastruktura projektowa, procedury składania zamówień, regulacje wydawania przepustek do obiektów szczególnie chronionych;
- ◆ *czynniki natury prawnej* (nasz produkt może być czuły na pewne kwestie legislacyjne, które leżąc w gestii ustawodawcy, są praktycznie poza kontrolą podmiotów gospodarczych, a nawet jeśli lobbowanie jest możliwe, oznacza to bardzo długotrwały proces);
- ◆ *czynniki natury gospodarczej* (kursy walut, stopy bazowe, od których zależy koszt finansowania projektu, zdolność kredytowa według banku).

W wielu publikacjach do czynników ryzyka zalicza się także elementy związane z otoczeniem konkurencyjnym, choć bywają one niewymierne. Jak bowiem ocenić ryzyko, że konkurencja szybciej dostarczy na rynek produkt mogący zagrozić naszemu własnemu? Aby realnie oceniać takie ryzyka, trzeba posiadać co najmniej podstawy w postaci danych z wywiadu, uważnie obserwować przetargi i wydarzenia w naszej branży, a to wszystko kosztuje sporo, jeśli nie stanowi głównej działalności naszej firmy!

Materializacja ryzyka i jego wpływ na projekt

Ryzyko, które się zmaterializowało, wpływa w określony sposób na projekt i tak naprawdę wiedza o tym, co może projektowi zaszkodzić, jest jedynym sensownym sposobem na dobre określenie listy ryzyk projektowych.

W projektach informatycznych jest zalecane szczególne zwracanie uwagi na ryzyka, które mają wpływ na terminy wykonania, jakość produktu i budżet (koszty). Czy brzmi to znajomo? Oczywiście, są to przecież znane nam z rysunku 1.5 czynniki krytyczne w projekcie!

Chciałbym zwrócić szczególną uwagę kierowników projektów na dwa elementy, które już na starcie ustawiają projekt na prostej drodze ku klęsce:

- ♦ *Niedoszacowanie pracochłonności* (błędne estymacje dotyczące planowanego wysiłku i czasu programistów).

Konsekwencje: niska jakość harmonogramu prac (szybko się dezaktualizuje), błędy w przypisaniu zasobów do zadań.

- ♦ Błędy popełnione na etapie *zbierania wymagań* (patrz poprzedni rozdział).

Konsekwencje: niska jakość produktu, ale i harmonogramu, który może nie zdzierżyć pojawienia się dodatkowych prac, związanych z doprecyzowaniem wymagań w fazie końcowego wytwarzania produktu.

Te dwa elementy powodują ewidentne kłopoty w realizacji projektu. Oczywiście może się zdarzyć, że dołożą się do nich inne elementy, np. zły (niekompetentny) zespół, niewydajność, błędne zaplanowanie prac i wynikające z niego złe wykorzystanie zdolności produkcyjnych zespołu, przypadki losowe... i mamy katastrofę!

Sam projekt nie jest zawieszony w próżni i jego kłopoty mogą się negatywnie odbić na kondycji i reputacji całej firmy. Czy do firmy, która dostarczyła na rynek niedziałający system wyborczy, przyjdzie jeszcze ktoś z nowym zleceniem? Byłoby to nieco... ryzykowne.

Ryzyka projektowe mogą się materializować w sposób nagły (dotyczy to zwłaszcza zjawisk zewnętrznych, nad którymi słabo panujemy, np.: prawo, polityka, gospodarka) lub w miarę kontrolowany (np. intuicja nam podpowiadała, że dostawca drukarek fiskalnych nie wyrobi się z naszym zamówieniem). W każdym przypadku należy jednak z mniejszym lub większym zaangażowaniem śledzić otoczenie, aby jak najwcześniej wykryć oznaki nadchodzących problemów. Oczywiście nie zawsze jest to możliwe (bo przypadki całkowicie losowe są w gestii sił wyższych), jednak wbrew pozorom sporo elementów w projekcie jest kontrolowanych (lub co najmniej daje się obserwować):

- ♦ Błędne (nierealne) oszacowania terminów lub kosztów są zazwyczaj rezultatem uległości wobec kierownictwa, które ma tendencję do żądania cudów bez oglądania się na realia.
- ♦ Problemy urlopowe można było przewidzieć już w lutym, nie czekając na lipiec, a daty długich weekendów są znane co najmniej na rok naprzód w przypadku poczynienia drobnej inwestycji w... kalendarz biurowy.
- ♦ Dokumentacja z poprzednich projektów w naszej firmie często zawiera ciekawe informacje, które można wykorzystać do przewidywania przyszłości w naszym projekcie.
- ♦ W dużych firmach warto sięgać po porady ekspertów, którzy chętnie dzielą się swoją wiedzą, jeśli tylko ładnie poprosimy o ich pomoc.

Podobnych przykładów można by wymieniać sporo. Sprowadzają się one do prostej porady, aby nie dać się uspić i wykazywać daleko idącą „czujność rewolucyjną”.

Szablon dokumentowania ryzyka

W celu łatwiejszego zarządzania listą ryzyk warto usystematyzować ich dokumentowanie na podobnej zasadzie jak wymagań projektowych, tj. z użyciem numerowania i formy tabelarycznej.

Elementy, które powinny być dokumentowane, na przykładzie prostego ryzyka, znajdziemy w tabeli 7.1.

Tabela 7.1. Szablon dokumentowania ryzyka

ID	Prawd.	Autor	Data rejestracji	12.03.2020
R5	5% (niskie)	Jan Kowalski	Data wystąpienia	
Opis krótki		Niewydajność bazy danych.		
Klasyfikacja (rodzaj ryzyka)		Środowisko techniczne.		
Opis pełny		Istnieje możliwość, że wybrana baza danych nie poradzi sobie z przetworzeniem zapytań podczas zamykania okresów rozliczeniowych.		
Wczesne wskazania		Duże logi transakcyjne, wysoka zajętość zasobów serwera, wolny czas reakcji systemu transakcyjnego.		
Środki zapobiegawcze		<ol style="list-style-type: none"> 1. Optymalizacja zapytań SQL. 2. Inna baza danych. 3. Zmiana schematu bazy danych (przeprojektowanie modelu danych). 4. ... 		
Koszt estymowany		100 000 PLN	Opóźnienie estymowane	1 m-c
Koszt realny			Opóźnienie realne	

Zestawienie listy ryzyk projektowych w formie skróconej pozwala ocenić całościowo stopień zagrożenia, jakiemu podlega nasz projekt. Przykład takiego zestawienia znajduje się w tabeli 7.2.

Tabela 7.2. Zestawienie ryzyk projektowych

ID	Prawdopodobieństwo	Wpływ na termin	Wpływ na koszt	Uwagi
R1	5%	1 m-c	Brak	
R2	15%	2 m-ce	100 000 zł	Krytyczne!
...

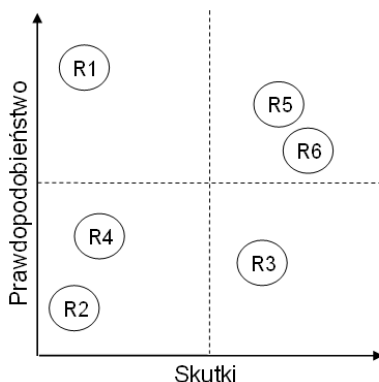
Długość i zawartość zestawienia ryzyk daje dobry pogląd na kondycję projektu. Można zaryzykować stwierdzenie, że projekt bez identyfikowanych ryzyk jest na dobrej drodze do klęski (nie wiadomo tylko, *kiedy i dlaczego coś się „posypie”*) — projekt z rozeznaniem zagrożeń ma znacznie większą szansę na powodzenie!

Mapa ryzyka

Mapa ryzyka jest świetnym narzędziem pokazującym stan projektu w kategoriach listy ryzyk projektowych. Przykład takiej mapy znajdziesz na rysunku 7.4.

Rysunek 7.4.

Mapa ryzyka



Nawet nie dysponując specjalistycznym oprogramowaniem, które pozwoli rysować takie mapy, nie jest trudno dla wybranej liczby ryzyk samodzielnie stworzyć taki diagram. Na pewno jest on znacznie czytelniejszy dla odbiorców (a zwłaszcza kierownictwa) niż surowe dane z tabelki.

Pytania kontrolne

1. Podaj definicję „ryzyka projektowego”.
2. Czy ryzykiem można zarządzać?
3. Jaki jest koszt zarządzania ryzykiem? Przedyskutuj tę kwestię na przykładzie małego i dużego projektu.
4. Który kwadrant (jeden z czterech obszarów wykresu) mapy ryzyka wydaje się szczególnie godny uwagi?

Skorowidz

3C (metoda), 287

A

Actual Cost of Work Performed, 159
ACWP, *Patrz* Actual Cost of Work Performed
affinity diagram, *Patrz* diagram podobieństwa
agenda, *Patrz* spotkania
Agile Modeling, 303
Agile Release Train, 324
Agile Unified Process, 235
AgileEVM, 314
AgilePM, 228
amortyzacja, 145
analiza, 29
analiza budżetowa, 146
arkusz ocen ofert, 211
ART, *Patrz* Agile Release Train
AUP, *Patrz* Agile Unified Process

B

Backlog grooming, 295
baseline, *Patrz* Wersja bazowa harmonogramu
baza wiedzy, 172, 251
BCWP, 159, *Patrz* Budgeted Cost of Work Performed
BCWS, *Patrz* Budgeted Cost of Work Scheduled
Beerware, 216
Belbin Meredith, 65
Billy Wake, *Patrz* INVEST
BPM, 106
Budgeted Cost of Work Performed, 159
Budgeted Cost of Work Scheduled, 159

budżet
 przychody bezpośrednie, 143
 przychody niebezpośrednie, 143
budżet projektu, 53, 143
 elementy, 145
build, *Patrz* wydanie

C

C3 (projekt), 238
CAPEX, 56, 145
CAS, 299, *Patrz* złożone system adaptacyjne
CASE, *Patrz* Computer Aided Software Engineering
certyfikacja project management, 219
CFD, *Patrz* kumulacyjny diagram przepływu
CFM, 255
change management, *Patrz* zarządzanie zmianą
change request, *Patrz* żądanie zmiany
CMM, 186, 189, 236
coaching, 62
Cockburn, Alistair, 236
Computer Aided Software Engineering, 89
continuous integration, *Patrz* systemy ciągłej integracji, *Patrz* systemy ciągłej integracji
Cost Performance Index, 160
Cost Variance, 160
COVID-19, 112
CP, *Patrz* Cost Performance Index
CRUFT, 303
Crystal Clear, 236
CV, *Patrz* Cost Variance
Cykl Deminga, 253
cykl życia projektu, 28

D

DAD, *Patrz* Disciplined Agile Delivery
 De Luca, Jeff, 240
 definicja akceptacji wykonania, 320
 Deming, William Edwards, 236
 demonstracja (produktu), 284
 design, 29
 DevOps, 329
 DFM, 255
 diagram pokrewieństwa, 105
 Disciplined Agile Delivery, 328
 DoD, 294, 328, *Patrz* definicja akceptacji wykonania
 dokumentacja projektowa, 167, 302
 dostawa, 196, 199, 201
 dotproject, 334
 DOWNTIME, 245
 DSDM, 228
 Duarte, Vasco, *Patrz* NoEstimates
 dziennik projektu, 169

E

EAC, *Patrz* Estimated At Completion, *Patrz* Estimated At Completion
 Earned Value, 357, *Patrz* metoda wartości wypracowanej
 Agile, 314
 procedura, 163
 EDUF, 277
 efekt synergii, 55
 effort-driven (planowanie w trybie), 131
 emergencja, 270
 Enron, 111
 Epika, 287, 300, 320, 321
 escrow agreement, *Patrz* zabezpieczenia depozytowe
 Estimated At Completion, 160, 161
 estymacja kosztów, 32, 129
 koszty zarządzania, 32
 Extreme programming, *Patrz* Programowanie ekstremalne

F

FDD, *Patrz* Feature Driven Development
 Feature Driven Development, 240
 Finish-Finish, 126
 Finish-Start, 126
 Fixed Duration, 135

fixed price, *Patrz* kontrakt o stałej wartości
 Fixed Units, 134
 Fixed Work, 136
 flow, 311
 FMCG, *Patrz* produkty szybkozbywalne
 FOSS, *Patrz* Free and Open Source Software, *Patrz* Free and Open Source Software klasyfikacja licencji, 215
 Free and Open Source Software, 207, 214
 FURPS, 95

G

Gantt, 28
 Gantt (diagram), 31
 Gantt (Henry), 28
 GanttProject, 123, 339
 Gemba walking, 246, 327
 Go See (LeSS), 327
 grooming, 182
 gwarancja, 204

H

harmonogram, 123
 kalendarze niestandardowe, 363
 Historyjka Użytkownika, 89, 293, 319, 321

I

IM, *Patrz* wiadomość błyskawiczna implementacja, 29
 interesariusze, 34, 90
 interim plan, *Patrz* plan pośredni
 Internet of Things, *Patrz* Internet Reczy INVEST, 321
 IoT, *Patrz* Internet of Things
 IOT, *Patrz* Interoperability Testing
 ISO, 186, 236
 ISO 9001, 167, 186

J

JAD, *Patrz* Joint Application Design
 jakość, 185
 jednostka pracy, 78
 Jenkins, 196
 Jira, 31
 Joint Application Design, 107

K

Kaizen, 254
kamień milowy, 44, 125
Kanban, 307
karta pracy, 150
karta projektu, 30, 40
karty pracy, 150
Kawakita, Jiro, 105
kick-off meeting, *Patrz* spotkanie rozpoczynające
kierownik projektu, 35
budowanie autorytetu, 43, 63
klasyfikacja dr. Belbina, 65
Dusza zespołu, 66
Koordynator, 65
Krytyk wartościujący, 66
Lokomotywa, 66
Myśliciel, 65
Realizator, 66
Skrupulatny wykonawca, 66
Specjalista, 66
klasyfikacja Myers-Briggs
ENFJ, 69
ENFP, 69
ENTJ, 69
ENTP, 69
ESFJ, 69
ESFP, 69
ESTJ, 69
ESTP, 69
INFJ, 70
INFP, 70
INTJ, 70
INTP, 70
ISFJ, 70
ISFP, 70
ISTJ, 70
ISTP, 70
komitet sterujący, 44, 50
komunikacja elektroniczna, 180
konserwacja, 29, 83
kontrakt o stałej wartości, 210
koronawirus, 112
koszty osobowe, 146
kumulacyjny diagram przepływu, 311

L

Large-scale Scrum, 326
Lean Manufacturing, 243
Lean Software Development, 247

LeSS, *Patrz* Large-scale Scrum
leveling, *Patrz* optymalizacja obciążenia zasobów
linia główna, 196
lista kontrolna, 155
lista produktów, 171
lista ryzyk, 173
Lista zmian, 85

M

macierz odpowiedzialności, 49, 57
maintenance, 29 *Patrz też* konserwacja
maintenance project, 90
Manifest Agile, 271, 299
mapa myśli, 104
mapa polityczna projektu, 93
Mapowanie Historyjek Użytkownika, 321
Mapowanie Strumienia Wartości, 244
Maslow, *Patrz* piramida Maslowa
master branch, *Patrz* linia główna
Maven, 335
Metoda 5 why, 246, 327
metoda ścieżki krytycznej, 28
metoda wartości wypracowanej, 158
Microsoft Project (format), 352
mikrozarządzanie, 44
milestone, 137, *Patrz* kamień milowy
Mind Mapping, *Patrz* mapa myśli (technika)
Minimum Viable Product, 273, 300, 315, 325
minutki, *Patrz* protokoły posiedzeń
model typów osobowości Myers-Briggs, 67
modelowanie wizualne, 235
modyfikacja wsteczna, 87
MoSCoW, 228, 276
motywacja, 60
MUST, 276
MVP, *Patrz* Minimum Vialable Product
Myers-Briggs, 67
typy osobowości w pigułce, 69

N

Nexus, 330
NoEstimates, 304
norma, 234
normy jakości, 34
nota wydania, 172
NPV, *Patrz* wartość bieżąca netto

O

ODM, *Patrz* Original Design Manufacturer
 OEM, *Patrz* Original Equipment Manufacturer
 OODO, 336
 Open Source, 214
 open space, 58
 OPEX, 56, 145
 opłacalność projektu, 144
 optymalizacja obciążenia zasobów, 137
 organizacja, 34
 Original Design Manufacturer, 213
 Original Equipment Manufacturer, 212
 outsourcing, *Patrz* podwykonastwo

P

pakiet serwisowy, 87
 Patton, Jeff, 321
 PC, *Patrz* Percent Completed
 PDCA, 270, *Patrz* Cykl Deminga
 PDU, 221, 224
 Percent Completed, 161
 Percent Planned, 161
 Percent Spent, 161
 PERT, 31, 360
 PERT/CPM, 128
 Peter DeGrace, *Patrz* sashimi
 piramida Maslowa, 62
 plan awaryjny, 114
 plan pośredni, 155
 plan projektu, 30, 45

- budżet, 53
- harmonogram prac, 52
- infrastruktura, 52
- procedury, 50
- szkolenia, 52
- uczestnicy, 48
- zakres prac, 46

 plan testów, 194
 Planning Poker, 294, 304
 planowanie zadań, 123
 PMBOK, 220
 PMI, *Patrz* Project Management Institute
 PMI-ACP, 223
 PMP, 220
 podwykonawstwo, 116, 149, 207
 portfele projektów, 27
 PP, *Patrz* Percent Planned
 praca głęboka, 58, 180
 praca zdalna, 59, 62
 prawa własności intelektualnej, 214

PRINCE2, 224
 PRINCE2 Agile Foundation, 227
 PRINCE2 Agile Practitioner, 228
 PRINCE2 Foundation, 227
 PRINCE2 Practitioner, 227
 proces, 187, 234
 Product Backlog, 30, *Patrz* Rejestr Produktu
 Product Owner, 292
 produkty szybkozbywalne, 86
 program, 27
 Program Increment, *Patrz* SAFE
 programowanie ekstremalne, 238
 project charter, *Patrz* karta projektu
 project management, 34
 Project Management Institute, 219
 ProjectLibre, 123, 349
 projekt, 21

- faza inicjacji, 28
- faza wytwórcza, 28
- inicjacja, 42
- planowanie, 28
- rejestracja stanu realizacji prac, 156
- stan projektu, 154
- symulowanie postępu prac, 165
- typy projektów, 21
- uruchamianie, 39
- zamknięcie, 28

 projektowanie systemu, 29
 protokoły posiedzeń, 175, 179
 prototypowanie, 106
 przebieg, 292
 przegląd postimplementacyjny, 203
 przegląd sprintu, 296
 przypadki testowe, 194
 przypisywanie zasobów zadaniom, 131
 Przyrost Produkt, 300
 PS, *Patrz* Percent Spent

Q

QA (środowisko), 196

R

R&D, 25
 RACI, 48, *Patrz* macierz odpowiedzialności
 Raport o stanie projektu, 167
 raportowanie w Agile, 307
 Rational Unified Process, 234
 RBS, 361
 RCA, 253
 regresja, 196

Rejestr Produktu, 84, 293
 rejestr sprintu, 308
 Rejestr Sprintu, 293
 release, *Patrz* wydanie
 release notes, *Patrz* nota wydania, *Patrz* Lista zmian
 Resource Sheet, 132
 Retrospekcja sprintu, 296
 retrofit, *Patrz* modyfikacja wsteczna
 rezerwa menedżerska, 131
 RFI, *Patrz* zapytanie informacyjne
 RFP, *Patrz* Zapytanie ofertowe
 role zespołowe, 65
 RT: Request Tracker, 336
 RUP, *Patrz* Rational Unified Process
 ryzyko
 definicja, 112
 materializacja, 119
 odkrywanie ryzyk, 115
 rodzaje, 116
 szablon, 120

S

SAFe®, 323
 sashimi, 29, 267
 Schedule Performance Index, 160
 Schedule Variance, 160
 SCM, 255
 Scrum Guide, 296
 Scrum Master, 292
 Service Level Agreement, 204, 212
 service pack, *Patrz* pakiet serwisowy
 SharePoint, *Patrz* Microsoft SharePoint
 SLA, *Patrz* Service Level Agreement
 SPI, *Patrz* Schedule Performance Index
 Spike (Scrum), 293
 spodziewany poziom obsługi, 204
 sponsor, 90
 spotkania
 dokumentowanie, 175
 strefy czasowe, 59
 spotkanie rozpoczynające, 43
 sprint, *Patrz* przebieg
 standardy, 234
 stand-up, 278, 286, 295
 Start-Finish, 127
 Start-Start, 127
 start-up, 234
 statement of work, 149
 stopa dyskonta, 145
 Story Point, 305
 strategia, 21

struktura
 funkcyjna, 23
 nieformalna, 23
 projektowa (idealna), 25
 struktura podziału pracy, 30
 struktury organizacyjne, 23
 Strumień Wartości, 244, 324
 sukces projektu, 33, 298
 SV, *Patrz* Schedule Variance
 systemy ciągłej integracji, 195, 196
 szkolenia użytkowników, 201

S

ścieżka krytyczna, 31, 128, 153, 348, 360
 środowisko, 171
 środowisko upolitycznione, 91

T

Tablica Projektu, 286
 TCPI, *Patrz* To Complete Performance Index
 TDD, 235
 Team Board (DSDM), *Patrz* Tablica Projektu
 team building, 62
 testowanie, 29
 testowanie oprogramowania, 190
 testy
 akceptacyjne, 191, 192
 czarna skrzynka, 193
 funkcjonalne, 191
 integracyjne, 191
 jednostkowe, 191
 niefunkcjonalne, 192
 regresji, 191
 systemowe, 191
 wydajnościowe, 192, 193
 tiger team, 268
 time and material, 210
 time sheet, 150
 timebox, 279
 To Complete Performance Index, 160
 Total Quality Management, 186
 Toyota Production System, 255
 TQM, *Patrz* Total Quality Management
 Trac, 337
 triada projektowa, 31
 Agile, 298
 typologia MTR-i™, 66
 Badacz, 67
 Krzyżowiec, 67
 Kustosz, 67

typologia MTR-i™
 Naukowiec, 67
 Przewodnik, 67
 Rzeźbiarz, 66
 Trener, 67
 Wynalazca, 67

U

udziałowcy, 90
 umiejętności miękkie, 285
 UML, 102, 106, 235
 User Story, *Patrz* Historyjka Użytkownika

V

VAC, *Patrz* Variance At Completion
 Value Stream, *Patrz* Strumień Wartości
 Value Stream Mapping, *Patrz* Mapowanie
 Strumienia Wartości
 Variance At Completion, 161
 velocity, 296, 305

W

wartość bieżąca netto, 145
 waterfall, 29
 WBS, 73, 199, 360, *Patrz* Struktura podziału
 prac
 Agile, 322
 poziomy, 75
 słownik, 75
 tworzenie, 78, 80
 wdrożenie, 29, 196
 Weighted Shortest Job First, 325
 Wekan, 337
 wersja bazowa harmonogramu, 153
 wiadomość błyskawiczna, 180
 wiki, 302

WIP, 258, 307
 wizja, 89
 work package, *Patrz* jednostki pracy
 WSJF, *Patrz* Weighted Shortest Job First
 WTC, 112
 wydanie, 87, 196
 Wydanie Produktu, 300
 wykres spalania, 309
 wymagania użytkownika, 89
 niestabilność, 92
 problemy, 90
 procedury zbierania wymagań, 100
 typologia, 95
 wymaganie, 89

X

XP, *Patrz* Programowanie ekstremalne

Z


zabezpieczenia depozytowe, 214
 zadania projektowe
 typy, 124
 zależności, 125
 zakres, 34
 zakupy, 146
 zamknięcie prac w projekcie, 202
 zamknięcie projektu, 199
 zapytanie informacyjne, 184
 zapytanie ofertowe, 184, 209
 zarządzanie konfliktami, 71
 zarządzanie zmianą, 83, 174
 Zentao, 338
 złożone system adaptacyjne, 269

Z

żądanie zmiany, 83

PROGRAM PARTNERSKI

— GRUPY 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
Helion

POZNAJ NAJSKUTECZNIEJSZE METODY ZARZĄDZANIA PROJEKTAMI

- Dowiedz się, na czym polega project management
- Opanuj zwinne metodyki zarządzania projektami
- Naucz się zarządzać wymaganiami, zmianami, budżetem i ryzykiem
- Sięgnij po najlepsze narzędzia wspomagające pracę PM-a

Zarządzanie projektami to rozległa i skomplikowana dziedzina wiedzy. Dobry kierownik projektu powinien być osobą kompetentną, komunikatywną, zdyscyplinowaną, wszechstronną i doświadczoną, łączącą zdolności techniczne z marketingowymi, a nawet psychologicznymi. Dobry PM to zdolny dyrygent, pod którego kierunkiem zespół deweloperski sprawnie osiąga wyznaczone cele i szybko realizuje powierzone mu zadania — gra bez fałszu, niczym świetnie zestrojona orkiestra.

W ostatnich latach metodyki zwinne przebojem wkroczyły do firm informatycznych. Rewolucyjnie zmieniły sposób wytwarzania oprogramowania, przenosząc akcent z planowania i budżetowania na dostarczanie wartości biznesowej oraz zaspokajanie biznesowych potrzeb klienta. Rewolucja ta sprawiła, że deweloperzy nie są już postrzegani wyłącznie w kategorii kosztów, zespołom oddaje się więcej inicjatywy, a efekty ich pracy można znacznie szybciej skonfrontować z oczekiwaniami.

Nie każdy łatwo odnajduje się w tej nowej rzeczywistości. Pomoże w tym praktyczny poradnik napisany przez doświadczonego kierownika projektów. Krótko, zwięźle i rzeczowo, bez nudnej teorii, krok po kroku wprowadzi Cię we współczesny świat zarządzania projektami i pokaże, jak radzić sobie z codziennymi wyzwaniami. Jeśli chcesz zacząć przygodę z project managementem, odświeżyć swoją wiedzę lub wzbogacić ją o znajomość metodyk Agile, jesteś na właściwym tropie!

- Koncepcje zarządzania projektami
- Właściwe uruchamianie projektu
- Zarządzanie zespołem i komunikacja
- Planowanie i tworzenie harmonogramów
- Zbieranie wymagań użytkownika
- Realizacja i metody kontroli prac
- Zarządzanie ryzykiem i zmianami
- Zarządzanie jakością i budżetem
- Dokumentacja projektowa i raportowanie
- Profesjonalne kończenie projektu
- Zwinne metodyki wytwarzania oprogramowania
- Darmowe oprogramowanie do zarządzania projektami

Zostań takim kierownikiem projektu, z jakim zawsze chciałeś pracować!

Piotr Wróblewski — doświadczony kierownik projektów informatycznych, wcześniej również release manager i product owner, obecnie pracuje w firmie Nokia stosującej od lat metodyki zwinne w działach R&D. Od 1992 roku stały współpracownik wydawnictwa Helion. Autor wielu cenionych książek.

  helion.pl	<i>Sprawdź nasze szkolenia!</i>  AKADEMIA IT & BUSINESS HELIONSZKOLENIA.PL	KOD KORZYŚCI <i>Sięgnij po więcej!</i>   ISBN 978-83-283-6654-1  9 788328 366541
 HELION SA ul. Kosciuszki 1c 44-100 Gliwice tel.: 32 230 98 63 helion@helion.pl		INFORMATYKA W NAJLEPSZYM WYDANIU Cena: 89,00 zł