
Sebastian Wilczewski | sebastian@people.pl

NR ORCID: 0000-0001-5929-7652

Politechnika Gdańska, Wydział Zarządzania i Ekonomii, Katedra Zastosowań Informatyki w Zarządzaniu

Analiza praktyk związanych z zarządzaniem projektami informatycznymi i utrzymaniem usług IT i ich wpływu na realne sukcesy przedsięwzięć. Identyfikacja luk i rekomendacje dotyczące usprawnień

Analysis of Practices Related to the Management of IT Projects and Maintenance of IT Services and their Impact on the Real Success of Projects. Gap Identification and Improvement Recommendations

Abstract

In this article analyzes of the practices related to the management of IT projects and the maintenance of IT services and their impact on the real success of projects were done. Next, the gaps between project management and management of IT services were identified. A directional improvement proposal for eliminating this gaps was done. These activities will increase the probability of real success of the IT projects. The definition of project success has been updated in this article.

Key words:

project management, IT service management, IT service maintenance, operational stability, non-functional requirements

Wstęp

Cel

Celem niniejszego artykułu jest dokonanie analizy praktyk związanych z zarządzaniem projektami informatycznymi, utrzymaniem usług IT i ich wpływu na realne sukcesy przedsięwzięć, a następnie identyfikacja luk występujących pomiędzy tymi obszarami oraz rekomendacje usprawnień eliminujących te luki.

Metoda

W ramach badań dokonano analizy literatury oraz standardów branżowych w zakresie zarządzania projektami i usługami IT. Przeprowadzono również analizę przypadków oraz syntezy doświadczeń w firmach realizujących projekty informatyczne w sektorze bankowym i ubezpieczeniowym. Na tej podstawie zidentyfikowano luki występujące na styku metodyk zarządzania projektami i metodyk związanych z zarządzaniem usługami IT wpływające na to czy projekt zakończy się sukcesem, czy porażką oraz opracowano metody ich niwelowania.

Wyniki i oryginalna wartość

Wynikiem niniejszej pracy jest zaproponowanie koncepcji działań, dopełniających działania podejmowane w ramach działań projektowych (realizowanych zarówno zgodnie z metodykami klasycznymi, jak i zwinnymi) oraz tych podejmowanych w zakresie utrzymania usług IT (zgodnych z metodyką ITIL, COBIT i innymi standardami). Celem tych działań jest zwiększenia prawdopodobieństwa realizacji projektów tak, aby kończyły się one realnym sukcesem.

Wprowadzenie

Efektom projektu jest dostarczenie produktów (rozumianych jako dobro, usługa bądź rezultat) według potrzeb klienta. Według Manifesto, C.H.A.O.S. [2013] 39% projektów IT kończy się sukcesem. Jest on tu rozumiany jako dostarczenie zdefiniowanej funkcjonalności zgodnie z ograniczeniami określonymi przez wszystkie wierzchołki tzw. klasycznego trójkąta projektowego, czyli czasu, budżetu, zakresu. W ocenie autora niniejszego artykułu podana powyżej charakterystyka sukcesu nie jest pełna i opisuje ona jedynie pozory sukcesu. Aby zrozumieć to zastrzeżenie należy zważyć, że AXELOS [2016] w metodyce ITIL wprowadza pojęcie użyteczności (*utility*) i gwarancji (*warranty*). Użyteczność rozumiana jest w ten sposób, że produkt lub usługa spełnia określone wymagania, czyli działa zgodnie z przeznaczeniem (np. system informatyczny zlicza głosy oddane w wyborach samorządowych zgodnie z określonym algorytmem). W tej samej publikacji wprowadzono również pojęcie gwarancji – poręcze-

nia, że usługa ma zdolność do użytku i spełnia ustalone wymagania (np. system ma wydajność pozwalającą na przyjęcie w ciągu 10 minut protokołów z co najmniej 5% obwodowych komisji wyborczych). Innymi słowy użyteczność dotyczy tego, co wdrażane rozwiązanie ma robić, natomiast gwarancja – jak dobrze system ma wykonywać swoje funkcje. Zespół właściwości związanych z użytecznością to wymagania niefunkcjonalne, jakie powinny zostać zdefiniowane wobec wdrażanego rozwiązania.

W definicji sukcesu projektu podanej przez Manifesto, C.H.A.O.S. [2013] brakuje właśnie aspektów związanych ze zdolnością systemu do użytku. Jak brzemienne ma to skutki można się było przekonać w Polsce podczas wyborów samorządowych w roku 2014. W przypadku systemu stworzonego przez firmę Nabino kluczowe niedoskonałości dotyczyły niedostosowanie jego wydajności do ilości użytkowników i wzorców ich działań, a więc nie zostały uwzględnione aspekty związane z gwarancją w rozumieniu podanym przez AXELOS [2016]. Zgodnie z definicją podaną przez Manifesto, C.H.A.O.S. [2013] projekt powinien zostać przypisany (z pewnymi zastrzeżeniami) do kategorii tych, które odniosły sukces. System był gotowy na termin wyborów samorządowych, zamawiający miał zapłacić za niego cenę nie wyższą niż ustalona, a system (co do zasady) pełnił funkcję, do której został stworzony, czyli zliczał głosy w wyborach samorządowych, według zdefiniowanego i zatwierzonego algorytmu. Wszystkie przesłanki sukcesu zdefiniowane w Manifesto, C.H.A.O.S. [2013] były więc spełnione. Całkiem zasadne jest jednak stwierdzenie, że w rzeczywistości wdrożenie systemu nie było sukcesem. Głównym i bardzo istotnym mankamentem był niespełnienie wymagań związanych gwarancją w rozumieniu metodyk ITIL, w szczególności wymagań związanych z wydajnością. Dlatego też według autora niniejszego opracowanie definicja sukcesu projektu powinna zostać zaktualizowana, aby uwzględniała wspomniane powyżej aspekty [por. Horobet, Miłosz 2017, s. 4].

Uwzględniając powyższe rozważania za sukces projekt należy uznać dostarczenie zdefiniowanej funkcjonalności i spełnienie wymagań niefunkcyjnych oraz innych wymagań zgodnie z ograniczeniami określonymi przez wszystkie wierzchołki tzw. klasycznego trójkąta projektowego (czas, budżet, zakres). Nie bez znaczenia są tutaj obserwacje Badewi [2016], który zwraca uwagę na istotność zarządzania korzyściami w kontekście osiągnięcia sukcesu projektu – korzyścią związaną właśnie z wymaganiami funkcjonalnymi i niefunkcyjnymi.

Podany wcześniej przykład dotyczący pozornego sukcesu projektu nie jest odosobniony – zauważalna część projektów kończy się pozornym sukcesem. Przyczyny takiej sytuacji zostaną zidentyfikowane w dalszej części tego artykułu. Zostaną również zaproponowane działania usprawniające.

Definicje kluczowych pojęć

Przed przejściem do kolejnych analiz konieczne jest zdefiniowanie pojęć używanych w tym artykule. Część z nich została już stosunkowo dobrze zdefiniowana w literaturze, pozostałe wymagają jednak doprecyzowania tak, aby ich rozumienie było ujednoczone. Definicje kluczowych pojęć zawartych w niniejszym artykule zostały podane w tabeli numer 1.

Tabela 1. Definicja kluczowych pojęć

Pojęcie	Definicja
Gwarancja (warranty)	Według AXELOS [2016] to poręczenie, że usługa ma zdolność do użytku i spełnia ustalone wymagania. Z pojęciem gwarancji ściśle związane jest pojęcie wymagań нефункциональных.
Projekt	Tymczasowe dążenie do stworzenia produktu, usługi lub rezultatu. Tymczasowość oznacza, że zdefiniowany jest jego początek i koniec. Koniec projektu następuje, gdy zostaną osiągnięte jego cele lub gdy zostanie on rozwiązany z uwagi na nieosiągnięcie celu, lub brak możliwości osiągnięcia celu, lub gdy jego uzasadnienie biznesowe już nie istnieje.
Produkt	Dobro, usługa bądź rezultat wytworzone według potrzeb klienta [Project Management Institute 2004].
Usługa	Sposób dostarczania wartości klientom (zewnętrznym lub wewnętrznym) poprzez ułatwienie im uzyskania rezultatu, którego oczekują bez wzięcia na siebie kosztów i ryzyka [AXELOS 2016].
Usługa IT	Usługa dostarczana przez dostawcę usług IT. Składają się na nią technologie informacyjne, ludzie i procesy. Poziom świadczonych usług musi być zdefiniowany w umowie o poziomie świadczonych usług (Service Level Agreement – SLA) [AXELOS 2016].
Użyteczność (utility)	Produkt lub usługa spełnia określone wymagania, czyli działa zgodnie z przeznaczeniem [AXELOS 2016].
Wymagania нефункциональные	Determinują jak dobrze oprogramowanie ma spełniać swoje funkcje [Paech, Kerkow 2004].
Wymagania funkcjonalne	Determinują co oprogramowanie ma robić [Paech, Kerkow 2004].

Źródło: opracowanie własne na podstawie źródeł podanych w tabeli.



Analiza praktyk związanych z zarządzaniem projektami informatycznymi i utrzymaniem usług IT

Usługa IT dostarczana jest zazwyczaj poprzez realizację projektu, mającego na celu jej wdrożenie, a następnie przekazywana jest ona do utrzymania. W tym rozdziale przeanalizowane zostaną zarówno dobre praktyki prowadzenia projektów, jak i utrzymania systemów IT oraz występujące między nimi zależności i luki.

Praktyki prowadzenie projektów są już stosunkowo dobrze zdefiniowane, a metodyki zarządzania projektami (zarówno klasycznymi, jak i zwinnymi) głęboko zakorzenione w sposobach działania widocznej części organizacji gospodarczych. Podobna sytuacja ma miejsce w przypadku zarządzania usługami IT – istnieją dojrzałe, powszechnie akceptowane metodyki takie jak ITIL czy COBIT definiujące właściwe praktyki w tym zakresie. Świat projektowy jest wzajemnie zależny od świata utrzymania usług IT.

Na potrzeby niniejszego opracowania dokonano analizy dobrych praktyk zarządzania projektami opisanych przez Project Management Institute oraz zasad zarządzania usługami IT opisanymi w cyklu publikacji Information Technology Infrastructure Library (ITIL), aby ustalić jak zdefiniowane są punkty styku między nimi oraz jak zdefiniowane zostały w tym miejscu odpowiedzialności i praktyki.

Analiza praktyk dotyczących zarządzania projektami

Analizując informacje podane przez Project Management Institute [2013] łatwo zauważyć, iż istotą projektu jest dostarczenie produktów projektu w określonym czasie, według określonych kosztów, realizując podany zakres, dochowując parametrów jakościowych, z wykorzystaniem dedykowanych dla projektu zasobów, świadomie zarządzając ryzykami projektowymi. Project Management Institute [2013], skupia się na zdefiniowaniu wejść, wyjść oraz technik i narzędzi dla poszczególnych procesów. Zdefiniowane metodyki mają charakter wytycznych i rekomendacji, jednakże zasady działań nie są w nich szczegółowo opisane.

Metodyki zarządzania projektami skupiają się właśnie na aktywnościach związanych ściśle z projektem, a nie na utrzymaniu produktu wytworzonego podczas pracy nad projektem. Aktywności projektowe mogą być przypisane do jednej z czterech faz realizacji przedsięwzięcia: inicjowania, organizowania i przygotowywania, realizacji oraz zamykania. Granice wydzielające projekt (zarówno jego początek, jak i koniec) są jasno określone. Niezależnie od zastosowanej metodyki zarządzania projektami, każde przedsięwzięcie kończy się etapem zamknięcia. Po nim produkty finalne (w szczególności usługi IT) dostarczane są do utrzymania i rozpoczyna się



okres ich utrzymania. Project Management Institute [2013] wskazał cztery punkty styku świata projektowego z utrzymaniem, w tym utrzymaniem usług IT. Są to:

- moment zamykania projektu lub fazy projektu,
- moment tworzenia nowego produktu lub modyfikacji już istniejącego,
- usprawnianie procesu utrzymania lub wytwarzania produktu,
- moment wycofywania produktu.

Wskazane punkty styku istnieją w fazie realizacji i zamykania projektu. Jest to zdecydowanie zbyt późno, aby móc bez wpływu na czas trwania i budżet projektu określić wymagania dotyczące przekazania rozwiązania do utrzymania. Potencjalnie ujawnione na tym etapie dodatkowe wymagania (pochodzące od działu utrzymania) będą wywierały wpływ na zakres i pozostałe parametry projektu. To z kolei spowoduje, że konieczna będzie modyfikacja projektu tak, aby spełnić wymagania osób odpowiedzialnych za utrzymaniu usług IT. Efektem będzie uruchomienie procedury wdrażania kontrolowanej zmiany, co w praktyce oznaczać może opóźnienia i wzrost kosztów realizacji przedsięwzięcia. Aby tego uniknąć konieczne jest uwzględnienie aspektów związanych z utrzymaniem usług już na etapie inicjowania, organizowania i przygotowywania projektu.

Analiza praktyk dotyczących utrzymania systemów IT

Na powyższe zagadnienie należy również spojrzeć z perspektywy procedur dotyczących utrzymania usług IT. Według AXELOS [2016] cykl życia usługi, składa się z pięciu etapów (strategii, projektowania, tranzycji, utrzymania i ciągłego doskonalenia) i określa, gdzie aspekty związane z utrzymaniem usług się rozpoczynają, a gdzie kończą. AXELOS [2016] dostrzega konieczność interakcji pomiędzy projektem a utrzymaniem. Procesy dotyczące tej interakcji zebrane są w grupie procesów nazwanej tranzycją usługi (*service transition*), do której należą w szczególności następujące procesy:

- planowanie tranzycji usług (*transition planning and support*),
- zarządzanie zmianą (*change management*),
- zarządzanie usługami i konfiguracją (*service asset and configuration management*),
- zarządzanie wydaniem i ich wdrażaniem (*release and deployment management*),
- zarządzanie usługami i ich testowaniem (*service validation and testing*),
- ocena zmiany (*change evaluation*),
- zarządzanie wiedzą (*knowledge management*).

Każdy z tych procesów rozpoczyna interakcję z projektem w stosunkowo późnej części cyklu życia projektu. Tym samym wymagania zdefiniowane wobec projektu będą miały istotny wpływ na wcześniej zdefiniowany zakres, budżet i czas projektu, co



może spowodować, że projekt nie zakończy się na czas lub zgodnie ze zdefiniowanym budżetem, a więc nie będzie można go uznać za taki, który zakończył się z sukcesem.

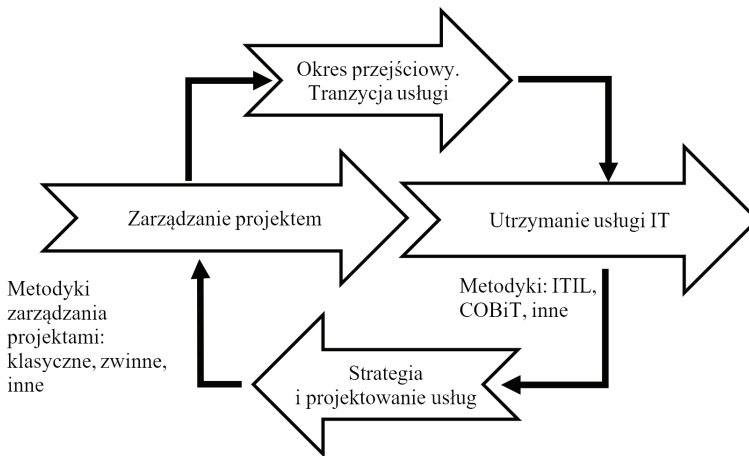
Analiza wpływu sposobu prowadzenia projektów IT na przekazywania produktów wdrożenia do utrzymania i eksploatacji

Zarówno ITIL, jak i PMBOK są zbiorami metodyk i dobrych praktyk. Każdy z nich ogranicza się do przedstawienia ogólnych wytycznych, bez konkretyzowania działań i nie pretendują do tego, aby być zbiorami szczegółowych i obowiązkowych praktyk. Obie metodyki są wzajemnie świadome swojego istnienia i zawierają obustronne referencje. Jest to zjawisko pozytywne, bo dzięki temu kierownicy projektów wiedzą, na jakie aspekty związane z utrzymaniem usług muszą zwrócić uwagę w trakcie planowania oraz realizacji projektu. Podobnie menadżerowie usług i osoby pełniące role opisane w ITIL, wiedzą jak projektowanie, wdrażanie, utrzymanie i doskonalenie usług może wpływać na realizację projektów [por. ItSMF 2012]. Ta sytuacja powoduje jednak również powstawanie zjawiska negatywnego. Są bowiem obszary, które nie są wystarczająco dobrze zdefiniowane w żadnej z tych metodyk, a mają później istotny wpływ na funkcjonowanie usługi. Brak szczegółowych praktyk jest zatem również wyzwaniem.

Zanim nowe lub zmodyfikowane rozwiązanie informatyczne będzie gotowe do użycia, konieczne jest jego wdrożenia (lub wdrożenie jego modyfikacji/aktualizacji). Systemy IT są wdrażane i modyfikowane poprzez realizację mniej lub bardziej skomplikowanych i czasochłonnych projektów. Na tym etapie stosowane są praktyki opisane w metodykach i zbiorach dobrych praktyk projektowych (takich jak np. *A guide to the project management body of knowledge (PMBOK guide) Fifth Edition*, *PRINCE2* czy *Manifesto for Agile Software Development*). W trakcie zamykania projektu lub fazy projektu rozwiązania przekazywane są do użycia oraz oddawane na utrzymanie wyspecjalizowanych służb IT. Zamknięcie projektu kończy jego odpowiedzialność za wytwarzane rozwiązania i przechodzą one do działu utrzymania. Od tego momentu szczególnie istotne jest stosowanie dobrych praktyk związanych z utrzymywaniem systemów IT opisanych np. przez AXELOS [2016]. Interakcje zachodzące pomiędzy tymi dwoma zbiorami dobrych praktyk przedstawiono na rysunku nr 1.



Rysunek 1. Powiązanie pomiędzy projektami i utrzymaniem usługami IT

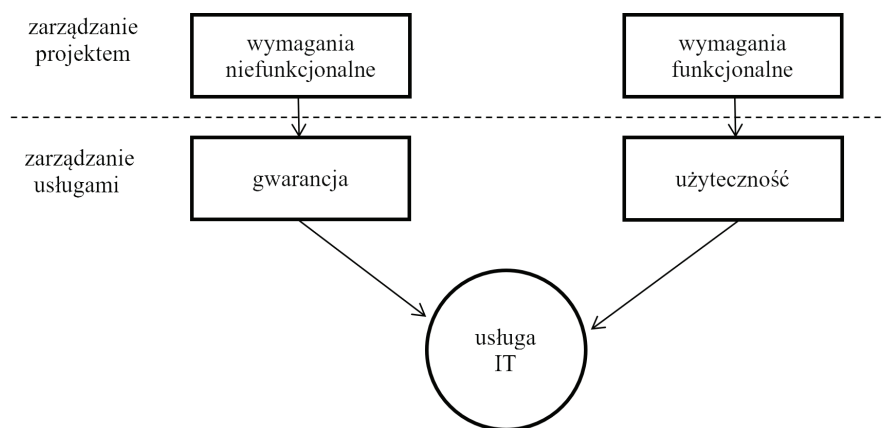


Źródło: opracowanie własne.

Biorąc pod uwagę powyższe należy zauważyć, że sposób wytwarzania systemu informatycznego w projekcie wpływa na to, w jaki sposób będzie on utrzymywany po jego zakończeniu. Ma to również znaczenie w przypadku projektów realizowanych w metodykach zwinnych [por. Stettina, Kroon 2013]. Równocześnie wymagania zdefiniowane przez dział utrzymania będą miały wpływ na zakres, koszt i czas trwania projektu. W tym momencie należy podkreślić, iż definiując zakres projektów informatycznych konieczne jest uwzględnienie zestawu wymagań dotyczących wdrażanych produktów. Są to zarówno wymagania funkcjonalne, niefunkcjonalne oraz inne. Te pierwsze określają co dany system ma robić, drugie – jak dobrze system ma funkcjonować oraz jakie warunki muszą być spełnione, aby system mógł w ten sposób funkcjonować. W konsekwencji należy dostrzec związek pomiędzy użytecznością usługi IT a sposobem, w jaki zostały zdefiniowane wymagania niefunkcjonalne. Opisane powyżej zależności zostały przedstawione na rysunku 2.



Rysunek 2. Wpływ wymagań na gwarancje i użyteczność usług IT



Źródło: opracowanie własne.

Konieczne jest zatem dokonanie weryfikacji czy pomiędzy metodykami zarządzania projektami i metodykami dotyczącymi zarządzania usługami IT nie istnieją luki, których efektem może być kończenie projektów z tylko pozornym (jak to zdefiniowano w Manifesto, C.H.A.O.S. [2013]), a nie realnym sukcesem (zdefiniowanym przez autora we wstępie). Analiza ta zostanie dokonana w dalszej części opracowania. Pomocne będą wskazówki teoretyczne zawarte w pracy Niazi, Mahmood, Alshayeb i in. [2016].

Identyfikacja luk występujących na styku zarządzania projektami i zarządzania usługami IT

Poszukując przyczyn leżących u podstaw tego, iż projekty osiągają pozorne, a nie pełne sukcesy zbadano Specyfikacje Istotnych Warunków Zamówienia (SIWZ) publikowanych za pośrednictwem Biuletynów Informacji Publicznej (BIP) na stronach jednostek administracji publicznej. Przeprowadzono również analizę przypadków oraz syntezy doświadczeń wynikających z pracy w firmach wdrażających projekty w branży bankowej i ubezpieczeniowej. Na tej podstawie stwierdzono, że na styku świata projektów i zarządzania usługami IT istnieje wiele wyzwań, których sprośowanie może przyczynić się do wdrożeń zakończonych realnym sukcesem. Są to w szczególności następujące sytuacje:



- użytkownicy systemów definiując swoje wymagania na etapie planowania projektu, skupiają się na tym co system ma robić (wymagania niefunkcjonalne), zasadniczo nie definiują jednak swoich wymagań odnośnie tego, jak dobrze system ma to robić (wymagania funkcjonalne). W rezultacie specyfikacja istotnych warunków zamówienia często nie odnosi się kompleksowo do wymagań niefunkcjonalnych;
- kierownicy projektów skupiają się na dostarczaniu funkcjonalności, a nie elementów, które są potrzebne do utrzymania usługi IT (np. dokumentacji, monitoringu rozwiązania, procedury odtwarzania systemu po awarii (*fail over*), szkoleń pracowników, wymagań odnośnie wydajności i pojemności itp.);
- osoby odpowiedzialne za utrzymanie usług IT nie zawsze są postrzegane jako interesariusze w projekcie, a w konsekwencji osoby te nie są przygotowane do przyjęcia rozwiązania w utrzymanie i nie są angażowane bezpośrednio w proces projektowy, zwłaszcza na etapie definiowania wymagań;
- kierownik projektu nie jest skłonny do wzięcia na siebie kosztów związanych z przygotowaniem usług do utrzymania i przenosi je na dział utrzymania;
- dokumenty pozwalające na inicjację projektu, takie Business Case czy dokument inicjujący projekt (DIP), odnoszą się do korzyści jakie odniesie dany podmiot po realizacji przedsięwzięcia. Bazuje on na perspektywach długookresowych. Charakterystyczne dla uzasadnienia biznesowego projektu jest definiowanie okresu zwrotu (PP – *Payback Period*) czy definiowanie stopy zwrotu (ROI – *Return on Investment*) dla określonej perspektywy czasowej, jednakże w momencie, gdy badane jest czy projekt zwrócił się w zadanym okresie lub czy uzyskał dla danego okresu oczekiwaną stopę zwrotu, projekt jest już formalnie zakończony, a zespół projektowy rozwiązany, w związku z czym projekt skupia się na tym, aby dostarczyć produkt, usługę lub efekt w określonym czasie i po określonych kosztach i z tego jest rozliczany kierownik projektu. Nie jest on jednocześnie realnie rozliczany z okresu zwrot i stopy zwrotu;
- osiągnięcie celów podanych w dokumencie Business Case lub DIP zależne jest zarówno od tego co się dzieje w projekcie oraz od tego co będzie się działo z wdrożonym rozwiązaniem po zakończeniu projektu. Jednak projekt jako byt tymczasowy, nie jest bezpośrednio zainteresowany tym, co będzie się działo po jego zakończeniu i nie jest z tego rozliczany.

Na podstawie zidentyfikowanych powyżej luk, wyciągnięto następujące wnioski:

- praktyki w zakresie współpracy pomiędzy projektem a działem utrzymania nie są jeszcze wystarczająco dobrze zdefiniowane,
- rozpoczęcie współpracy pomiędzy projektem i działem odpowiedzialnym za utrzymanie usług następuje zbyt późno,



- projekt skupia się na dostarczeniu oczekiwanej funkcjonalności w podanym czasie, kosztach i w zdefiniowanym wcześniej zakresie co oznacza osiągnięcie tylko pozornego sukcesu,
- aspekty związane z gwarancją (*warranty*) i wymaganiami niefunkcjonalnymi nie są dla projektów wystarczająco widoczne.

Kierunkowe propozycja usprawnień

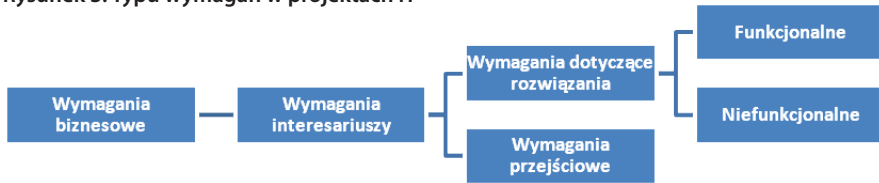
Zidentyfikowane luki wpływają tylko na pozorne odnoszenie sukcesów przez projekty informatyczne. Z tego względu konieczne jest zastosowanie praktyk wskazanych poniżej.

1) Właściwe zdefiniowanie wymagań projektowych ze szczególnym uwzględnieniem wymagań niefunkcjonalnych, które w sposób bezpośredni wpływają na jakość wdrażanego rozwiązania [por. Saini, Dubey, Rana 2011] i sposób jakie będzie ono utrzymywane. Na tym etapie szczególnie istotne jest zastosowanie właściwych praktyk związanych z definiowaniem wymagań funkcjonalnych i niefunkcjonalnych. Dobre praktyki w tym zakresie zdefiniowane są przez International Institute of Business Analysis w publikacji *A guide to the Business Analysis Body of Knowledge v 3*, a model FURPS oraz FURPS+ [Paech, Kerkow 2004] pozwala na skatalogowanie wymagań, tak aby żadne z nich nie zostało pominięte. Grupy wymagań które powinny być zebrane już jest etapie planowania projektu przedstawione zostały na rysunku nr 3. Na tym etapie należy również wykorzystać praktyki w zakresie zarządzania wymaganiami w projektach jakie definiuje Kossmann [2013].

2) Wczesne zaangażowanie w projekt (który taką usługę dostarcza lub modyfikuje) zespołów odpowiedzialnych w przyszłości za utrzymanie tejże usługi IT jest konieczne. Kossmann [2013] określił praktyki, które powinny być stosowane na tym etapie.

3) Rozliczenie kierowników projektów z sukcesu projektu według nowej definicji podanej w tym artykule, skupiającej się na długofalowych korzyściach z realizacji projektu, a nie tylko na dostarczeniu projektu w zadany czasie, budżecie i zakresie. W tym obszarze krytyczne są wskazówki przekazywane przez Badewi [2016].

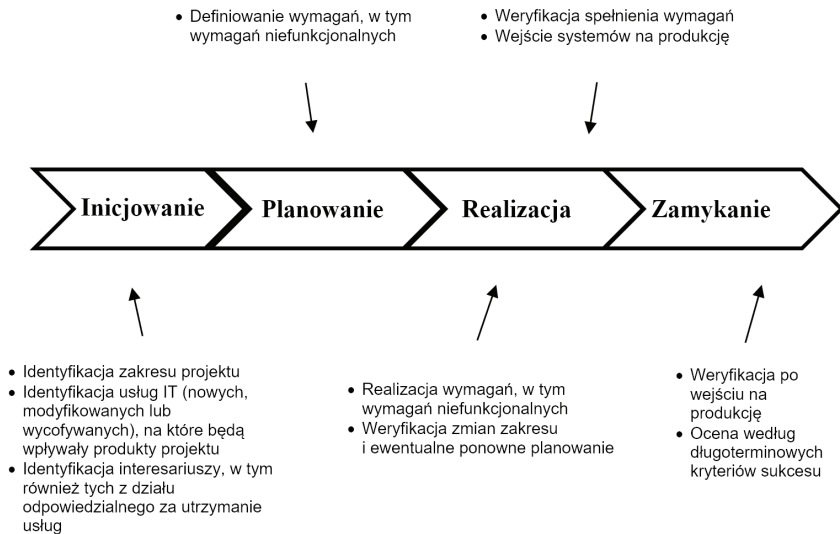
Rysunek 3. Typu wymagań w projektach IT



Źródło: opracowanie własne na podstawie International Institute of Business Analysis [2015].

Koncepcja praktyk jakie powinny być stosowane w różnych fazach projektowych, aby zwiększyć realne szanse sukcesu projektu została przedstawiona na rysunku nr 4.

Rysunek 4. Koncepcja praktyk jakie powinny być stosowane w różnych fazach projektowych, aby zwiększyć szanse realne sukcesu projektu



Źródło: opracowanie własne.

Dodatkowo, bazując na zidentyfikowanych lukach, zaproponowano kolejne działania usprawniające. Przedstawiono je w tabeli 2. Mają one charakter kierunkowy i wymagają doszczegółowienia w ramach dalszych badań.

Tabela 2. Możliwe działania usprawniające na styku działań projektowych i utrzymaniowych

Zidentyfikowana luka	Proponowane rozwiązanie
Użytkownicy systemów definiując swoje wymagania na etapie planowania projektu, skupiają się na tym co system ma robić (wymagania funkcjonalne), zasadniczo nie definiują jednak swoich wymagań odnośnie tego, jak dobrze system ma to robić (wymagania нефunkcjonalne).	Obowiązkowe definiowanie wymagań нефunkcjonalnych już na etapie inicjowania i planowania projektu. Pozwoli to jednocześnie uniknąć zmian w projekcie wydłużających jego czas, zwiększających koszty oraz poszerzających jego zakres. Pomocne będą tutaj wytyczne zawarte w A guide to the Business Analysis Body of Knowledge v 3.
Kierownicy projektów skupiają się na dostarczaniu funkcjonalności, a nie elementów, które są potrzebne do jej utrzymania.	Testy rozwiązania powinny również uwzględniać testy wymagań нефunkcjonalnych. Spełnienie wymagań нефunkcjonalnych powinno być warunkiem koniecznym do zamknięcia całego projektu lub jego fazy.
Osoby odpowiedzialne za utrzymanie usług IT nie zawsze są postrzegane jako interesariusze w projekcie, a w konsekwencji osoby odpowiedzialne za utrzymanie usług IT nie są przygotowane do przyjęcia rozwiązania w utrzymanie i nie są angażowane bezpośrednio w proces projektowy.	Zaktualizowanie zbiorów dobrych praktyk w zakresie zarządzania projektami w zakresie postrzegania osób odpowiedzialnych za utrzymanie systemu jako kluczowych interesariuszy. Wykorzystanie dobrych praktyk zarządzania interesariuszami opisanych przez International Institute of Business Analysis [2015]. Formalne angażowanie osób odpowiedzialnych za utrzymanie przyszłego rozwiązania w definiowanie wymagań нефunkcjonalnych, które będą obowiązujące dla danego projektu.
Project manager nie jest skłonny wzięcia na siebie kosztów związanych z utrzymaniem ani przygotowaniem usług do utrzymania i przetrzuca te koszty na utrzymanie.	Przypisanie project managerowi odpowiedzialności za dostarczenie kompleksowego rozwiązania, w pełni zdatnego do utrzymania po wdrożeniu na produkcję, spełniającego wszystkie typy wymagań.
Projekt skupia się na tym, aby dostarczyć produkt, usługę lub efekt w określonym czasie i po określonych kosztach i z tego rozliczany jest kierownik projektu.	Oparcie kryteriów sukcesu projektów i premii dla project managera również o wskaźniki długo-okresowe.

Źródło: opracowanie własne.

Zakończenie i kierunki dalszych badań

Metodyki zarządzania projektami oraz te dotyczące zarządzania usługami IT są istotnymi opracowaniami definiującymi zasady postępowania we właściwych im obszarach. Mają one charakter wytycznych i nie zawierają detalicznych wskazówek dotyczących postępowania w konkretnych sytuacjach. Niewątpliwie oba obszary wzajemnie się przenikają i posiadają części wspólne oraz punkty styku. Jednak zasady odpowiedzialności w tych miejscach nie są jasno ustalone.



Istotnym obszarem, który nie jest wprost zdefiniowanych w żadnej z tych metodyk jest obszar związany z zarządzaniem wymaganiami niefunkcjonalnymi. Definiowanie wymagań niefunkcjonalnych powinno odbywać się już w fazie planowania projektu, gdyż ma to istotny wpływ na zakres i harmonogram oraz budżet projektu. W tym zakresie powinny zostać zastosowane najlepsze praktyki wskazane w International Institute of Business Analysis [2015].

Stosowanie powyższych praktyk w oparciu o przedstawioną w tym artykule koncepcję pozwoli na zwiększenie prawdopodobieństwa sukcesu projektu informatycznego.

Bibliografia

AXELOS (2016), ITIL® Practitioner Guidance

Badewi A. (2016), *The impact of project management (PM) and benefits management (BM) practices on project success: Towards developing a project benefits governance framework*, „International Journal of Project Management”, 34(4), ss. 761–778.

Horobet Y., Miłosz M. (2017). *Pojęcie, modele i metryki jakości oprogramowania – przegląd*, „Journal of Computer Sciences Institute”, nr 4, ss. 92–98.

ITSMF, U.K. (2012), *ITIL Foundation Handbook*, The Stationery Office, Norwich.

International Institute of Business Analysis (2015), *A guide to the Business Analysis Body of Knowledge®*, v 3.

Kossmann M. (2013), *Requirements management: How to ensure you achieve what you need from your projects*, Gower.

Manifesto C.H.A.O.S. (2013), *Think Big, Act Small*, The Standish Group International Inc.

Niazi M., Mahmood S., Alshayeb M. i in. (2016), *Toward successful project management in global software development*, „International Journal of Project Management”, 34(8), ss. 1553–1567.

Paech B., Kerkow D. (2004), *Non-functional requirements engineering-quality is essential* [w:] 10th International Workshop on Requirements Engineering Foundation for Software Quality, Ryga, ss. 237–250.

Project Management Institute (2013), *A guide to the project management body of knowledge (PMBOK guide) Fifth Edition*, Project Management Institute, Newtown Square.

Saini R., Dubey S.K., Rana A. (2011), *Analytical study of maintainability models for quality evaluation*, „Indian Journal of Computer Science and Engineering”, 2(3), ss. 449–454.



Stettina C.J., Kroon E. (2013). *Is there an agile handover? an empirical study of documentation and project handover practices across agile software teams* [w:] *Engineering, Technology and Innovation (ICE) & IEEE International Technology Management Conference, 2013 International Conference, IEEE*, ss. 1–12.