

UNIwersytet Warszawski
Wydział Zarządzania



MGR INŻ. TOMASZ DERĘGOWSKI

Autoreferat pracy doktorskiej pt.

**METODYKA ADAPTACYJNEGO ZARZĄDZANIA PROJEKTEM
INFORMATYCZNYM W ŚRODOWISKU KORPORACYJNYM**

Opiekun naukowy:
prof. dr hab. Witold Chmielarz
Uniwersytet Warszawski, Wydział Zarządzania

Warszawa, 2017

Spis treści konspektu

1	Uzasadnienie wyboru tematu	2
2	Cele rozprawy doktorskiej	6
3	Teza rozprawy i hipotezy badawcze	7
4	Zakres pracy. Metody i techniki badawcze	9
5	Charakterystyka i wyniki przeprowadzonych badań	12
6	Wnioski końcowe	16
7	Ograniczenia i dalsze kierunki badań	Error! Bookmark not defined.
	Wybrane pozycje literatury	20
	Bibliografia.....	Error! Bookmark not defined.

1 Uzasadnienie wyboru tematu

Rozwiązania informatyczne stanowią kluczowy element wspierający działalność operacyjną większości współczesnych firm, niezależnie od dziedziny biznesowej w ramach, której operują. Oprogramowanie stanowi kluczowy element budowy przewagi konkurencyjnej przedsiębiorstw działających w tak zróżnicowanych branżach jak handel, finanse, produkcja czy też usługi medyczne. Firmy te coraz częściej są nie tylko klientami dla zewnętrznych dostawców oprogramowania, lecz również same tworzą dedykowane działy IT zajmujące się wytwarzaniem oprogramowania oraz wdrażaniem i integracją gotowych rozwiązań, na potrzeby macierzystych organizacji.

W ramach tego samego działu IT istnieje potrzeba realizacji często zróżnicowanych projektów informatycznych. Te same organizacje IT odpowiadają za wspieranie istniejących już i działających produkcyjnie systemów, integrację i wdrażanie rozwiązań dostarczonych przez firmy trzecie czy wreszcie za tworzenie zupełnie nowych, unikalnych rozwiązań. Projekty te korzystają z wielu zróżnicowanych i złożonych technologii, realizowane są przez niejednorodne zespoły różniące się liczebnością, doświadczeniem czy też rozproszeniem geograficznym.

Oczekiwania wobec organizacji IT zajmujących się realizacją zróżnicowanych projektów w tak niejednorodnym i niestabilnym środowisku są bardzo wysokie. Z jednej strony klienci wymagają elastycznego i zwinnego podejścia przejawiającego się otwartością na dynamicznie zmieniające się wymagania. Oczekują też szybkiego zwrotu z inwestycji, który w kontekście projektów informatycznych oznacza dostarczenie i produkcyjne wdrożenie przynajmniej części funkcjonalności tworzonego produktu już w początkowych etapach realizacji projektu. Odbiorcy rozwiązań informatycznych liczą też, że dostawca oprogramowania będzie w stanie na bieżąco adaptować się do zmieniających się wymagań i uwarunkowań środowiska projektowego.

Ten sam klient oczekuje jednocześnie od dostawcy rozwiązań informatycznych zastosowania dojrzałego, stabilnego procesu, który pozwoli przed przystąpieniem do realizacji projektu na precyzyjnie oszacowanie całościowego kosztu przedsięwzięcia oraz pozwoli na realizację projektu zgodnie z opracowanym na jego wstępie szczegółowym harmonogramem.

Spełnienie tak różnorodnych i złożonych wymagań przy jednoczesnym uwzględnieniu niestabilności środowiska projektowego, jest zadaniem nietrywialnym, któremu wiele organizacji informatycznych nie jest w stanie sprostać. Badania przeprowadzone w 2017 przez firmę KPMG¹, pokazują, jak niewiele projektów informatycznych kończy się sukcesem,

¹ **KPMG New Zealand**, *Driving business performance. Project Management Survey*. s.l. : KPMG, 2017.

niezależnie od tego jak sukces ten został zdefiniowany. Zgodnie z wynikami cytowanych badań tylko 31% organizacji informatycznych jest w stanie zrealizować projekt zgodnie z harmonogramem, 29% jest w stanie dostarczyć końcowe rozwiązanie w ramach inicjalnie ustalonego budżetu, natomiast tylko 33% organizacji jest w stanie dostarczyć rozwiązanie, które spełnia inicjalnie wyznaczone cele biznesowe. Jednocześnie porażka projektu ma wpływ na całość organizacji w ramach, której jest on realizowany. Badania przeprowadzone przez McKinsey & Company we współpracy z Uniwersytetem Oxford² pokazują, że porażka 17% projektów informatycznych o dużej skali może w konsekwencji zagrozić funkcjonowaniu całego przedsiębiorstwa.

Aby wspomóc organizacje informatyczne i zwiększyć prawdopodobieństwo sukcesu realizowanych projektów i odpowiadających za nie organizacji, opracowano szereg metodyk, narzędzi i standardów wspierających zarządzanie projektami informatycznymi. Narzędzia te podzielić można na dwie główne kategorie. Pierwsza z nich zawiera metodyki i narzędzia reprezentujące podejście klasyczne, procesowe, które często ze względu na swoją graficzną reprezentację nazywane jest podejściem kaskadowym³. Podejścia kaskadowe składają się z kilku niezależnych faz, każda z nich reprezentuje kolejny etap cyklu wytwarzania oprogramowania taki jak zbieranie wymagań, przygotowanie projektu architektury, implementacja, testowanie czy produkcyjne wdrożenie systemu. Poszczególne fazy są rozdzielne, przejście do kolejnego etapu jest możliwe tylko w przypadku zakończenia poprzedzającej go fazy. Ponieważ powrót do wcześniejszych etapów jest często bardzo kosztowny, podejścia procesowe są wysoce sformalizowane, produkt każdego etapu jest szczegółowo udokumentowany a przejście pomiędzy kolejnymi fazami wymagają szeregu akceptacji. Jednocześnie w ramach podejścia procesowego możliwe jest opracowanie precyzyjnych planów i harmonogramów, które szczegółowo opisują kolejne etapy realizacji projektu. Podejście procesowe zawierają też często szereg dodatkowych mechanizmów pozwalających precyzyjnie regulować wiele istotnych aspektów związanych z wytwarzaniem oprogramowania takich jak zarządzanie ryzykiem czy definiowanie i zarządzanie zmianą wymagań. Doskonale sprawdzają się one w środowisku, gdzie kluczowym wymaganiem jest przewidywalność i stabilność realizowanych procesów.

Druga grupa narzędzi zawiera podejścia nazywane powszechnie zwinnymi (ang. *Agile*) opiera się na założeniach zdefiniowanych w ramach Zwinnego Manifestu (ang. *Agile Manifesto*) i Zwinnych Zasad⁴ (ang. *Agile Principles*). Zwinne metodyki i narzędzia

² Michael Bloch, Sven Blumberg, Jürgen Laartz. Digital McKinsey. *Delivering large-scale IT projects on time, on budget, and on value*. [Online] McKinsey & Company, 2012. <http://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/delivering-large-scale-it-projects-on-time-on-budget-and-on-value>.

³ Winston Royce, *Managing the Development of Large Software Systems*, Proceedings of IEEE WESCON, 26 (Sierpień 2017): 1–9

⁴ Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor,

charakteryzują się wysoką elastycznością i niskim stopniem formalizacji procesów. Oparte są na iteracyjnym i przyrostowym podejściu do wytwarzania oprogramowania. Podobnie jak w przypadku podejść klasycznych, projekt jest również podzielony na fazy, które w przypadku podejść zwinnych nazywane są iteracjami. Czas trwania pojedynczej iteracji waha się pomiędzy dwoma a czterema tygodniami. Celem każdej iteracji jest dostarczenie konkretnych funkcjonalności. W ramach pojedynczej iteracji nowe funkcjonalności są projektowane, implementowane, testowane i wdrażane. Celem kolejnych iteracji jest dostarczenie zestawu funkcjonalności, które gotowe są do produkcyjnego wdrożenia. Podejście zwinne powstało niejako w opozycji do podejść kaskadowych, większość decyzji podejmowanych jest w trakcie trwania projektu, kolejne funkcjonalności są projektowane i dostarczane cyklicznie, a wszelkie zmiany w wymaganiach i uwarunkowaniach projektu są na bieżąco adaptowane. Jednocześnie podejścia zwinne regulują ograniczony zestaw aspektów związanych z realizacją projektu informatycznego i skupiają się głównie na samym procesie wytwarzania oprogramowania.

Obie grupy podejść są powszechnie stosowane od dziesięcioleci, zostały wielokrotnie zweryfikowane a ich wady i zalety są powszechnie znane. Istnieje wiele przykładów projektów, które były zarządzane z wykorzystaniem zarówno podejść klasycznych jak i zwinnych i zakończyło się sukcesem. Równie wiele projektów zakończyło się porażką pomimo zastosowania jednego z dwóch wymienionych podejść. To, czy projekt zakończył się sukcesem czy też niepowodzeniem świadczy nie tyle o skuteczności poszczególnych metodyk i narzędzi wspierających zarządzanie projektami, ile o tym, czy metodyka lub standard zostały właściwie dobrane do charakterystyki określonego projektu, do zarządzania, którym zostały wykorzystane.

Literatura dziedziny oraz liczne badania ⁵ potwierdzają związek pomiędzy zastosowaniem poszczególnych metodyk i narzędzi wspierających proces zarządzania a rezultatem projektu. Większość z nich skupia się jednak na pokazaniu przewagi jednej z grup narzędzi nad pozostałymi, opisując, jak zastosowanie określonego rodzaju narzędzi przekłada się na współczynnik sukcesu projektu. Kilku badaczy⁶⁷ zwraca uwagę, że nie tyle użycie określonej metodyki jest czynnikiem decydującym o sukcesie projektu, ile doświadczenie w jej użyciu i umiejętność jej dostosowania do potrzeb wynikających z charakterystyki konkretnego projektu. W literaturze zarządzania brakuje jednak pozycji, które opisywałyby

Ken Schwaber, Jeff Sutherland, Dave Thomas. *Manifesto for Agile Software Development*. [Online] Agile Manifesto, 2011. <http://agilemanifesto.org/>.

⁵ **Standish Group.** *CHAOS Report 2015*, <http://blog.standishgroup.com/post/50>.

⁶ **Fortune, J., & White, D.** (2006). Framing of project critical success factors by a systems model. *International Journal of Project Management*, 24(1), 53–65.

⁷ **Shenhar, A., Tishler, A., Dvir, D., Lipovetsky, S., & Lechler, T.** (2002). Refining the search for project success factors: a multivariate, typological approach. *R and D Management*, 32(2), 111–126.

kryteria i sposób dobrania i dostosowania metodyk i narzędzi wspierających zarządzanie projektami do potrzeb wynikających z charakterystyki poszczególnych projektów.

Osoby odpowiedzialne za kierowanie Biurami Projektów a także kierownicy projektów nie dysponują narzędziami pozwalającymi dobrać lub dostosować metodykę zarządzania projektem do potrzeb projektu. Choć zdecydowana większość wyzwań i problemów związanych z poszczególnymi aspektami realizacji projektów informatycznymi znajduje pokrycie w dostępnych technikach zarządzania projektami, brak jest pojedynczej metodyki lub standardu prowadzenia projektów, które pozwalałyby zaadresować wszystkie potencjalne problemy i wyzwania związane z wytwarzaniem oprogramowania. Zagadnienie dodatkowo komplikuje fakt unikalności projektów informatycznych. Ze względu na ich zróżnicowaną charakterystykę, każdy z nich wymaga specyficznego podejścia, różnych narzędzi i metod.

Optymalnym rozwiązaniem wydaje się dostarczenie decydom odpowiedzialnym za wybór metodyk i procesów wspierających zarządzanie projektem narzędzia, które pozwalałyby każdorazowo, przed przystąpieniem do realizacji projektu, dobierać metodyki i procesy zarządcze, do potrzeb konkretnego projektu, którego realizację mają wspierać. Celem wyeliminowania problemów związanych z niekompletnością poszczególnych metodyk i z faktem, że nie adresują one wszystkich potencjalnych wyzwań związanych z dostarczeniem produktu informatycznego, zasadnym wydaje się również wzbogacenie funkcjonalności wybranej metodyki o narzędzia i techniki zaczerpnięte z innych metodyk i standardów. Tworzony proces miałby z definicji być hybrydą łączącą w sobie elementy zaczerpnięte z różnorodnych metodyk i standardów prowadzenia projektów, które zostałyby zaadoptowane do potrzeb konkretnego projektu. Dzięki takiemu podejściu wygenerowany proces pozwoliłby w optymalny sposób zarządzać projektem, uwzględniając unikalny zestaw wymagań i oczekiwań klienta oraz dostawcy a także niepowtarzalną specyfikę samego projektu. Równoległe zastosowanie kilku podejść mogło by również pozwolić na wywołanie efektu synergii i podniesienie jakości hybrydowego procesu zarządczego do poziomu niemożliwego do osiągnięcia przez pojedyncze metodyki i standardy stosowane w oderwaniu od siebie. Efektem zastosowania opisanego podejścia mogłaby być pomoc organizacjom dojrzałym, stosującym procesowe podejście do wytwarzania oprogramowania, stać się bardziej elastycznymi i lepiej reagować na nieustannie zmieniające się wymagania i otoczenie projektowe. Jednocześnie organizacje stosujące zwinne podejście do wytwarzania oprogramowania dzięki wykorzystaniu elementów podejść procesowych mogłyby podnieść dojrzałość stosowanych procesów i rozwiązań. Proponowane podejście hybrydowe stanowi alternatywę dla powszechnie stosowanego podejścia homogenicznego, w ramach, którego organizacje informatyczne decydują się na wybór jednego z dwóch podejść: procesowego lub zwinnego, jednocześnie odcinając się od potencjalnych korzyści oferowanych przez podejścia alternatywne.

Decyzja o wyborze odpowiedniej metodyki zarządzania projektem jest często kluczowa i może w sposób istotny wpłynąć na rezultat projektu. Na chwilę obecną brak jest niestety metod i standardów wspierające opisane podejście do kreowania procesów zarządczych na potrzeby konkretnych projektów informatycznych. Brak jest narzędzi pozwalających opisywać i analizować charakterystykę projektu informatycznego by na tej podstawie wybrać metodykę dostosowaną do jego unikalnych potrzeb i uwarunkowań. Podejście polegające na łączeniu elementów różnych metodyk w ramach jednego procesu nie jest również powszechnie stosowane, brakuje też jego opisu w literaturze fachowej.

Jako że dobór odpowiednik metodyk i narzędzi stanowi jeden z kluczowych czynników decydujących o powodzeniu projektów informatycznych, analiza metodyk i narzędzi wspierających proces zarządzania projektami informatycznymi pod kątem ich dopasowania do charakterystyki projektu oraz kompletności, powinny być przedmiotem dalszych badań. Analiza taka powinna być prowadzona zarówno na poziomie pojedynczych organizacji informatycznych, zajmujących się wytwarzaniem oprogramowania, jak i całego przemysłu.

W świetle przedstawionej analizy stanu rzeczy dotyczącej metod zarządzania projektami informatycznych sformułowane zostały dwa problemy badawcze rozprawy:

1. Brak jest mechanizmów i narzędzi pozwalających dobierać metodyki zarządzania projektami do potrzeb wynikających z charakterystyki projektu.
2. Brak jest narzędzi i metod, umożliwiających budowę adaptacyjnego procesu zarządzania projektem informatycznym, który w ramach jednej, spójnej definicji procesu, łączyłby elementy różnych, często odległych podejść do zarządzania projektami informatycznymi.

Niniejsza rozprawa stanowi próbę rozwiązania obu problemów. Zaproponowany w jej ramach model ma za zadanie wspomóc organizacje informatyczne w określeniu charakterystyki projektu i w zbudowaniu na jej podstawie optymalnego, dostosowanego do unikalnej specyfiki projektu procesu zarządczego. Zbudowany z wykorzystaniem proponowanego modelu proces stanowi z założenia hybrydę, łączącą w ramach jednego procesu elementy różnych, często odległych, podejść do zarządzania projektami informatycznymi.

2 Cele rozprawy doktorskiej

Celem nadrzędnym dysertacji, było opracowanie nowatorskiej metodyki zarządzania projektami informatycznymi, która pozwalałaby na budowę dedykowanego procesu zarządzania projektem, dostosowanego do unikalnej charakterystyki projektu. Działanie metodyki polega na adaptacji elementów istniejących metodyk i narzędzi wspierających zarządzanie projektami informatycznymi do unikalnych potrzeb noworozpoczynanego projektu. Zbudowany z wykorzystaniem proponowanej metodyki proces stanowi z założenia

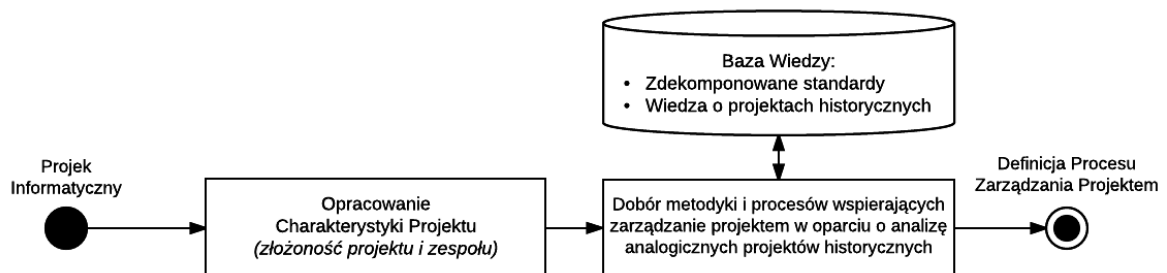
hybrydę, łączącą w ramach jednego, spójnego procesu, elementy różnych, często odległych, podejść do zarządzania projektami informatycznymi.

Celem metodyki jest wsparcie decydentów odpowiedzialnych za realizację projektów informatycznych i dostarczenie im narzędzia umożliwiającego wybór optymalnych, dostosowanych do unikalnej specyfiki projektu, metodyk, technik i narzędzi wspierających zarządzanie projektem.

Projektowanie procesu składa się z dwóch etapów. W ramach pierwszego z nich, dokonywana jest analiza złożoności projektu i zespołu projektowego, które składają się na opis charakterystyki projektu. Celem drugiego etapu jest dobór metodyki oraz procesów i technik wspierających zarządzanie projektem, które byłyby optymalnie dobrane i dostosowane do potrzeb wynikających z charakterystyki analizowanego projektu. Wiedzy o tym, które narzędzia i techniki są optymalne w kontekście analizowanego projektu, dostarcza analiza zakończonych projektów historycznych o zbliżonej charakterystyce. Informacje o projektach historycznych, ich specyfice, rezultacie i czynnikach go kształtujących, a także wiedza o zastosowanych do zarządzania nimi metodykach i narzędziach wspomagających zarządzanie, zgromadzona jest w Bazie Wiedzy.

Zaproponowane rozwiązanie przeznaczone jest dla środowiska korporacyjnego. Działanie metodyki opiera się na założeniu, że wdrażająca ją organizacja dysponuje wiedzą i doświadczeniem wynikającym z realizacji przynajmniej kilkunastu wcześniejszych projektów historycznych, które były przez nią realizowane i których charakterystyki mogą zasilić Bazę Wiedzy. Zastosowane w metodyce mechanizmy budowy zestawu rekomendacji na podstawie analizy analogicznych projektów historycznych wymagają bowiem materiału porównawczego, który jest konieczny do budowy wartościowego zestawu rekomendacji. Mniejsze organizacje informatyczne mogą nie dysponować wiedzą i doświadczeniem wynikającym z realizacji wystarczającej liczby projektów.

Produktem końcowym metodyki jest definicja procesu zarządzania projektem, który to proces dostosowany jest do unikalnej specyfiki projektu. Koncepcja działania metodyki została przedstawiona na Rys. 1



Rys. 1 Koncepcja modelu wspierającego budowę dedykowanego procesu zarządzania projektem w oparciu o analizę analogicznych projektów historycznych

Oprócz celów nadrzędnych w ramach dysertacji wyróżniono i zrealizowano również cele poznawcze, metodyczne oraz utylitarne.

Cele poznawcze

Cele poznawcze rozprawy obejmowały przede wszystkim analizę funkcjonalności i ocenę możliwości wybranych metodyk i narzędzi wspierających zarządzanie projektami informatycznymi. Przeprowadzona analiza miała na celu lepsze poznanie poszczególnych narzędzi a także pełniejsze zrozumienie ich braków i ograniczeń. Na tym etapie podjęta została również próba wykazania, jak ograniczenia poszczególnych metodyk mogą zostać zniwelowane poprzez zastosowanie wybranych elementów zaczerpniętych z innych, często odległych podejść. Wykazanie możliwości i celowości łączenia elementów różnych metodyk i podejść celem niwelowania ich wzajemnych ograniczeń stanowi podstawową wartość dodaną, wpisującą się w realizację celów metodycznych rozprawy.

Cele metodyczne

Mechanizm porównywania i wyszukiwania projektów o analogicznej charakterystyce, który stanowi jeden z fundamentów działania opisanej w ramach niniejszej rozprawy metodyki, wymagał opracowania zestawu kryteriów pozwalających na jednoznaczny i precyzyjny opis charakterystyki projektu. Dla tego też w ramach celów metodycznych rozprawy zaproponowano i określono katalog cech składających się na charakterystykę projektu informatycznego i realizującego go zespołu projektowego, które mają kluczowe znaczenie dla doboru odpowiednich metodyk i narzędzi wspierających proces zarządzania projektem. Celem było określenie zestawu parametrów definiujących złożoność projektu i zespołu projektowego oraz opracowanie narzędzi służących ich mierzeniu.

Drugi z celów naukowych dotyczył opracowania metody budowy adaptacyjnego procesu zarządczego w oparciu o analizę charakterystyki projektu, na potrzeby którego tworzony jest proces zarządzania oraz analizę analogicznych projektów historycznych o zbliżonej charakterystyce. Celem było również określenie, czy możliwe jest stworzenie spójnego, adaptacyjnego procesu zarządczego, łączącego w ramach jednego procesu zarządzania elementy różnych metodyk i standardów zarządzania projektami.

Cele utylitarne

Sformułowane cele naukowe miały podłoże utylitarne. Równoległe z realizacją celów metodycznych, zrealizowane zostały również cele praktyczne, polegające na opracowaniu założeń i specyfikacji funkcjonalno-technicznej do budowy metodyki wspierającej budowę optymalnych procesów zarządczych, dostosowanych do unikalnej specyfiki projektu. Osiągnięcie celów utylitarnych wymagało po pierwsze opracowania metody pozyskiwania od ekspertów wiedzy o projektach informatycznych (zarówno historycznych jak i takich, do których realizacji jeszcze nie przystąpiono) celem opracowania ich charakterystyki. Drugim elementem tworzonej metodyki jest mechanizm przetwarzania informacji o projektach historycznych i budowania na ich podstawie zestawu rekomendacji dotyczących optymalnego kształtu procesu zarządczego na potrzeby projektów noworozpoczynanych.

Osiągnięcie dwóch opisanych celów umożliwiło osiągnięcie nadrzędnego celu utylitarnego rozprawy, którym była budowa i weryfikacja adaptacyjnej metodyki służącej budowie procesu zarządczego, dostosowanego do unikalnej specyfiki projektu. Budowa procesu odbywa się w oparciu o analizę charakterystyki projektu i analogicznych projektów historycznych.

3 Teza rozprawy i hipotezy badawcze

Podstawowym celem rozprawy, było opracowanie nowej metodyki zarządzania projektem informatycznym. Metodyka umożliwia budowę dedykowanego procesu zarządzania, dostosowanego do unikalnej charakterystyki projektu. Docelowy proces zarządczy powstaje w wyniku adaptacji istniejących metodyk, technik i standardów wspierających zarządzanie projektami informatycznymi. Kształt tworzonego procesu opiera się o doświadczenia analogicznych projektów historycznych i czynnikach, które miały decydujący wpływ na ich rezultat. Potrzeba stworzenia nowej metodyki wynika z przeświadczenia o niekompletności i fragmentaryczności istniejących standardów, które nie adresują wszystkich potencjalnych potrzeb i wymagań związanych z realizacją współczesnych projektów informatycznych. Cel rozprawy wyrażony został w jej tezie głównej:

„Zastosowanie dostosowanego do unikalnej charakterystyki projektu informatycznego procesu zarządzania projektem, wpływa pozytywnie na przebieg projektu i jego rezultat”

Ze względu na złożoność i wieloaspektowość dziedziny, obok tezy głównej rozprawy, wyodrębniono trzy hipotezy pomocnicze, które wyznaczyły kierunek i zakres prac nad będącą przedmiotem niniejszej rozprawy adaptacyjną metodyką zarządzania projektem informatycznym:

1. Istnieje możliwość łączenia elementów różnych metodyk i standardów zarządzania projektami informatycznymi w ramach jednej, spójnej definicji procesu zarządzania projektem.
2. Specyfika organizacji i realizowanych przez nią projektów, determinują wymagania odnośnie kształtu optymalnego procesu zarządzania projektem.
3. Zasadnym jest wykorzystanie wiedzy i doświadczeń z realizacji analogicznych projektów historycznych przy projektowaniu dedykowanego procesu zarządzania na potrzeby projektu noworozpoczętego.

4 Zakres pracy

Proces weryfikacji hipotezy głównej oraz hipotez pomocniczych przebiegał w dwóch etapach. Pierwszy z nich obejmował opracowanie autorskiej metodyki umożliwiającej adaptację elementów istniejących narzędzi i standardów wspierających zarządzanie projektami do wymagań określonego projektu informatycznego. Do budowy procesu wykorzystana została wiedza i doświadczenia analogicznych projektów historycznych, a wynikowy proces dostosowany został do niepowtarzalnej charakterystyki projektu, na potrzeby którego został stworzony. Drugi z etapów weryfikacji polegał na zastosowaniu i zweryfikowaniu zaprojektowanego z wykorzystaniem metodyki AZP procesu zarządzania projektem w rzeczywistym środowisku projektowym. Etap ten objął również ocenę procesu przez końcowego odbiorcę, oraz określenie wpływu zastosowania metodyki na przebieg projektu i jego rezultat.

Struktura rozprawy odzwierciedla przyjęty kierunek badań. Tekst rozprawy podzielony został na siedem części, których struktura oddaje chronologiczny przebieg badań nad metodyką. Wstęp oraz cztery pierwsze rozdziały stanowią podsumowanie pierwszego etapu prac w trakcie, którego opracowana została Metodyka Adaptacyjnego Zarządzania Projektem (AZP). Rozdział piąty stanowi opis procesu weryfikacji metodyki i jego rezultatów. W podsumowaniu dysertacji, przedstawiona została ewaluacja przeprowadzonych badań.

We wstępie do rozprawy zdefiniowano kluczowe zagadnienia takie jak problem badawczy, pytania badawcze, hipotezy pracy a także cele rozprawy (zarówno badawcze jak i użyteczne). W rozdziale tym przedstawione zostały tło i geneza poruszanych w rozprawie problemów oraz uzasadnienie wyboru tematu rozprawy, które wskazują na potrzeby współczesnych organizacji zajmujących się wytwarzaniem oprogramowania.

W rozdziale pierwszym opisano uwarunkowania organizacji zajmujących się wytwarzaniem oprogramowania, podsumowany został również stan wiedzy dotyczący istniejących metodyk i narzędzi wspierających realizację projektów informatycznych. W ramach tego samego rozdziału została także przeprowadzona kwalifikacja i analiza porównawcza narzędzi wspierających realizację projektów informatycznych, w ramach, której wyodrębnione zostały i opisane trzy grupy narzędzi:

- metodyki zarządzania projektem,
- techniki inżynieryjne,
- modele służące badaniu jakości i doskonaleniu procesów.

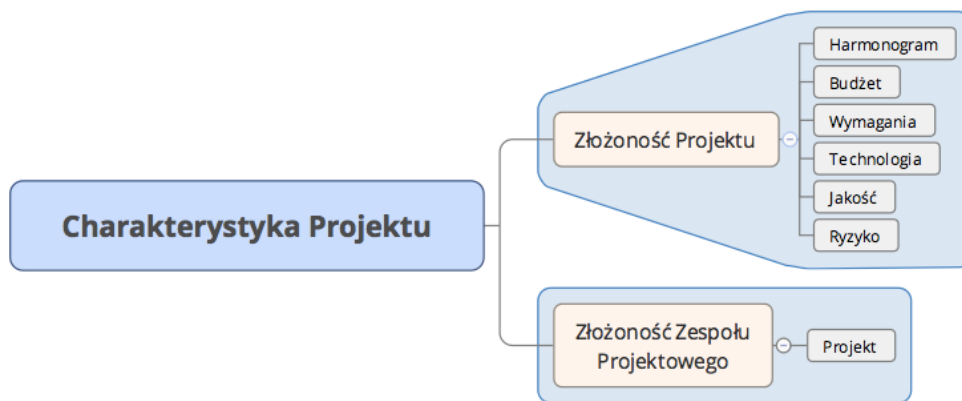
Dla każdej z grup narzędzi omówione zostały dwa przykładowe rozwiązania, z których każde reprezentuje jedno z dwóch podejść do zarządzania projektami: procesowe oraz zwinne. W rozdziale tym przedstawiono także zalety i ograniczenia poszczególnych metodyk i narzędzi, pokazane zostało również, jak poszczególne narzędzia i ich elementy łączyć można w jeden spójny proces zarządzania projektem informatycznym. Rozdział drugi wprowadził także szereg podstawowych pojęci wykorzystywanych w dalszych rozważaniach.

W rozdziale drugim przedstawione zostały podstawowe założenia przyjęte przy projektowaniu Metodyki AZP. Szczegółowo omówiono koncepcję modułowej konstrukcji procesu zarządczego a także iteracyjnego podejścia do jego budowy i wdrażania procesów zarządczym. W rozdziale tym opisano również sposób opracowywania charakterystyki projektu i zakres potrzebnych do tego informacji.

W rozdziale trzecim omówiony został mechanizm służący pozyskiwaniu wiedzy o projektach i zespołach projektowych. Przedstawiona została koncepcja rozwiązania służącego budowie charakterystyki projektu i zespołu projektowego. System pozyskiwania wiedzy o projekcie i zespole projektowym, który został opisany w ramach tego rozdziału, pozwala na dekompozycję charakterystyki projektu i zespołu projektowego na pojedyncze, dające się opisać liczbowo czynniki. Wspomniane czynniki stanowią zmienne decyzyjne wykorzystywane do porównywania projektów, wyszukiwania projektów o zbliżonej charakterystyce oraz doboru optymalnych narzędzi i procesów wspierających zarządzanie projektem.

Na potrzeby Metodyki AZP przyjęto, że charakterystyka projektu zdefiniowana jest poprzez złożoność projektu i zespołu projektowego. Złożoność projektu stanowi zagregowaną zmienną decyzyjną składającą się z sześciu aspektów, którymi są kolejno: harmonogram, budżet, wymagania, technologia, jakość oraz ryzyko. Złożoność poszczególnych aspektów określona jest przez szereg parametrów, wykorzystanych do ich opisu. W ramach analizy specyfiki projektu określa się złożoność kolejnych parametrów. Złożoność aspektu stanowi wypadkową złożoności poszczególnych parametrów wykorzystanych do jego opisu, natomiast złożoność projektu stanowi wypadkową złożoności poszczególnych aspektów.

W ramach przyjętej definicji charakterystyki projektu przyjęto, że złożoność zespołu projektowego jest jednoaspektowa. Do jej opisu wykorzystane zostały cztery parametry, którymi są kolejno: liczność zespołu, jego skład, poziom doświadczenia, jaki poszczególni członkowie zespołu mają we wspólnej pracy oraz rozproszenie geograficzne zespołu projektowego. Na Rys. 2 przedstawiono uproszczony schemat przedstawiający sposób definiowania charakterystyki projektu.



Rys. 2 Zbiór aspektów składających się na charakterystykę projektu informatycznego.

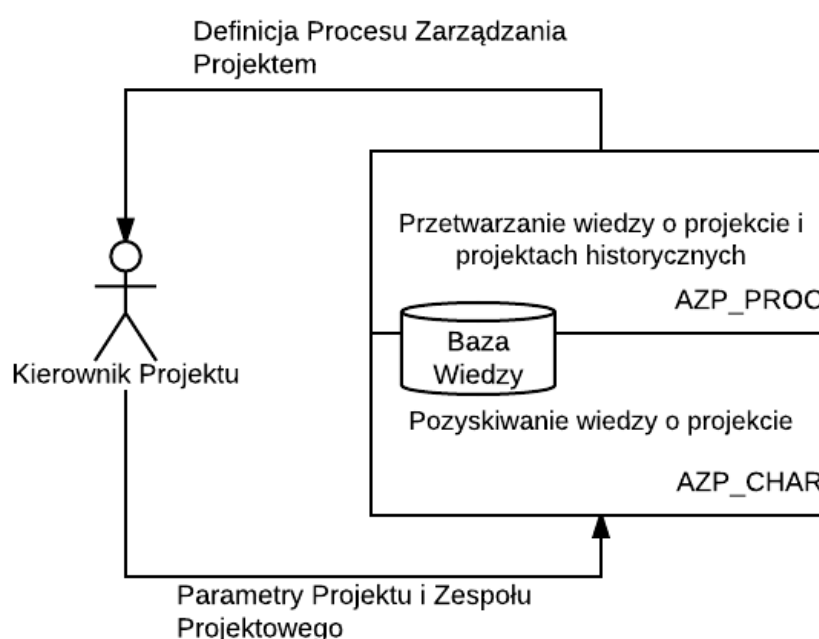
W ramach rozdziału czwartego opisana została będąca przedmiotem niniejszej rozprawy Metodyka Adaptacyjnego Zarządzania Projektem Informatycznym. Omówiona została w nim architektura metodyki, w ramach, której wyszczególniono trzy główne elementy składowe:

- *Submodel Adaptacyjne Zarządzanie Projektem – Charakterystyka (AZP_CHAR)*, który odpowiada za budowę charakterystyki projektu, zespołu projektowego oraz w przypadku projektów zakończonych zebranie informacji o rezultacie projektu i czynnikach, które miały na niego kluczowy wpływ,
- *Submodel Adaptacyjne Zarządzanie Projektem – Procesowanie (AZP_PROC)*, którego zadaniem jest przygotowanie na potrzeby projektów noworozpoczynanych zestawu rekomendacji dotyczących kształtu procesu zarządzania projektem w oparciu o analizę danego projektu i analizę analogicznych projektów historycznych,

- *Baza Wiedzy*, która wykorzystywana jest do przechowywania informacji o projektach zakończonych oraz reguł, które są podstawą przeprowadzanych mechanizmów analizy i wnioskowania.

Finalnym produktem Metodyki AZP jest definicja procesu zarządzania projektem. Definicja składa się z wybranej metodyki oraz zestawu procesów pomocniczych i technik inżynierskich, które należy wdrożyć, aby zmaksymalizować prawdopodobieństwo powodzenia projektu. Dodatkowo dla zaprojektowanego procesu, określony zostaje adekwatny stopień wydolności, który określa między innymi wymagania związane z dokumentacją i szkoleniami organizacyjnymi, a także ewentualne wymogi związane z monitorowaniem stosowanych procesów i poddawaniem ich cyklicznym audytom.

Schemat ogólny Metodyki AZP z uwzględnieniem jej głównych części składowych: submodelu AZP_CHAR, submodelu AZP_PROC oraz Bazy Wiedzy przedstawiono na Rys. 3



Rys. 3 Schemat Ogólny Metodyki Adaptacyjnego Zarządzania Projektem Informatycznym (AZP)

Drugi z etapów pracy nad metodyką, który objął jej wdrożenie i weryfikację w rzeczywistym środowisku produkcyjnym, opisany został w ostatnim, piątym rozdziale rozprawy.

Rozprawę zamyka podsumowanie, w którym opisano wnioski a także zawarto wskazówki dotyczące dalszych kierunków badań.

5 Charakterystyka i wyniki przeprowadzonych badań

Weryfikacja metodyki przebiegała w dwóch etapach. W pierwszej kolejności przeprowadzone zostały dwa eksperymenty w ramach, których weryfikacji replikatywnej poddane zostały submodele AZP_CHAR i AZP_PROC. Celem eksperymentów była

weryfikacja elementów składowych Metodyki AZP i sprawdzenie w rzeczywistym środowisku projektowym, czy są one w stanie realizować postawione przed nimi zadania. Metodyka AZP składa się z trzech elementów składowych, którymi są kolejno submodel AZP_CHAR, submodel AZP_PROC oraz Baza Wiedzy. Dane pozyskane z wykorzystaniem submodelu AZP_CHAR stanowią wejście submodelu AZP_PROC. Dla tego też w ramach pierwszego eksperymentu procesowi weryfikacji poddany został submodel AZP_CHAR. Celem przeprowadzonego eksperymentu było sprawdzenie czy zakres wiedzy zebranej z wykorzystaniem submodelu AZP_CHAR jest wystarczający dla zbudowania użytecznego zestawu rekomendacji odnośnie kształtu procesu zarządzania projektem. Po wprowadzeniu w nim zmian, będących konsekwencją przeprowadzonego eksperymentu, submodel AZP_CHAR w swej ostatecznej wersji wykorzystany został do zebrania informacji wejściowych niezbędnych do działania i weryfikacji submodelu AZP_PROC, która dokonana została w ramach drugiego eksperymentu.

W ramach trzeciego eksperymentu dokonano weryfikacji predyktywnej, która polegała na zastosowaniu Metodyki AZP w rzeczywistym środowisku projektowym. W ramach eksperymentu zbudowano z wykorzystaniem Metodyki AZP dedykowany proces zarządczy, który został następnie wdrożony, a wpływ jego zastosowania na rezultat projektu i jakość stworzonego przy jego pomocy produktu poddany został weryfikacji. Celem testowego wdrożenia Metodyki AZP było wykazanie, że koncepcja adaptacyjnego procesu zarządzania projektem stworzonego w oparciu o analizę analogicznych projektów historycznych, może zostać zastosowana w praktyce, a jej wpływ na rezultat projektu jest korzystny.

Weryfikacja replikatywna

W rezultacie pierwszego eksperymentu określono ostateczny format i zakres wiedzy pozyskiwanej z wykorzystaniem submodelu AZP_CHAR i wykorzystywanej do budowy charakterystyki projektu oraz zespołu projektowego. Wpływ niektórych z mierzonych do tej pory aspektów, na ostateczny kształt procesu zarządzania projektem, uznany został za nie wnoszący a ich badanie zostało wyeliminowane z submodelu AZP_CHAR. Przykładem takiego aspektu może być zarządzanie relacjami z dostawcami. Z kolei analiza innych aspektów, takich jak istotność projektu, które nie były uwzględnione w pierwszej wersji submodelu AZP_CHAR, została dodana do ostatecznej wersji modelu.

W trakcie pierwszego eksperymentu wykazano również, że celem usprawnienia modelu i poprawienia jakości przygotowanych z jego wykorzystaniem rekomendacji, zasadnym jest poszerzenie grona odbiorców ankiety o dodatkowe osoby i role, których wiedza i doświadczenie są kluczowe dla uzyskania pełnego obrazu projektu. W zależności od specyfiki danego projektu, obok Kierownika Projektu, w budowie charakterystyki projektu powinni również uczestniczyć eksperci z zakresu testowania, kodowani czy też wdrażania tworzonego produktu. Poziom znajomości poszczególnych aspektów przez przedstawicieli poszczególnych grup, którzy biorą udział w budowie charakterystyki, reprezentowany jest przez system wag, pozwalających określić stopień ich ekspertyzy.

W stosunku do pierwotnych założeń zasadnym okazało się również przeprowadzanie oceny klienta i zespołu jako całości, a nie w oderwaniu od siebie. Zgodnie z uwagami pochodzącymi od uczestników eksperymentu, w przypadku zdecydowanej większości projektów klient i dostawca funkcjonują jako całość, tworząc zespół projektowy.

Ostatnim z efektów pierwszego eksperymentu, było ustalenie sposobu definiowania złożoności charakterystyki projektu oraz metody porównywania projektów gwarantującej jednoznaczność i szybkość przeprowadzanej analizy.

Z kolei drugi eksperyment, doprowadził do kilku zmian w submodelu AZP_PROC. Pierwszą z nich było wykorzystanie jako szkieletu tworzonego projektu Specyficznych Praktyk a nie, jak miało to miejsce w pierwszej wersji modelu, Obszarów Procesowych. Okazało się bowiem, że koncepcja Obszarów Procesowych jest zbyt ogólna i a tworzenie w oparciu o nią precyzyjnych i wartościowych rekomendacji nie jest możliwe. Zdecydowano również, że opracowanie charakterystyki projektu będzie następować w oparciu o Specyficzne Praktyki zdefiniowane w ramach sześć z dwudziestu dwóch Obszarów Procesowych. Uznano, że takie rozwiązanie stanowić będzie rozsądny kompromis pomiędzy koniecznością zbudowania miarodajnego obrazu projektu a czasem potrzebnym na zasilenie Bazy Wiedzy i przeprowadzenie analizy projektów historycznych.

Jedną z najistotniejszych korzyścią wynikającą z przeprowadzonego drugiego eksperymentu, było jednak zweryfikowanie niektórych fundamentalnych dla metodyki AZP założeń. W efekcie pilotażowego wdrożenia submodelu AZP_PROC udało się dowieść, że wyciąganie użytecznych wniosków dotyczących konstrukcji docelowego procesu zarządzania projektem, w oparciu o analizę projektów historycznych, jest możliwe. Przeprowadzony eksperyment pokazał również w praktyce, że łączenia różnych, często odległych metodyk zarządzania projektami i modeli doskonalenia procesów (Scrum i Zintegrowanego Modelu Dojrzałości i Wydolności (ang. *Capability Maturity Model Integration, CMMI*)) w ramach jednego procesu zarządzania projektem przynosić może wymierne korzyści. Dzięki przeprowadzonym badaniom zasadności replikatywnej udało się w praktyce dowieść, że stojące za modelem

Do dalszych badań polegających na weryfikacji predyktywnej metodyki AZP wykorzystano drugą, modyfikowaną wersję metodyki AZP.

Weryfikacja predyktywna

Weryfikacja replikatywna, polegająca na wdrożeniu Metodyki Adaptacyjnego Zarządzania Projektem (AZP) pokazała, że koncepcja hybrydowego procesu zarządzania projektem, stworzonego w oparciu o analizę analogicznych projektów historycznych, jest możliwa do praktycznego zastosowania. W ramach eksperymentu udało się przygotować zestaw wartościowych rekomendacji dotyczących kształtu docelowego procesu zarządzania projektem. Rekomendacje stworzone zostały w oparciu o analizę doświadczeń analogicznych projektów historycznych, o których wiedza zgromadzona była w Bazie Wiedzy.

Testowe wdrożenie Metodyki AZP pokazało również, że możliwe jest zastosowanie w praktyce hybrydowego procesu zarządzania, który łączy w sobie elementy podejść zwinnych i kaskadowych.

Trzeci z przeprowadzonych eksperymentów wykazał również, że w przypadku, gdy wybrana do zarządzania projektem metodyka nie implementuje kluczowych z punktu widzenia charakterystyki danego projektu procesów, możliwe jest znalezienie alternatywnych sposobów ich implementacji. W przypadku projektu będącego środowiskiem weryfikacji predyktywnej, proces oparty na metodyce Scrum nie regulował wielu aspektów, które były

istotne z punktu widzenia obszaru Zarządzanie Konfiguracją. Metodyka AZP wykorzystana została do określenia alternatywnych sposobów implementacji pozwalających osiągnąć cele związane z tym obszarem (luki metodyki Scrum uzupełnione zostały o elementy metody DevOps i koncepcję Ciągłej Integracji).

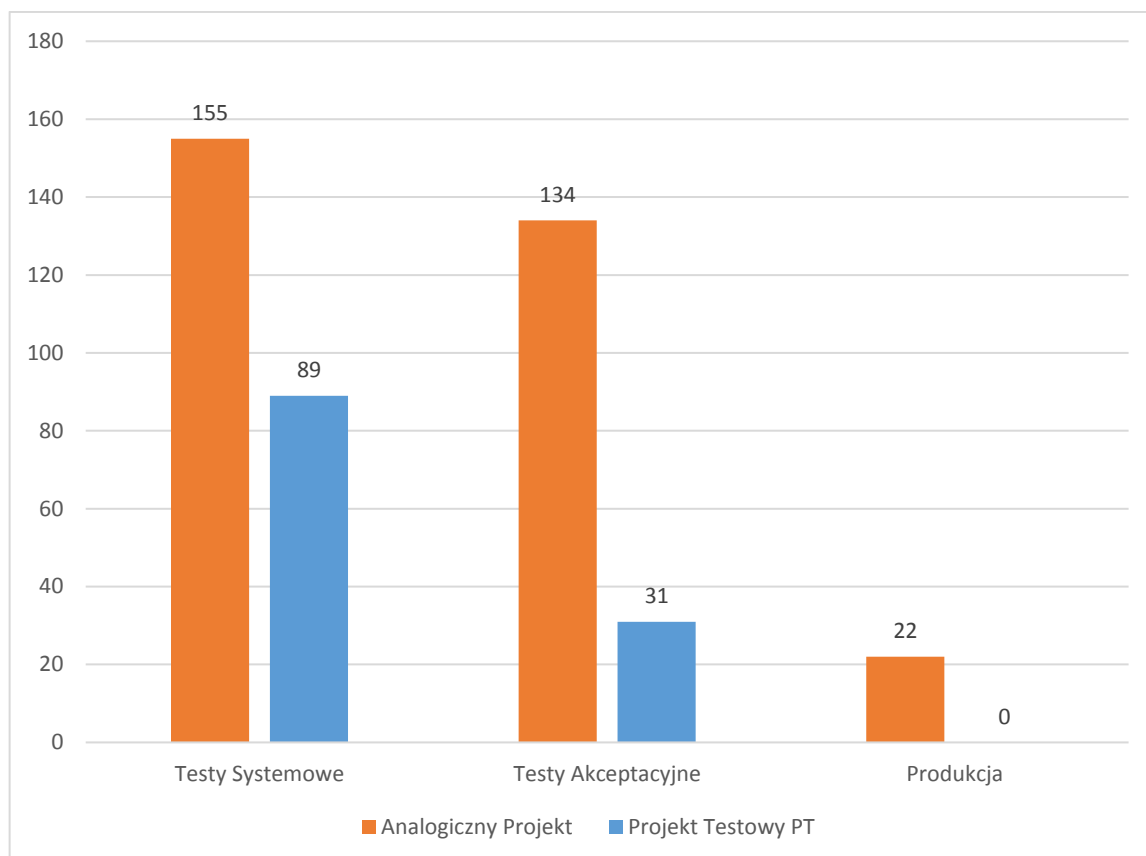
Obok empirycznego wykazania możliwości praktycznego zastosowania Metodyki AZP w ramach przeprowadzonego eksperymentu udało się również dowiedzieć, że jego zastosowanie ma pozytywny, wymierny, wpływ na projekt na potrzeby którego metodyka została wykorzystana. W ramach przeprowadzonego eksperymentu udało się zmierzyć i wykazać wymierne korzyści, które wiążą się z zastosowaniem Metodyki AZP.

Podsumowanie wpływu wdrożenia zaprojektowanego z wykorzystaniem metodyki AZP procesu zarządzania projektem na wydajność wdrażania nowych wersji produktu zostało podsumowane w Tab. 1

Tab. 1 Wpływ zmian procesu zarządzania projektem w obszarze Zarządzania Konfiguracją na wydajność projektu

	<i>Pierwotny proces</i>	<i>Proces AZP</i>	<i>Wzrost efektywności</i>
Liczba inżynierów potrzebnych do uruchomienia nowej wersji produktu	4	1	75%
Czas wdrożenie nowej wersji produktu w środowisku produkcyjnym [dni]	5	0,5	90%
Czas wdrożenie nowej wersji produktu w środowisku testowym [dni]	3	0,125	96%
Liczba wdrożeń produktu w tygodni	1	10	1000%

Zastosowany proces zarządzania projektem miał również pośredni wpływ na jakość tworzonego oprogramowania. Zaprojektowany z wykorzystaniem Metodyki AZP proces zarządzania projektem, zawierał istotne elementy wpływające na jakość tworzonego oprogramowania, takie jak zobowiązanie programistów do tworzenia testów jednostkowych, pełna automatyzacja testów systemowych, integracyjnych i regresyjnych oraz zastosowanie statycznej analizy kodu. Zastosowanie wspomnianych dodatkowych praktyk i procesów miało znaczący wpływ na jakość tworzonego rozwiązania i zmniejszenie liczby defektów aplikacji. Zestawienie metryk związanych z jakością w Projekcie Testowym P_T z metrykami analogicznego projektu realizowanego przez ten sam zespół, pokazują, jak znacząco spadła liczba defektów. Znacznie mniej błędów zostało wykrytych zarówno w trakcie testów systemowych jak i w fazie testów akceptacyjnych użytkownika. Liczba błędów zidentyfikowanych w trakcie testów systemowych została zredukowana o 42%. W trakcie testów akceptacyjnych wykryto z kolei o 75% mniej defektów. Co kluczowe, w przypadku omawianego projektu, żaden defekt nie został zgłoszony w trakcie produkcyjnego użytkowania systemu. Podsumowanie dotyczące liczby defektów przedstawione zostało na Rys. 4.



Rys. 4 Redukcja liczby błędów będąca efektem wprowadzenia usprawnień w obszarze Zarządzania Konfiguracją

Opisany eksperyment pokazał możliwość praktycznego zastosowania metodyki AZP oraz dowiódł, że koncepcja metodyki AZP może być z powodzeniem stosowana do wsparcia organizacji informatycznych. Efekty badania pilotażowego stanowią dobry prognostyk na przyszłość i stanowią uzasadnienie dla podjęcia dalszych działań, których celem jest rozwój metodyki AZP.

6 Wnioski końcowe i dalsze kierunki badań

Podstawowym celem rozprawy było opracowanie nowej metodyki, która umożliwiałaby tworzenie dedykowanego, dostosowanego do unikalnej charakterystyki projektu, procesu zarządzania projektem informatycznym. Tworzony z wykorzystaniem Metodyki Adaptacyjnego Zarządzania Projektem Informatycznym (AZP) proces zarządzania projektem powstaje w wyniku adaptacji wybranych elementów istniejących metodyk i zestawów technik inżynierskich do wymagań projektu, na potrzeby którego tworzony jest proces zarządczy. Do definicji tworzonego procesu zarządzania włączane są te elementy istniejących standardów, które w przypadku analogicznych projektów historycznych przyczyniły się do ich sukcesu, lub których brak zdecydował o ich porażce.

Propozycja stworzenia nowej metodyki wynika z przeświadczenia o niekompletności i fragmentaryczności istniejących standardów wspierających zarządzanie projektami informatycznymi, które gdy są stosowane w oderwaniu od siebie, nie adresują wszystkich

potencjalnych potrzeb i wymagań związanych z realizacją różnorodnych projektów informatycznych.

Przeprowadzone i opisane w ramach rozprawy badania i eksperymenty pozwoliły na weryfikację głównej hipotezy rozprawy, mówiącej o tym, że zastosowanie dostosowanego do unikalnej charakterystyki projektu informatycznego procesu zarządzania projektem, zwiększa prawdopodobieństwo jego powodzenia. W rozprawie wykazano, że istnieje możliwość łączenia elementów różnych metodyk i standardów zarządzania projektami informatycznymi w ramach jednej, spójnej definicji procesu zarządzania projektem. Pokazano również, że głównym czynnikiem decydującym o wyborze metodyki i narzędzi wspierających zarządzanie projektem jest specyfika organizacji i charakterystyka realizowanych przez nią projektów. Złożoność poszczególnych aspektów składających się na charakterystykę projektu i zespołu projektowego determinuje wymagania odnośnie kształtu optymalnego procesu zarządzania projektem. Wykazano wreszcie, że wiedza o analogicznych projektach historycznych może być wartościowym źródłem informacji dotyczących kształtu optymalnego procesu zarządzania projektem.

Opracowanie Metodyki AZP, jej testowe wdrożenie i weryfikacja wpływu jej zastosowania na rezultat projektu pozwala na stwierdzenie, że jest ona rozwiązaniem kompletnym. Pozwala na opracowanie charakterystyki projektu, jej analizę, porównywanie z charakterystykami projektów historycznych i opracowanie użytecznego zestawu rekomendacji, w oparciu o które zbudować można, optymalny, dostosowany do unikalnej specyfiki projektu, proces zarządzania tym projektem. Choć zestaw aspektów, składających się na charakterystykę projektu i zespołu projektowego został ograniczony, to wybrane aspekty agregują większość kluczowych parametrów, które mają wpływy na budowę optymalnego procesu. Potwierdzają to eksperymenty i weryfikacja Metodyki AZP przeprowadzona przez doświadczonych Kierowników Projektów.

Poniżej przedstawiono wady i zalety zaprezentowanej w ramach niniejszej rozprawy Metodyki Adaptacyjnego Zarządzania Projektem (AZP):

Zalety Metodyki AZP

1. *Innowacyjność*: podejście polegające na każdorazowym budowaniu dedykowanego, dostosowanego do unikalnej specyfiki projektu, procesu zarządzania tym projektem, w oparciu o analizę doświadczeń analogicznych projektów historycznych, nie było do tej pory stosowane. Również wykorzystane w ramach metodyki AZP podejście hybrydowe polegające na łączeniu w ramach jednego procesu zarządzania projektem informatycznym elementów różnych, często odległych metodyk i standardów zarządzania, projektami, stanowi nowatorskie odczytanie powszechnie znanych i stosowanych standardów.
2. *Wartość biznesowa*: sposób działania metodyki i stworzonego w oparciu o nią procesu zarządzania projektem oparty jest na praktycznych doświadczeniach komercyjnych przedsiębiorstw informatycznych i realizowanych przez nie projektów. Działanie metodyki i skuteczność stworzonego z jej wykorzystaniem procesu zarządzania projektem zostały dowiedzione w rzeczywistym środowisku projektowym. Środowisko kształtowania i weryfikacji metodyki pozwala na stwierdzenie, że jej

komercyjne wykorzystanie jest uzasadnione i może wpłynąć pozytywnie na rezultaty realizowanych przy jej wykorzystaniu projektów.

3. *Rozszerzalność*: opisany w ramach niniejszej rozprawy wariant metodyki AZP został oparty na kilku wybranych metodykach i standardach wspierających zarządzanie projektami informatycznymi. Jednym z głównych założeń przyjętych przy jej projektowaniu jest umożliwienie jej rozszerzania o dodatkowe standardy i narzędzia. Dzięki temu metodyka może być na bieżąco rozszerzana o funkcjonalności zaczerpnięte z istniejących lub nowopowstałych metodyk i standardów wspierających zarządzanie projektami.
4. *Uwzględnienie informacji zwrotnej*: Baza Wiedzy, która stanowi integralny element metodyki AZP, jest na bieżąco uzupełniana o informacje dotyczące skuteczności zaprojektowanego z wykorzystaniem metodyki AZP procesu zarządzania projektem i jego wpływu końcowy rezultat projektu. Dzięki temu możliwe jest ciągle doskonalenie architektury metodyki i sposobu jej działania.

Wady i ograniczenia Metodyki AZP

1. *Analiza ograniczonego zestawu aspektów realizacji projektów informatycznych*: Wariant metodyk, której proces weryfikacji i powstawania został opisany w niniejszej rozprawie oparty został na sześciu z dwudziestu dwóch obszarów procesowych zawartych w modelu CMMI. Oznacza to, że zakres analizowanych aspektów realizacji projektów informatycznych wynosi 27%. Nie jest to mało, zważywszy na fakt, że w opisanym wariantcie metodyki uwzględniono najistotniejsze, zdaniem autorów modelu CMMI, aspekty realizacji projektów informatycznych, które mają kluczowe znaczenie dla ich przebiegu i rezultatu. Dodatkowo nic nie stoi na przeszkodzie, aby w dalszych etapach rozwoju Metodyki AZP nie poszerzyć zakresu analizy o pozostałe Obszary Procesowe modelu CMMI.
2. *Ograniczony zestaw metodyk, modeli i standardów*: wariant metodyki AZP, który został zweryfikowany w ramach niniejszej rozprawy opiera się na ograniczonym zestawie metodyk i zestawów technik inżynierskich. W procesie weryfikacji wykorzystano dwie metodyki (Projekty w Kontrolowanych Środowiskach (ang. *Projects In Controlled Environments, PRINCE2*) oraz Scrum) oraz dwa zestawy technik inżynierskich (Proces Osobistego Programowania (ang. *Personal Software Process, PSP*) oraz Programowanie Ekstremalne (ang. *Extreme Programming, XP*)). Analogicznie jak w przypadku Obszarów Procesowych modelu CMMI, Metodyka AZP może być rozszerzana o dodatkowe metodyki, modele i standardy prowadzenia projektów.
3. *Środowisko weryfikacji*: działanie Metodyki AZP i wpływ zastosowania stworzonego z jej wykorzystaniem procesu zarządzania na rezultat projektu zostały zweryfikowane w jednej organizacji informatycznej. Rozmiar organizacji oraz różnorodność realizowanych przez nią projektów pozwalają jednak na stwierdzenie, że Metodyka AZP może być zastosowana z powodzeniem w innych organizacjach.

Opisane wady i ograniczenia metodyki AZP wyznaczają dalsze kierunki badań i jej rozwoju. Zasadnym wydaje się, aby w dalszych pracach nad metodyką AZP uwzględnić

wszystkie Obszary Procesowe modelu CMMI celem dokonania pełnej analizy obszarów i czynników mających wpływ na przebieg i rezultat projektu informatycznego. Celowym jest również przeprowadzenie dekompozycji większej liczby metodyk i zestawów technik inżynierskich wzbogacając w ten sposób zestaw narzędzi, procesów, artefaktów i ról, które włączyć można do definicji hybrydowego procesu zarządzania projektem. Wykorzystanie szerszego zestawu narzędzi ma na celu jeszcze lepsze dostosowanie wynikowego procesu zarządzania projektem do charakterystyki projektu i związanych z jego realizacją wyzwań.

Przedstawiony w niniejszej rozprawie wariant Metodyki AZP oparty została na modelu CMMI, którego przeznaczeniem jest ocena i doskonalenie procesów zarządzania projektem. W zależności od potrzeb organizacji chcącej zastosować Metodykę AZP i specyfiki realizowanych przez nią projektów, model CMMI może zostać zastąpiony dowolnym innym modelem oceny i doskonalenia procesów. Przykładowo organizacje serwisowe lub organizacje wsparcia mogłyby zastąpić model CMMI modelem Zarządzanie Infrastrukturą Informatyczną (ang. *Information Technology Infrastructure Management, ITIL*) i dzięki temu badać jakość świadczonych usług i budować zestawy rekomendacji pozwalające na ich usprawnienie. Wykorzystanie innych modeli mogłoby również pozwolić na zastosowanie metodyki AZP w firmach nie będących organizacjami informatycznymi.

W dalszych etapach badań nad metodyką AZP wskazanym byłoby również zweryfikowanie jej przydatności i skuteczności w większej liczbie organizacji informatycznych. Organizacja, w której przeprowadzono opisaną w ramach niniejszej rozprawy weryfikację stanowi zróżnicowane i wiarygodne środowisko oceny. Wdrożenie metodyki AZP w większej liczbie organizacji mogłoby jednak pozwolić na jej pełniejszą weryfikację i poszerzenie obszaru jej potencjalnych zastosowań. Rozszerzenie środowiska weryfikacji umożliwiłoby bowiem współdzielenie doświadczeń i wniosków pomiędzy różnymi organizacjami informatycznymi. Wykorzystanie w procesie analizy i wnioskowania doświadczeń wielu organizacji mogłoby pozwolić na poszerzenie możliwych zastosowań Metodyki AZP o organizacje spoza środowiska korporacyjnego. Mniejsze przedsiębiorstwa, które nie mogą sobie pozwolić na poniesienie inicjalnych kosztów związanych z zasileniem Bazy Wiedzy, lub takie, które zrealizowały ograniczoną liczbę projektów i nie dysponują doświadczeniami pozwalającymi na wyciąganie miarodajnych wniosków i budowanie wartościowych zestawów rekomendacji, mogłoby stosować Metodykę AZP wykorzystując do analizy i budowy rekomendacji doświadczenia organizacji trzecich.

Wykorzystanie w metodyce AZP elementów Nauczania Maszynowego (ang. *Machine Learning*) (Kelleher, Namee i D'Arcy, 2015) mogłoby również stanowić kierunek jej dalszego rozwoju. Automatyzacja rozpoznawania wzorców oraz wykorzystanie algorytmów, które uczą się w na podstawie danych historycznych i potrafią budować zestawy rekomendacji i przewidywać oparte na danych, mogłoby podnieść Metodykę AZP na nowy poziom.

Wybrane pozycje literatury

1. Wybrane pozycje literatury Aamodt, A., & Plaza, E. (1994). Case-Based Reasoning: Foundational Issues, Methodological Variations and System Approaches. *AI Communications*, 17(1).
2. Ahern, D. M., Armstrong, J., Clouse, A., Ferguson, J. R., Hayes, W., & Nidiffer, K. E. (2005). *CMMI SCAMPI Distilled: Appraisals for Process Improvement*. Addison-Wesley Professional.
3. Arikani, A. (2008). *Multichannel Marketing: Metrics and Methods for On and Offline Success*. Sybex.
4. AXELOS. (2011). *Management of Portfolios*. The Stationery Office.
5. Beck, K., & Andres, C. (2004). *Extreme Programming Explained: Embrace Change (2nd edition)*. Boston: Addison-Wesley.
6. Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., . . . Thomas, D. (2011). *Manifesto for Agile Software Development*. (Agile Manifesto) Retrieved from <http://agilemanifesto.org/>
7. Bergstrom, S., & Reberg, L. (2003). *Adopting the Rational Unified Process: Success with the RUP*. Boston: Addison-Wesley Professional.
8. Bisht, S. (2013). *Robot Framework Test Automation*. Packt Publishing.
9. Bloch, M., Blumberg, S., & Laartz, J. (2012). *Delivering large-scale IT projects on time, on budget, and on value*. McKinsey & Company.
10. Briski, K. A., Chitale, P., Hamilton, V., Pratt, A., Starr, B., Veroulis, J., & Villard, B. (2008). *Minimizing code defects to improve software quality and lower development costs*. Rational Software. Retrieved from <https://www.isixsigma.com/industries/software-it/defect-prevention-reducing-costs-and-enhancing-quality/>
11. Chess, B. (2007). *Secure Programming with Static Analysis*. Addison-Wesley Professional.
12. Chrissis, M. B., Konrad, M., & Shrum, S. (2011). *CMMI for Development: Guidelines for Process Integration and Product Improvement*. Addison-Wesley Professional.
13. CMMI Insitute. (2017). *Published Appraisla Results*. (CMMI Insitute) Retrieved 2017, from <https://sas.cmmiinstitute.com/pars/>
14. CMMI Insitute. (2017). *The CMMI Institute's 2017 Annual Report*. CMMI Insitute.
15. Cockburn, A. (2004). *Crystal Clear: A Human-Powered Methodology for Small Teams: A Human-Powered Methodology for Small Teams*. Addison-Wesley Professiona.
16. Cohn, M. (n.d.). *Agile Estimating and Planning*. 2005: Pearson Education.
17. Deming, E. W. (1986). *Out of the Crisis*. MIT Center for Advanced Engineering Study.
18. Deregowski, T., & Ziolkowski, A. (2014). Hybrid approach in project management – mixing CMMI model with Agile Practices. *Social Science*, 85, 64.
19. Deręowski, T., Orłowski, C., Kurzawski, M., & Artur, Z. (2016). Building Project and Project Team Characteristic for creating Hybrid Management Process.
20. Deręowski, T., Orłowski, C., Kurzawski, M., & Ziółkowski, A. (2016). Model for building Project Management processes as a way of increasing organization readiness

- for Agile transformation. *Konferencja Innowacje w Zarządzaniu i Inżynierii Produkcji, Tom II*.
21. Deręgowski, T., Orłowski, C., Kurzawski, M., & Ziółkowski, A. (2017). Building Dedicated Project Management Process Basing on Historical Experience. *ACIIDS*.
 22. Dobson, M. (2004). *The Triple Constraints in Project Management*. Management Concepts Inc.
 23. Drucker, P. (1957). *Landmarks of Tomorrow*. New York: Harper & Row.
 24. Duvall, P., Matyas, S., & Glover, A. (2007). *Continuous Integration. Improving Software Quality and Reducing Risk*. Pearson Education.
 25. Fortune, J., & White, D. (2006). Framing of project critical success factors by a systems model. *International Journal of Project Management*(24(1)), 53-65.
 26. Gene, K., Humble, J., Debois, P., & Willis, J. (2016). *The DevOps Handbook. How to Create World-Class Agility, Reliability, & Security in Technology Organisation*. Portland: IT Revolution Press.
 27. George, B., & Williams, L. (2003). An initial investigation of test driven development in industry. *Proceedings of the 2003 ACM symposium on Applied computing, 3*, 1135 - 1139.
 28. Goldratt, E. (1997). *Critical Chain*. North River Press.
 29. Grundmann, M. (2005). *A CMMI Maturity Level 2 assessment of RUP*. Retrieved from <http://www.ibm.com/developerworks/rational/library/dec05/grundmann/>
 30. Henney, K. (2010). *97 Things Every Programmer Should Know: Collective Wisdom from the Experts*. O'Reilly Media.
 31. Hornby, A. S. (2005). *Oxford Advanced Learner's Dictionary 7th edition*. Oxford University Press.
 32. Humble, J., & Farley, D. (2010). *Continuous Delivery. Reliable Software Releases through Build, Test, and Deployment Automation*. Pearson Education.
 33. Humphrey, W. S. (2005). *PSP - A Self-Improvement Process for Software Engineers*. Addison-Wesley Professional.
 34. Hunt, A., & Thomas, D. (2000). *The Pragmatic Programmer. From Journeyman to Master*. Addison-Wesley.
 35. Johnson, J. (2002). ROI, It's Your Job. *Third International Conference on Extreme Programming*. Alghero, Italy. Retrieved from <http://www.xp2003.org/talksinfo/johnson.pdf>
 36. Johnson, J., Crea, J., Vianna, L., Mulder, T., & Lynch, J. (2015). *CHAOS Report*. Standish Group.
 37. Kelleher, J. D., Namee, B. M., & D'Arcy, A. (2015). *Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies*. The MIT Press.
 38. Kim, G. (2016). *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations*. IT Revolution Press.
 39. Kniberg, H. (2007). *Scrum and XP from the Trenches*. Lulu.com.
 40. Kniberg, H., & Ivarsson, A. (2012). *Scaling Agile @ Spotify*. Retrieved 2017, from <http://abinoda.com/clippings/Scaling-Agile-at-Spotify.pdf>
 41. KPMG. (2017). *Driving Business Performance. Project Management Survey*.

42. KPMG New Zeland. (2017). *Driving business performance. Project Management Survey*. KPMG.
43. Kruchten, P. (2004). *The Rational Unified Process: An Introduction*. Addison-Wesley Professional.
44. Larman, C. (2003). *Agile and Iterative Development: A Manager's Guide 1st Edition*. Addison-Wesley Professional.
45. Martin, R. C. (2002). *Agile Software Development: Principles, Patterns and Practices*. Prentice Hall.
46. Marz, N. (2015). *Big Data: Principles and best practices of scalable realtime data systems*. Manning Publications.
47. Office of Government Commerce. (2009). *Managing Successful Projects with PRINCE2*. Stationery Office Books.
48. Ohno, T. (1988). *The Toyota Production System: Beyond Large-Scale Production*. Productivity Press.
49. Pichler, R. (2010). *Agile Product Management with Scrum: Creating Products That Customers Love*. Addison-Wesley Professional.
50. Pichler, R. (n.d.). *Agile Product Management with Scrum. Creating Products that Customers Love*. 2010: Addison-Wesley Professional.
51. Poppendieck, M., & Poppendieck, T. (2003). *Lean Software Development: An Agile Toolkit*. Addison-Wesley.
52. Project Management Institute. (2013). *A Guide to the Project Management Body of Knowledge (5th edition)*. USA: Project Management Institute.
53. Project Management Institute. (2013). *Organizational Project Management Maturity Model (OPM3) Knowledge Foundation*. Project Management Institute.
54. Richard Knaster, D. L. (2017). *SAFe 4.0 Distilled: Applying the Scaled Agile Framework for Lean Software and Systems Engineering*. Addison-Wesley Professional.
55. Roden, E. (2013). *Portfolio, Programme and Project Offices*. The Stationery Office.
56. Royce, W. (1970). Managing the Development of Large Software Systems. *Proceedings of IEEE WESCON*.
57. Sanchez, J. C., Williams, L., & Maximilien, E. M. (2007). On the Sustained Use of a Test-Driven Development Practice at IBM. *Agile Conference (AGILE)*.
58. Sante, T. v. (2007). *TOGAFTM The Open Group Architecture Framework A Management Guide*. Van Haren Publishing.
59. Schank, R. (1982). *Dynamic Memory: A Theory of Learning in Computers and People*. New York: Cambridge University Press.
60. Schwaber, K., & Beedle, M. (2001). *Agile Software Development with Scrum*. Prentice Hall.
61. Scott W. Ambler, M. L. (2012). *Disciplined Agile Delivery: A Practitioner's Guide to Agile Software Delivery in the Enterprise*. IBM Press.
62. Shenhar, A. J., & Dvir, D. (2013). *Innovation, Reinventing Project Management: The Diamond Approach To Successful Growth And*. Harvard Business Press.
63. Shenhar, A. T. (2002). Refining the search for project success factors: a multivariate, typological approach. *R and D Management*(32(2)), 111–126.

64. Stapleton, J., & Constable, P. (1997). *DSDM: Dynamic Systems Development Method: The Method in Practice*. Addison-Wesley Professional.
65. Stephen R. Palmer, J. M. (2002). *A Practical Guide to Feature-Driven Development*. Prentice Hall.
66. Sutherland, J. J. (2010). *Scrum and CMMI Level 5: The Magic Potion for Code Warriors*. Retrieved from <http://www.scruminc.com/wp-content/uploads/2014/05/Scrum-and-CMMI-Level-5-A-Magic-Potion-for-Code-Warriors.pdf>
67. The Stationery Office. (2016). *ITIL Practitioner Guidance*. The Stationery Office.
68. Thimbleby, H. (1988). Delaying Commitment. *IEEE Software*(5(3)).
69. Trendowicz, A. (2014). *Software Project Effort Estimation - Foundations and Best Practice Guidelines for Success*. Springer.
70. VersionOne. (2018). *Annual State Of Agile Report*. VersionOne Inc.
71. Wiegers, K. E. (2003). *Software requirements (2nd edition)*. Washington: Microsoft Press.
72. Witten, I. H. (2011). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann.