

Jolanta Pondel

Wyższa Szkoła Handlowa we Wrocławiu

Maciej Pondel

Uniwersytet Ekonomiczny we Wrocławiu

**CZYNNIKI POWODZENIA
PROJEKTU INFORMATYCZNEGO**

Streszczenie: Celem artykułu jest scharakteryzowanie projektu/przedsięwzięcia informatycznego oraz czynników mających wpływ na jego powodzenie (przyczyn utrudniających jego zakończenie). Przybliżono także kolejne etapy tworzenia projektów i strategie ich wdrażania.

Słowa kluczowe: projekt informatyczny, zarządzanie projektem, etapy tworzenia, strategie, przyczyny niepowodzeń projektów.

1. Wstęp

Warunki funkcjonowania współczesnych przedsiębiorstw powodują, że zarządzający nimi właściciele/menedżerowie nieustannie poszukują nowoczesnych form, narzędzi i technologii, które umożliwiają zwiększanie efektywności zarządzania przedsiębiorstwem. Dlatego też najczęstszymi inwestycjami w przedsiębiorstwach w ostatnich latach są inwestycje w nowe rozwiązania informatyczne. Działania te nie tylko związane są z zakupem nowych technologii, ale obejmują również opracowanie wizji nowoczesnego, elastycznego przedsiębiorstwa. Obecnie trudno byłoby znaleźć przedsiębiorstwo, które nie wykorzystuje systemu informatycznego. Konieczność sprawnej i terminowej realizacji złożonych i w dużej mierze niepowtarzalnych przedsięwzięć sprawia, że projekty i podejście projektowe na stałe zagościły w bieżącej aktywności przedsiębiorstw. Wdrożenie nowego rozwiązania informatycznego w przedsiębiorstwie jest najczęściej przedsięwzięciem o niepowtarzalnym charakterze. Tego rodzaju przedsięwzięcia określane są mianem projektu informatycznego.

2. Projekt informatyczny

Międzynarodowa organizacja skupiająca firmy i osoby zainteresowane zarządzaniem projektami Project Management Institute definiuje projekt jako tymczasową

działalność podejmowaną w celu wytworzenia (dostarczenia) unikatowego wyrobu, usługi bądź osiągnięcia unikatowego rezultatu [*Kompendium wiedzy...* 2006]. Z. Szyjewski uważa, że projekt to niepowtarzający się, nierutynowy proces realizacji określonych celów, w określonym czasie i za pomocą określonych środków [Szyjewski 2004].

Projekt bywa też definiowany przez wskazywanie jego poszczególnych cech. H. Kellner twierdzi, iż projekt charakteryzuje się tym, że jest on niepowtarzalny i ograniczony czasowo, ma określone cele, obejmuje różne metody i techniki zarządzania, umożliwia rozwiązywanie nowych i nieznanych wcześniej problemów, jest z nim związane określone ryzyko, ma przypisany odpowiedni budżet, a podczas realizacji prac uczestnicy projektu znajdują się pod szczególną presją [Kellner 2001]. W artykule przyjęto, że projekt (przedsięwzięcie) informatyczny to umiejscowiony w czasie zespół działań podejmowanych w celu stworzenia niepowtarzalnego (o określonych atrybutach) produktu lub usługi przy wykorzystaniu środków informatycznych w ustalonym czasie, zgodnie z ustalonym budżetem.

Niezależnie od specyfiki i wielkości projektu można wyróżnić jego następujące cechy:

- skończony czas trwania,
- jest zorientowany na cel,
- zawiera skoordynowane ze sobą działania,
- kolejne etapy realizacji podlegają stałej kontroli,
- produkt lub usługa końcowa są niepowtarzalne (charakterystyczne atrybuty).

Projekt informatyczny ma na celu wytworzenie produktu, który wspiera system informacyjny przedsiębiorstwa za pomocą technologii informacyjnej. Projekty informatyczne można różnicować przede wszystkim według: celu realizacji (podstawowe, uzupełniające), zakresu przedmiotowego (wycinkowe, tematyczne, problemowe, zintegrowane), sposobu realizacji (własny zespół – wewnętrzne, obcy zespół – zleczone, zespół mieszany), skali (proste, złożone), fazy cyklu życia (definiowanie strategii, analityczne, projektowe, programowe, wdrożeniowe, doskonalące), obszaru (organizatorskie, technologiczne), charakteru zmian (pierwotne, podtrzymujące, modernizacyjne, rozwojowe), rodzaju (infrastrukturalne, usprawniające, innowacyjne – nowego typu) [Dyczkowski 2003].

3. Etapy tworzenia projektu informatycznego

Projekty informatyczne to przedsięwzięcia bardzo złożone – unikatowe, niepowtarzalne, które charakteryzują się określonym cyklem życia. Tradycyjny cykl życia projektu charakteryzuje się następującymi cechami [Westland 2006]:

- stopniowość kolejnych etapów (jeden po drugim; początek i koniec etapu określa najczęściej przekazanie informacji lub produktów technicznych),
- koszty prac oraz stopień zaangażowania poszczególnych pracowników zespołu wzrastają w trakcie poszczególnych etapów i spadają w końcowym etapie całości projektu,

- poziom niepewności oraz ryzyka w początkowych etapach projektu jest najwyższy, a w miarę realizacji projektu prawdopodobieństwo udanego zakończenia wzrasta,
- największy wpływ na końcowy efekt przedsięwzięcia i na ostateczne koszty takiej realizacji inwestor ma na początku przedsięwzięcia. W miarę realizacji projektu wpływ ten maleje.

Cykl życia projektów informatycznych to sekwencja faz/etapów stanowiących jego ścieżkę realizacyjną. Nie można wskazać jednego idealnego modelu cyklu życia. Wiele przedsiębiorstw informatycznych zajmujących się projektami wypracowało własne modele, które są zależne od specyfiki przedsięwzięcia czy zaawansowania inwestora. Określają one, jakie prace należy wykonać na każdym z etapów, czy/kiedy powstają produkty cząstkowe, kto uczestniczy w realizacji danego etapu, sposób kontroli i zatwierdzania etapu.

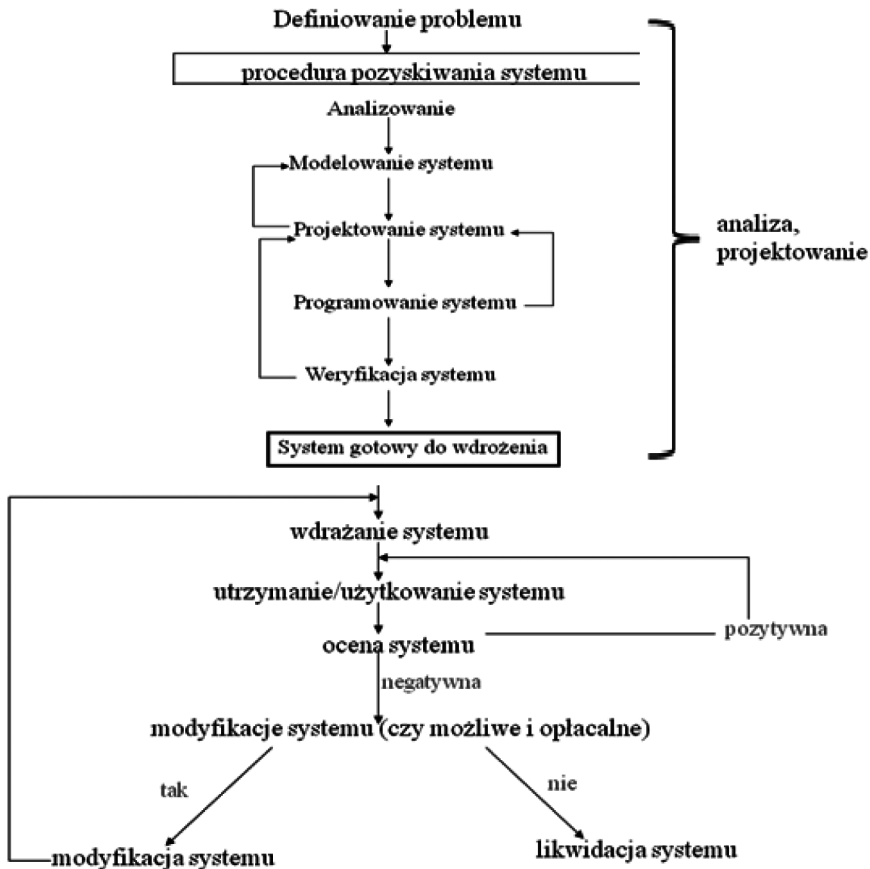
W większości przedsięwzięć informatycznych można wyróżnić następujące etapy (rys. 1):

- analizę,
- projektowanie i implementację (wykonywanie/programowanie),
- wdrażanie,
- utrzymanie/użytkowanie,
- modernizację lub likwidację systemu (obumieranie systemu).

Etap analizy określa: zadanie projektowe (zgodnie z potrzebami użytkownika/inwestora), założenia systemowe (zakres funkcjonalny, rodzaj technologii, zasady współpracy użytkowników z systemem), przewidywane koszty, czas, kadre, przewidywane zmiany w funkcjonowaniu organizacji. Kadre wybiera się zarówno po stronie inwestora projektu (osoby, których zadaniem będzie kontakt z zespołem projektowym – z jego kierownikiem lub analitykiem – wskazywanie, analizowanie prac), jak i po stronie firmy informatycznej odpowiedzialnej za wykonanie projektu informatycznego (zespół projektowy).

Zespół projektowy w zależności od rodzaju i wielkości projektu może stanowić mieszankę pracowników z różnych działów firmy wraz z osobami z zewnątrz organizacji. Przy ustalaniu zespołu (wybieraniu osób) istotne jest wzięcie pod uwagę czynników:

- organizacyjnych – jakie komórki organizacyjne powinny być w zespole, czy uczestniczą one w całym projekcie, czy tylko w wybranych etapach,
- technicznych – określających zakres kwalifikacji i doświadczenia poszczególnych członków zespołu (np. liczbę programistów, analityków, grafików),
- logistycznych – oznaczających swobodny dostęp do wiedzy przydzielonych członków zespołu (np. pracownicy z innych oddziałów przedsiębiorstwa, możliwość dojazdu na spotkania, uprawnienia dostępowe do informacji, baz danych),
- interpersonalnych – polegających na doborze pracowników zapewniających właściwą atmosferę (niekonfliktowych, bez emocjonalnych antagonizmów),



Rys. 1. Etapy tworzenia tradycyjnych projektów informatycznych

Źródło: opracowanie własne.

- politycznych – istotne jest, aby uczestnicy zespołów, a przede wszystkim kierownik, mieli decydujące zdanie w procesach decyzyjnych związanych z realizacją projektu.

Pracownicy wchodzący w skład zespołu projektowego powinni się cechować bogatym doświadczeniem, ale też określonym zestawem cech osobowościowych pozwalających na sprawną organizację i zarządzanie projektem. Przy doborze członka zespołu projektowego powinny być brane pod uwagę trzy główne cechy, które powinien posiadać:

- kompetencje,
- motywacja do pracy,
- atrybuty osobiste.

W skład zespołu projektowego wchodzi:

- kierownik projektu (*Projekt Manager*) – osoba odpowiedzialna zarówno od strony wykonawcy, jak i zamawiającego za prawidłowy przebieg projektu,
- projektanci, analitycy – osoby odpowiedzialne za wykonanie analizy biznesowej i systemowej oraz za opracowanie szczegółowego projektu systemu na podstawie wymagań,
- programiści, inżynierowie – osoby odpowiedzialne za implementację rozwiązania, integrację podsystemów, integrację z systemami zewnętrznymi oraz budowę rozwiązań sprzętowych wymaganych do realizacji zaprojektowanego systemu,
- komisja akceptacyjna – osoby odpowiedzialne za dopuszczenie zrealizowanego systemu informatycznego do pracy produkcyjnej, zarówno od strony klienta, jak i wykonawcy.

Podczas organizowania pracy zespołu projektowego powinno się brać pod uwagę m.in. [Chomiak-Orsa 2009]:

- miejsce pracy zespołu, w tym możliwości komunikacji między członkami zespołów,
- rozwiązania organizacyjne w zakresie oddelegowania pracowników do zadań projektowych, naboru nowych pracowników (procedury rekrutacji, kwalifikacje),
- koszty pracy poszczególnych członków (stałych, nowych, ekspertów spoza firmy),
- termin rozpoczęcia i zakończenia prac projektowych,
- opracowanie planów alternatywnych w przypadku zaistnienia odstępstw od zaplanowanych działań (działania korygujące).

Określenie powyższych zagadnień, odpowiedni dobór uczestników przedsięwzięcia, nadanie im właściwych obowiązków, właściwy system motywacji członków zespołu to elementy, które odgrywają znaczną rolę w odniesieniu sukcesu/niepowodzeniu projektu.

Etap modelowania przedsięwzięcia obejmuje czynności, których wynikiem powinno być opracowanie modelu przyszłego systemu informatycznego. Do prac tych można zaliczyć:

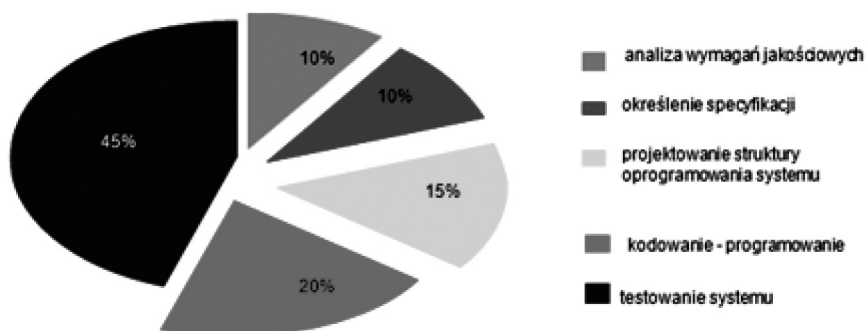
- identyfikację istniejącego systemu (jeśli istnieje w przedsiębiorstwie system, należy zbadać dokumentację, przeprowadzić rozmowy z jego użytkownikami itp.),
- analizę wyników (należy wskazać wady i zalety istniejącego systemu),
- proponowanie rozwiązania problemów i przedstawienie modelu nowego systemu.

Etap projektowania systemu to opracowanie szczegółowego projektu systemu, za pomocą którego inwestor zweryfikuje jego funkcjonalność, a programiści z wyznaczonego zespołu projektowego będą go mogli stworzyć. Etap programowania obejmuje prace programistyczne oraz konfigurację systemu.

Etap weryfikacji przedsięwzięcia informatycznego obejmuje prace polegające na:

- testowaniu (szukaniu błędów),
- sprawdzaniu funkcjonalnym (możliwość realizacji funkcji systemowych oraz ich opcji, poprawność obliczeń (realizacji algorytmów obliczeniowych),
- walidacji zgodności z projektem,
- testach integracyjnych,
- testach akceptacyjnych (zgodność z wymaganiami użytkownika).

Jak wskazują wyniki badań [Pondel, Pondel 2010], najwięcej czasu przy tworzeniu projektów informatycznych zajmuje etap weryfikacji systemu (rys. 2) – ok. 45% czasu przeznaczanego na cały projekt. Analiza wymagań jakościowych i określenie specyfikacji są działaniami podobnymi czasowo i zajmują, wraz z projektowaniem struktury oprogramowania systemu, ok. 35% czasu. Prace polegające na programowaniu/kodowaniu zajmują ok. 20% czasu całego projektu. Wyniki uogólniono na tworzone projekty. W zależności jednak od zaawansowania, doświadczenia firmy informatycznej (twórcy projektu), inwestora, a także wielkości i specyfiki projektu szacunki te mogą od siebie nieznacznie odbiegać (zob. [Pondel, Pondel 2010]).



Rys. 2. Szacunkowy czas tworzenia projektów informatycznych

Źródło: opracowanie własne.

Etap wdrażania można przeprowadzić według trzech poniższych strategii [Mingus 2009]:

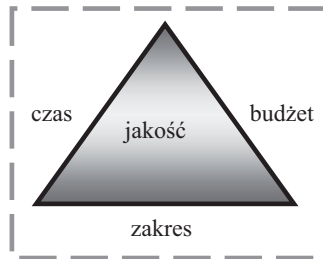
- wdrażanie totalne (całościowe),
- wdrażanie cząstkowe,
- wdrażanie równoległe.

Strategia wdrażania totalnego polega na natychmiastowym wprowadzeniu całości projektu do eksploatacji przy równoczesnej rezygnacji z istniejącego już systemu. Wdrażanie cząstkowe zalecane jest przy wprowadzaniu złożonych i skomplikowanych systemów informacji. Polega na podzieleniu na części (struktury) całego systemu ze względu na określone kryteria, np. podział przestrzenny, na moduły funkcjonalne, i wprowadzaniu poszczególnych części do przedsiębiorstwa.

Strategia wdrażania równoległego polega na przetwarzaniu tych samych danych za pomocą obu systemów, tzn. starego i przygotowanego do wdrażania. Przy wykorzystywaniu systemu przez firmę następują jego aktualizacje (modernizacje) do czasu, kiedy jest to możliwe, opłacalne dla przedsiębiorstwa. Po tym czasie należy wprowadzić na jego miejsce nowy system (następuje likwidacja systemu).

4. Ograniczenia dla projektu informatycznego

Każdy z projektów informatycznych podlega czterem podstawowym ograniczeniom nazywanym trójkątem ograniczeń (rys. 3). Te ograniczenia to: czas, budżet, zakres i jakość. Każdemu z elementów należy nadać jeden i tylko jeden z atrybutów: ustalony, wybrany, uzgodniony [Koszłajda 2010].



Rys. 3. Ograniczenia dla projektu

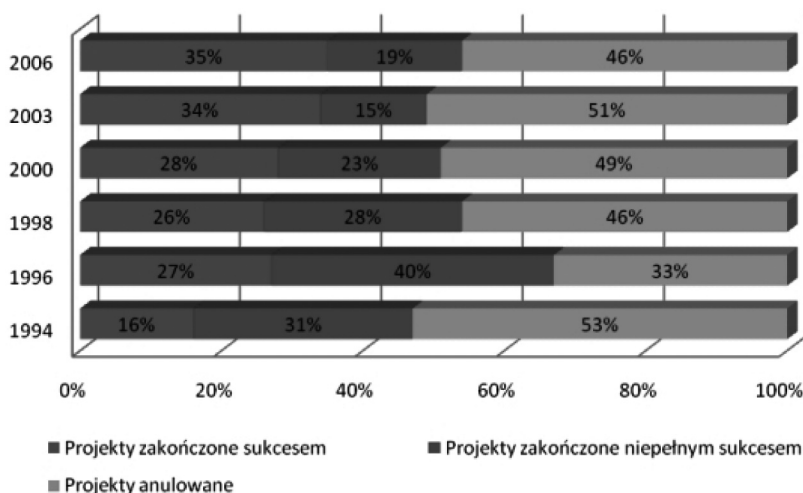
Źródło: [Pondel 2007].

Właściwe zarządzanie projektem polega na bieżącej analizie, planowaniu i dopasowywaniu rozwiązań tak, by wymienione ograniczenia były w równowadze. Jak można zauważyć, występują między nimi silne zależności – chcąc zmodyfikować jedno z ograniczeń, należy zmodyfikować przynajmniej jeszcze jedno, by została zachowana równowaga trójkąta (np. w ustalonym czasie i uzgodnionym budżecie nie można zwiększyć wybranego wcześniej zakresu projektu, nie tracąc przy tym na jakości produktu końcowego).

Dane dotyczące sukcesów/niepowodzeń w projektach IT zostały opublikowane po raz pierwszy przez The Standish Group w roku 1995 [Frączkowski 2003]. Wyniki przedstawiono na rys. 4. Projekty podzielono na trzy grupy:

- projekty zakończone sukcesem – projekty, które zostały zrealizowane w zaplanowanym budżecie, czasie oraz w pełnym zakresie,
- projekty zakończone niepełnym sukcesem – projekty, w których jeden lub więcej z podstawowych parametrów został przekroczony,
- projekty zakończone niepowodzeniem – projekty, które zostały przerwane jeszcze w trakcie ich trwania.

Celem badań było wskazanie przyczyn niepowodzeń projektów oraz czynników wpływających na ich sukces. Jako najważniejsze czynniki sukcesu projektów wskazano: zaangażowanie klienta, wsparcie kierownictwa i jasno określone wymagania.



Rys. 4. Rezultaty projektów

Źródło: [Frączkowski 2003].

Jak wskazują wyniki badań, ponad połowa projektów IT, pomimo dużych nakładów pieniężnych, kończy się niepowodzeniem wynikającym z przekroczenia jednego z czterech podstawowych parametrów projektu, tj.: czasu, budżetu, zakresu lub jakości. Z analizy badanych okresów wynika, że liczba projektów zakończonych sukcesem stale rośnie, jednak w 2006 r. nadal mniej niż połowa z ogólnej liczby projektów kończyła się pełnym sukcesem.

Przygotowany raport wskazał dziesięć głównych przyczyn porażek [Frączkowski 2003]:

- 1) brak informacji kluczowych od klienta/sponsora projektu,
- 2) brak kompleksowego zdefiniowania wymagań i funkcjonalności projektu,
- 3) zmiana wymagań i funkcjonalności w czasie trwania projektu,
- 4) brak wsparcia ze strony kierownictwa projektu,
- 5) brak wystarczającej wiedzy i doświadczenia w danej dziedzinie,
- 6) niekompletny zespół,
- 7) zbyt wygórowane oczekiwania klienta,
- 8) zbyt słabo zdefiniowane cele,
- 9) trudne do spełnienia ramy czasowe projektu,
- 10) nowe technologie.

W.B. Boehm dokonał analizy najczęstszych przyczyn ryzyka projektowego (w odstępnie 6 lat: 1989-1995) [Boehm i in. 2000]. Najczęstsze przyczyny niepowodzeń zawarto w tab. 1.

Jak można zauważyć, w badanym okresie część pozycji uległa zmianie, jednakże nadal są one aktualne [Marasco 2006]. Jak wykazały badania [Pondel, Pondel

Tabela 1. Przyczyny niepowodzeń projektów IT

Lp.	Lata			
	1989	1995	2010	
Przyczyny niepowodzeń	1	Brak wykwalifikowanej kadry specjalistów	Brak wykwalifikowanej kadry specjalistów	Nierealistyczne harmonogramy i budżet
	2	Nierealistyczne harmonogramy i budżet	Nierealistyczne harmonogramy i budżet	Zmiana wymagań i funkcjonalności w czasie trwania projektu
	3	Nieodpowiednia funkcjonalność systemu	Niedostatecznie przygotowane komponenty przez firmy zewnętrzne	Brak kompetencji specjalistów w technologii, w jakiej prowadzi się wdrożenie
	4	Niewłaściwy interfejs użytkownika	Niewłaściwe wymagania inwestora	Niewystarczająca jakość dostarczanych produktów
	5	Ukrywanie problemów projektowych	Niewłaściwy interfejs użytkownika	Błędy metodyczne w zarządzaniu projektem
	6	Ciągłe zmiany wymagań inwestora	Problemy z wydajnością, jakością i architekturą	Problemy z integracją z innymi systemami
	7	Niedostatecznie przygotowane komponenty przez firmy zewnętrzne	Zmiany wymagań przez inwestora	Nieprzewidziane problemy z nowymi technologiami
	8	Nierzetelnie realizowane prace przez podwykonawców	Powiązania z istniejącymi systemami	Błędy metodyczne w wytwarzaniu oprogramowania (brak odpowiednich testów funkcjonalnych, integracyjnych, wydajnościowych; nieodpowiednie decyzje architektoniczne; brak komunikacji między członkami zespołu projektowego)
	9	Problemy z wydajnością	Nierzetelnie realizowane prace przez podwykonawców	Problemy wydajnościowe
	10	Rozwój informatyki	Rozwój informatyki	Niewystarczające zaangażowanie odbiorcy w prace projektowe

Źródło: opracowanie własne.

2010], podczas identyfikacji czynników ryzyka firmy wykorzystują doświadczenia wynikające z prowadzenia własnych projektów. Obejmują one m.in. szacowanie niepowodzenia w zakresie:

- harmonogramów – najczęściej wymienianym niepowodzeniem wynikającym z harmonogramów było wywieraniem presji na skracanie harmonogramów (np. przez inwestorów, którzy zwykli oczekiwać gotowego rozwiązania „na wczoraj”);

- kosztów, których minimalizowanie jest celem zarówno inwestora (chce zapłacić jak najmniej za dostarczenie projektu), jak i twórcy (dla którego koszt pracowników, sprzętu czy oprogramowania jest czynnikiem, który umniejsza zysk z przedsięwzięcia);
- wymagań i oczekiwań użytkowników (często największe wyzwanie dla kierownika projektu). Zdarza się, że najlepiej opisane wymagania są niepoprawne (niezgodne z rzeczywistymi oczekiwaniami), niekompletne (nie uwzględniają wszystkich rzeczywistych potrzeb), niejasne lub niespójne (różni użytkownicy opisują różnie te same wymagania lub różne wymagania podobnie), zmienne (rosnąca świadomość użytkowników powoduje zmiany wcześniejszych wymagań);
- rozbieżności między precyzowaniem wymagań na etapie wyceny systemu (przed podpisaniem umowy z dostawcą) a wymaganiami zgłaszanymi podczas etapu analizy (po podpisaniu umowy). Na etapie wyceny zamawiający ma tendencję do stosowania uproszczeń (dla minimalizacji wartości projektu, która zostanie zapisana w umowie). Po podpisaniu umowy bardzo często wymagania się zwiększają, ponieważ zamawiającemu zależy, aby budowany system zawierał dużo funkcjonalności najlepiej dopasowanych do organizacji. Po przeprowadzeniu analizy powinna nastąpić weryfikacja budżetu i harmonogramu projektu w kontekście uszczegółowionych wymagań. Bardzo często wycena dokonana na etapie analizy w znaczny sposób przekracza zadeklarowaną w umowie. Może to spowodować problemy natury formalnej (konieczność anektowania umów i pozyskania dodatkowego finansowania), biznesowej (niemożność dotrzymania terminów wymaganych do powodzenia projektu) czy wręcz może doprowadzić do rezygnacji z wdrożenia zaplanowanego systemu;
- jakości rozumianej jako poziom akceptacji produktu przez inwestora lub twórcę. Obniżenie jakości projektu może spowodować brak akceptacji inwestora, co może doprowadzić do odrzucenia projektu. W interesie twórcy nie jest obniżanie jakości produktu, ponieważ inwestor nie będzie zadowolony (nie będzie chciał w przyszłości współpracować, polecać firmy innym), a produkt nie będzie się nadawał do dalszego rozwoju;
- działalności operacyjnej, czyli czynników ryzyka wynikających bezpośrednio z konsekwencji działań projektowych lub realizacji projektu (jak np. czynniki losowe, zwolnienia pracowników);
- „typowych błędów”, wynikających z błędnych decyzji podejmowanych przez zespół projektowy, jak np. zwiększanie zakresu projektu przy utrzymaniu ustalonych terminów realizacji, niewłaściwy dobór członków zespołu projektowego (o kompetencjach niewystarczających do realizacji wyznaczonych zadań);
- braku lub też niewystarczającego zaangażowania inwestora w prace projektowe (utrudnienie/uniemożliwienie realizacji wyznaczonych zadań przez zespół projektowy);

- niedostatecznego przygotowania organizacji do używania nowego systemu. Podczas trwania projektu zazwyczaj wszystkie siły i środki są zaangażowane w przygotowanie możliwie najlepszego produktu. Często zapomina się o właściwej komunikacji z użytkownikami nowego systemu oraz określeniu kompetencji i odpowiedzialności kluczowych użytkowników w stosunku do nowego systemu.

Istnieją różne metodyki zarządzania projektami, np. Prince2, PMBoK, Scrum, ale jak wykazały badania, pomimo przyjętych standardów zarządzania projektami, znaczna liczba przedsiębiorstw informatycznych ma problemy ze skutecznym prowadzeniem swoich projektów. Wynika to z tego, że bardzo często z powodów budżetowych, harmonogramowych lub innych rezygnuje się z istotnych części wymaganych przez metodykę procesów, czyniąc zarządzanie projektem bardzo nieefektywnym.

5. Koszty napraw

Błędy popełnione podczas realizacji projektu są możliwe do poprawy. Im szybciej określony błąd zostanie zidentyfikowany, tym mniejszym kosztem będzie można go usunąć. Na rys. 5 zaprezentowano koszty usunięcia błędu występujące w poszczególnych fazach projektu. Nawet niewielkie błędy popełnione w fazie specyfikacji wymagań lub projektowania, które nie zostaną odpowiednio szybko zidentyfikowane, mogą determinować dalsze koszty, aż do fazy wdrożenia systemu, jego utrzymania i konserwacji, a przede wszystkim mogą spowodować, że cały projekt zakończy się niepowodzeniem.



Rys. 5. Koszty naprawy błędów w poszczególnych fazach projektu

Źródło: [Leffingwell, Widrig 2003].

Należy mieć świadomość, że próba wyeliminowania wszystkich błędów na etapie specyfikacji wymagań i projektowania może doprowadzić do znacznego wydłu-

żenia harmonogramu tych faz (a w konsekwencji całego projektu). Produktem takiej fazy może być specyfikacja o dużej objętości oraz bardzo dużej liczbie szczegółów. Taka specyfikacja jest bardzo trudna do zrozumienia, a co za tym idzie – do świadomego zaakceptowania przez inwestora. Aby uniknąć takich sytuacji, często używa się tzw. lekkich metodyk wytwarzania oprogramowania, które są alternatywą dla klasycznego podejścia opartego na modelu kaskadowym. Metodyki te, do których należą programowanie zwinne (*agile software development*), SScrum czy programowanie ekstremalne, zakładają iteracyjne podejście do tworzenia kodu i częste inspekcje wymagań oraz dostosowywanie specyfikacji i oprogramowania do zmieniających się wymagań. Nie można jednak autorytatywnie stwierdzić, że lekkie metodyki są lepszym rozwiązaniem od metodyk klasycznych. Ich słabością jest m.in. brak dokładnej specyfikacji, co utrudnia utrzymanie systemu i jego rozwój po pewnym czasie. Zarządzanie budżetem, który nie jest do końca sprecyzowany, również może budzić sprzeciw wielu inwestorów projektu. Wybór podejścia do prowadzenia projektu jest uzależniony od specyfiki systemu, jaki ma być stworzony, jego celów i założeń biznesowych, a także od technologii, w jakiej projekt jest/ma być realizowany.

6. Zakończenie

Mimo istnienia wielu standardów zarządzania projektami, metodyk, według których prowadzi się projekty, wiele z nich kończy się niepowodzeniem lub niepełnym sukcesem. Nie ma uniwersalnej metodyki dobrej dla wszystkich projektów, gwarantującej sukces. Kierownik projektu powinien umieć zauważyć, odpowiednio zanalizować i skutecznie reagować na pojawiające się błędy/nieprawidłowości podczas realizacji przedsięwzięcia. Zadania powinny być ustalone/organizowane tak, by możliwa była szybka reakcja na zmiany, przy zminimalizowanym ryzyku niepowodzenia przedsięwzięcia. Mimo że analiza przyczyn porażek przedsięwzięć została w artykule przeprowadzona głównie dla branży IT, są to jednak czynniki uniwersalne, niezależnie od obszaru gospodarki, w którym występują. Rozważania na ten temat będą kontynuowane przez autorów.

Literatura

- Boehm W.B. i in., *Software Cost Estimation with COCOMO II*, Prentice-Hall, July 2000.
- Chomiak-Orsa I., *Tworzenie zespołów projektowych i kierowanie nimi jako jeden z czynników decydujących o sukcesie przedsięwzięcia informatycznego*, [w:] A. Nowicki, I. Chomiak-Orsa, *Informatyka ekonomiczna. Rynek usług informatycznych*, Prace Naukowe Uniwersytetu Ekonomicznego we Wrocławiu nr 82, UE, Wrocław 2009.
- Dyczkowski M., *Istota i organizacja przedsięwzięć informatycznych*, [w:] E. Niedzielska (red.), *Informatyka ekonomiczna*, Prace Naukowe Akademii Ekonomicznej we Wrocławiu, AE, Wrocław 2003.

- Frączkowski K., *Zarządzanie projektem informatycznym. Projekty w środowisku wirtualnym. Czynniki sukcesu i niepowodzeń projektów*, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 2003.
- Kellner H., *Die Kunst, IT-Projekte zum Erfolg zu führen. Ziele-Strategien-Teamleistungen*, Hanser, Wien 2001.
- Kompendium wiedzy o zarządzaniu projektami*, PMBOK® Guide 2000 Edition, MT&DC, Warszawa 2006.
- Koszłajda A., *Zarządzanie projektami IT. Przewodnik po metodach*, Helion, Gliwice 2010.
- Leffingwell D., Widrig D., *Inżynieria oprogramowania, Zarządzanie wymaganiami*, Wydawnictwa Naukowo-Techniczne, Warszawa 2003.
- Marasco J., *Zarządzanie projektami informatycznymi*, Helion, Gliwice 2006.
- Mingus N., *Zarządzanie projektami*, Helion, Gliwice 2009.
- Pondel J., Pondel M., *Badania przeprowadzone na grupie 50 kierowników projektów branży IT*, Wrocław 2010.
- Szyjewski Z., *Metodyki zarządzanie projektami informatycznymi*, Agencja Wydawnicza Placet, Warszawa 2004.
- Westland J., *Project Management Lifecycle*, Kogan Page, London 2006.

FACTORS OF SUCCESS IN IT PROJECTS

Summary: For the efficient information processing a modern company needs modern and effective IT systems. Such systems are implemented in a company by the IT project. The main goal of this paper is to characterize IT project and factors determining its success (as well as the reasons of its failure). This paper also presents the order of project's phases and strategies and methodologies of systems implementation.

Key words: IT project, project management, formation phases, strategies, reasons of projects failure.