

Podejście zwinne a tradycyjne do projektów wytwarzania oprogramowania

Waldemar Łabuda*

Warszawska Wyższa Szkoła Informatyki

Streszczenie

Artykuł stanowi próbę porównania podejścia tradycyjnego oraz zwinnego do projektów wytwarzania oprogramowania. Omówione zostały podstawowe pojęcia dotyczące kontekstu, w którym realizowane są projekty produkcji oprogramowania. Przedstawiono różnice pomiędzy tradycyjnym a zwinnym podejściem do projektów wytwarzania oprogramowania oraz krótko scharakteryzowano nowe metodyki zarządzania projektami wytwarzania oprogramowania. Następnie skupiono się na omówieniu podejścia adaptacyjnego do zarządzania projektami informatycznymi bazującego na Manifeście Agile. Scharakteryzowano zasady i praktyki oraz cykl życia projektu produkcji oprogramowania w podejściu adaptacyjnym. Artykuł zakończono omówieniem istoty podejścia zwinnego do zarządzania projektami wytwarzania oprogramowania na przykładzie metodyki ATERN.

Słowa kluczowe –podejście zwinne do projektów wytwarzania oprogramowania, metodyka ATERN.

* E-mail: w_labuda@poczta.wysi.edu.pl

1 Wprowadzenie

1.1 Dylematy ery e-gospodarki

Żyjemy w erze nowej gospodarki, zwanej e-gospodarką. Tradycyjne podejście do zarządzania projektami informatycznymi staje się mało przydatne w przypadku wytwarzania oprogramowania dla e-gospodarki, ponieważ nie radzi sobie ze zmiennością wymagań oraz prowadzi do długotrwałych projektów. W jaki zatem sposób dostarczyć klientowi biznesowemu oczekiwaną przez niego funkcjonalność? Jak poradzić sobie z nieustannymi zmianami wymagań? Odpowiedzi na te pytania zawiera niniejszy artykuł.

Zacznijmy od porównania podejścia tradycyjnego i nowego, tzw. zwinnego (ang. *agile*), do realizacji projektów wytwarzania oprogramowania dla e-gospodarki (tabela 1).

Tabela 1. Porównanie podejścia tradycyjnego z podejściem zwinnym do realizacji projektów wytwarzania oprogramowania dla e-gospodarki.

Podejście tradycyjne	Podejście zwinne
Stałe, szczegółowe plany	Płynne, ogólne plany
Procesy, zadania i czynności	Wyniki biznesowe, działający produkt
Planowanie i harmonogramowanie	Interakcje i zarządzanie wiedzą
Niska niepewność i ryzyko	Wysoka niepewność i ryzyko
Średnia lub mała szybkość wytwarzania	Duża szybkość wytwarzania
Średnie lub niewielkie zmiany	Bardzo duże lub duże zmiany
Jednorodne zespoły projektowe	Niejednorodne zespoły projektowe
Duże doświadczenie zespołów	Niewielkie doświadczenie zespołów
Skupione podejmowanie decyzji	Rozproszone podejmowanie decyzji

1.2 Przewidywalność a adaptacyjność

Kluczową sprawą jest zapewnienie zespołowi projektowemu informacji zwrotnej (sprzężenia zwrotnego), pozwalającej odpowiedzieć na pytanie, w jakim miejscu procesu

wytwarzania oprogramowania aktualnie się znajduje. Umożliwia to – stosowana od dawna – iteracja oraz przyrostowy cykl życia oprogramowania. Istotą procesu iteracyjnego jest częste dostarczanie działających fragmentów oprogramowania (kolejnych przyrostów), realizujących wybrane zbiory funkcji, które łącznie składają się na użyteczność finalnego produktu. Iteracyjny cykl wytwarzania oprogramowania prowadzi do takiego stylu zarządzania, w którym plany długoterminowe są płynne, podczas gdy stabilny plan można stworzyć na bardzo krótki czas (zazwyczaj jest to czas realizacji iteracji).

Iteracyjny i przyrostowy sposób wytwarzania oprogramowania prowadzi do zupełnie nowych relacji z klientem biznesowym. Jego wynikiem są również nowe zasady funkcjonowania zespołu projektowego.

1.3 Nowe metodyki zarządzania projektami informatycznymi

Koncepcja iteracyjnego i przyrostowego cyklu wytwarzania programowania, jako lekarstwa na dylematy ery e-gospodarki spowodowała powstanie nowych metodyk zarządzania projektami informatycznymi. Metodyki te, zwane zwinnymi, nie odcinają się zupełnie od zorientowanych na dokumenty, sformalizowanych metodyk tradycyjnych. Jednakże posiadają swoje specyficzne cechy [1]:

- są bardziej adaptacyjne niż predykcyjne – metodyki tradycyjne nie radzą sobie z częstymi zmianami wymagań, podczas gdy zwinne przyjmują je za zasadę (procesy metodyk zwinnych próbują się do tych częstych zmian wymagań adaptować),
- są zorientowane na ludzi, a nie na procesy (ich podstawą jest przyjęcie charakterystycznego dla człowieka twórczego stylu pracy oraz włączenie, jako elementów, wszelkiego rodzaju osobliwości kultury i zachowań członków zespołów projektowych).

W ostatnich latach pojawiła się duża liczba nowych, zwinnych metodyk zarządzania projektami wytwarzania oprogramowania. Najbardziej znanymi są [1]:

- **XP** (eXtreme Programming) – powstała z inspiracji Kenta Becka w środowisku informatyków z kręgu języka i środowiska programowania Smalltalk.
- **Crystal Clear Family** – koncepcja opracowana i popularyzowana przez Alistaira Cockburna, eksperta w dziedzinie projektowania obiektowego (uważa on za konieczne, by metodykę opracowywać dla każdej klasy projektu).
- **Adaptative Software Development** – rozbudowana metodyka adaptacyjna opracowana i opublikowana w książce pod tym samym tytułem przez Jima Highsmitha.
 - **SCRUM, ATERN (DSDM), APM**[†].

[†]ATERN i APM to zastrzeżone znaki towarowe dla produktów firm Dynamic Systems Development Method Limited (ATERN) i Project Management Institute (APM).

2 Podejście adaptacyjne do zarządzania projektami informatycznymi

2.1 Manifest Agile

Manifest Agile (pełna nazwa Manifest Zwinnego Wytwarzania Oprogramowania, oryginalne nazwy: Agile Manifesto, Manifesto for Agile Software Development) to deklaracja wspólnych zasad dla zwinnych metodyk tworzenia oprogramowania. Została opracowana na spotkaniu w lutym 2001 roku w Snowbird w USA, w którym uczestniczyli reprezentanci nowych metodyk tworzenia oprogramowania, będących alternatywą dla tradycyjnego podejścia opartego na modelu kaskadowym [1]. Zgodnie z jej ideą, w pracy nad wytwarzaniem oprogramowania ceni się szczególnie:

- poszczególnych ludzi i interakcje między nimi (bardziej niż procesy i narzędzia);
- działające produkty, czyli oprogramowanie (bardziej niż kompleksową dokumentację);
- współpracę z klientem (bardziej niż negocjowanie kontraktu);
- reagowanie na zmiany (bardziej niż trzymanie się planu).

Narzędzia są niezbędne do przyspieszenia wytwarzania oprogramowania i redukcji kosztów, kontrakty są kluczowe dla rozwoju relacji klient – dostawca, a dokumentacja wspomaga komunikację. Jednak bez utalentowanych i wykształconych ludzi, działających produktów (oprogramowania), bliskiego współdziałania z klientami biznesowymi i użytkownikami końcowymi oraz reagowania na częste zmiany, dostarczenie nowego, spełniającego oczekiwania klienta oprogramowania, nie będzie niemożliwe.

Autorzy manifestu zdefiniowali następujące zasady adaptacyjnego (zwinnego – oba określenia będą dalej używane zamiennie) podejścia do zarządzania projektami wytwarzania oprogramowania [1]:

- **Satysfakcja klienta** jest najwyższym priorytetem, a osiągnięta powinna być poprzez wczesne i ciągłe dostarczanie wartościowego oprogramowania.
- **Zmienność wymagań** dotyczy zarówno nowych, jak i wymagań zmieniających się podczas realizacji projektu (adaptacyjny proces wytwarzania oprogramowania jest w stanie dać sobie radę ze zmianami dla spełnienia oczekiwań klienta).
- **Częste dostarczanie działającego oprogramowania** w okresach od kilku do kilkunastu tygodni, przy czym preferowane są krótsze ramy czasowe.
- **Zbliżenie do biznesu** poprzez wspólną, codzienną pracę przedstawicieli biznesu i zespołu projektowego przez cały czas realizacji projektu.

- **Zdolni ludzie** – projekty wytwarzania oprogramowania powinny być realizowane przez zdolnych, zmotywowanych i wyszkolonych ludzi. Należy stworzyć im środowisko pracy i dać wsparcie, jakiego potrzebują, a przede wszystkim zaufać im.
- **Komunikacja**, jako cecha kluczowa, a jej najbardziej efektywna metoda to konwersacja twarzą-w-twarz.
- **Działający software**, ponieważ działające oprogramowanie jest najważniejszą miarą postępu realizacji prac w projekcie.
- **Trwale i ciągle wytwarzanie**, ponieważ adaptacyjność tworzenia oprogramowania polega na zdolności utrzymania odpowiedniego tempa pracy podczas realizacji projektu przez wszystkich członków zespołu projektowego.
- **Doskonałość techniczna** polegająca na wyzwaniu adaptacyjności tj. dostosowywania produktu (oprogramowania) do często zmieniających się wymagań klienta.
- **Prostota**, jako ciągle upraszczanie kodu i struktury produktu (oprogramowania).
- **Projektowanie poprzez wylanianie najlepszego, spośród rozważanych, rozwiązań**, a więc najlepsze rozwiązanie powstaje w wyniku synergicznej pracy samoorganizujących się zespołów projektowych.
- **Zdanie się na decyzje zespołu**, czyli przyjęcie, iż w regularnych interwałach cały zespół projektowy zastanawia się, jak być bardziej efektywnym (wydajnym) i następnie reguluje odpowiednio swoje postępowanie.

Zasady adaptacyjnego podejścia do zarządzania projektami wytwarzania oprogramowania, zdefiniowane w Manifeście Agile, można znaleźć we współczesnych metodykach zwinnych.

2.2 Zasady i praktyki adaptacyjnego (zwinnego) zarządzania projektami

Zasady adaptacyjnego zarządzania projektami wskazują zespołom projektowym kierunek, natomiast do faktycznej realizacji prac niezbędna jest konkretna praktyka. Struktura procesu i konkretne praktyki tworzą minimalne, elastyczne ramy strukturalne dla samoorganizujących się zespołów.

2.3 Cykl życia projektu w podejściu adaptacyjnym

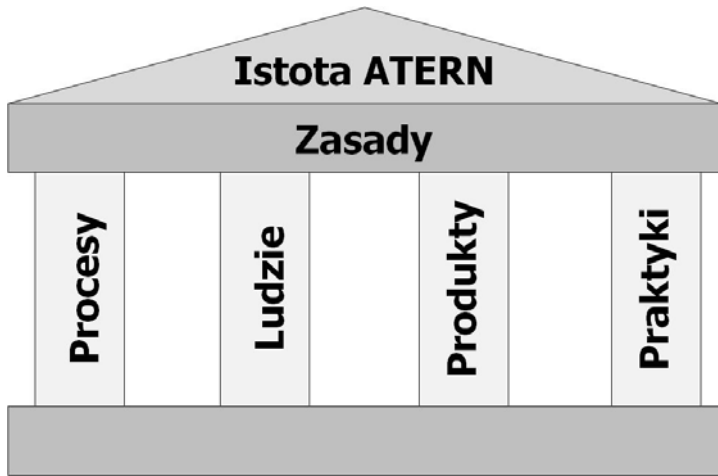
Cykl życia projektu, zgodny z Manifestem Agile, w ogólnym schemacie koncepcyjnym podejścia adaptacyjnego do zarządzania projektami powinien składać się z pięciu etapów [1]:

- **Faza tworzenia wizji** – w wyniku daje dobrze zdefiniowaną wizję produktu lub biznesu, wystarczającą do stworzenia właściwych ram dla pozostałych etapów.
- **Faza planowania adaptacyjnego** – tworzy się w niej hipotezy dotyczące specyfikacji produktu, wiedząc, że projekt spełniający specyfikacje technologiczne oraz specyfikacje klienta biznesowego będzie rozwijał się w miarę nabywania nowej wiedzy.
- **Faza eksploracji** – działania równoległe i iteracyjne, w których wstępne specyfikacje i rozwiązania projektowe są implementowane. Składniki określone, jako niepewne podlegają eksperymentom w większym stopniu, niż te pewniejsze.
- **Faza adaptacji** – wyniki eksperymentów (fazy eksploracji) podlegają przeglądom i ocenie od strony technicznej i strony uzasadnienia biznesowego projektu oraz z perspektywy klienta biznesowego. Zadania adaptacyjne będące wynikiem tej fazy stają się częścią kolejnej iteracji.
- **Faza zamknięcia** – działania umożliwiające zamknięcie projektu.

3 Istota podejścia zwinnego do realizacji projektów informatycznych na przykładzie metodyki ATERN

Zasady zwinnego podejścia do zarządzania projektami wytwarzania oprogramowania zawarte w Manifeście Agile można znaleźć w metodyce ATERN. Podstawowymi jej komponentami są: filozofia ATERN, zasady, procesy (cykl życia projektu), ludzie (role i odpowiedzialności), produkty (produkty zdefiniowane w każdej fazie cyklu życia projektu) oraz praktyki (rekomendowane techniki) [2]. Komponenty metodyki ATERN przedstawiono na rysunku 1 na następnej stronie.

Zgodnie z filozofią ATERN, projekt realizowany w tej metodyce musi posiadać uzgodnione i jasno zdefiniowane cele strategiczne i skupiać się na wczesnym dostarczaniu klientowi biznesowemu rzeczywistych korzyści. Kluczowi interesariusze projektu powinni rozumieć cele biznesowe na odpowiadających im poziomach zaangażowania i współdziałać dla dostarczenia rozwiązania. Rozwiązanie to powinno być dostarczone w uzgodnionym czasie zgodnie z priorytetami ustanowionymi przez biznes. Interesariusze powinni być przygotowani do dostarczania rozwiązań spełniających cele biznesowe, jak również muszą być przygotowani na ciągłe zmiany.

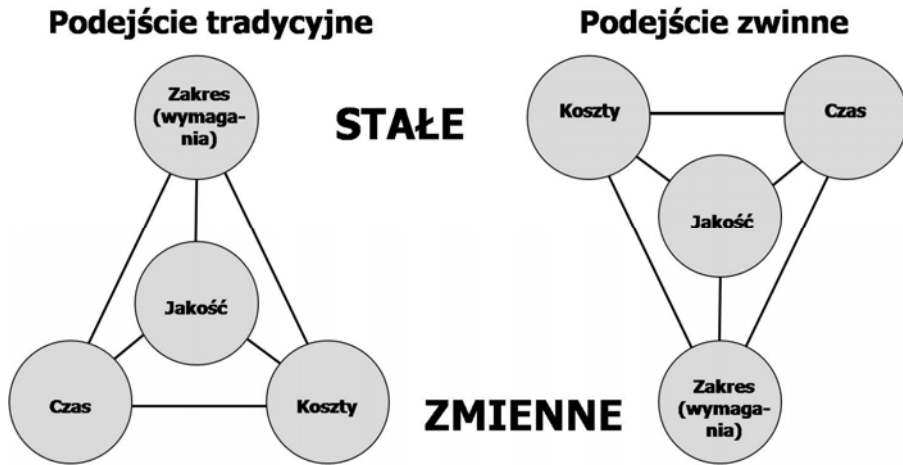


Rysunek 1. Komponenty metodyki ATERN [2]

Podstawowymi ograniczeniami (parametrami) projektów są: czas, koszty, zakres i jakość. Próba utrzymania ich stałych wartości może prowadzić do problemów, w tym porażek projektów. W podejściu tradycyjnym do zarządzania projektami (rysunek 2 na następnej stronie) zakres (wymagania) pozostaje stały, podczas gdy czas i koszty podlegają zmianom. Wynika do z faktu, że w podejściu tradycyjnym zakres projektu jest uzgodniony z klientem i występuje w postaci zatwierdzonej specyfikacji wymagań. Realizacja jej za wszelką cenę może, i zazwyczaj prowadzi do opóźnienia projektu i zwiększenia kosztów, jak również obniżenia jakości produktu końcowego projektu.

W podejściu zwinnym do zarządzania projektami za stałe uznawane są: czas, koszty i jakość, podczas gdy zakres (wymagania) podlega zmianom. Zmiany te realizowane są w cyklu życia projektu poprzez zastosowanie technik priorytetyzacji wymagań (metoda MoSCoW) oraz ram czasowych (ang. *Timeboxing*). Dlatego gwarantowany jest produkt końcowy projektu charakteryzujący się tzw. minimalnym użytecznym zbiorem wymagań (ang. *Minimum Useable Subset*) w uzgodnionym czasie i budżecie. Jakość jest stała, ponieważ kryteria akceptacji produktu końcowego projektu zostały uzgodnione i zatwierdzone przez klienta biznesowego we wczesnych fazach cyklu życia projektu tak samo, jak czas i budżet.

Porównanie podejścia tradycyjnego oraz „zwinnego” do realizacji projektów wytwarzania oprogramowania na przykładzie metodyki ATERN przedstawiono na rysunku 2 na następnej stronie.



Rysunek 2. Porównanie podejścia tradycyjnego oraz zwinnego do realizacji projektów wytwarzania oprogramowania na przykładzie metodyki ATERN [2]

3.1 Zasady podejścia zwinnego według ATERN

Osiem zasad ATERN organizuje filozofię zarządzania projektem. Pominięcie przez zespół projektowy, choć jednej z nich osłabia tę filozofię i uniemożliwia organizacji dostarczanie najlepszych rozwiązań biznesowych.

Zasady te są następujące [3]:

- **Skupienie na potrzebach biznesowych.** Każda decyzja podejmowana podczas realizacji projektu powinna być zgodna z jego celami biznesowymi. Zespół powinien skoncentrować się na identyfikacji priorytetów biznesowych, przygotowaniu uzasadnienia biznesowego, ciągłym zaangażowaniu biznesu, zagwarantowaniu implementacji minimalnego, użytecznego zbioru wymagań. Podstawowymi technikami realizacji tej zasady są priorytetyzacja (metoda MoSCoW) oraz ramy czasowe.
- **Dostawy na czas.** Mają one bezpośredni wpływ na sukces projektu. Wszyscy muszą korzystać z techniki ram czasowych, skupiać się na priorytetach biznesowych (wymagania), zawsze dotrzymywać uzgodnionych terminów. Podstawową techniką niezbędną dla realizacji tej zasady jest wysuwanie na pierwszy plan (priorytetyzacja).
- **Współpraca.** Zespół pracuje w duchu aktywnej współpracy i zaangażowania. Dla realizacji tej zasady należy: włączyć do pracy w odpowiednim czasie odpowiednich interesariuszy, zapewnić członkom zespołu prawo

podjmowania decyzji w imieniu tych, których reprezentują, aktywnie włączać osoby reprezentujące biznes, budować kulturę zespołu. Podstawową techniką realizacji tej zasady jest warsztat ułatwiający (ang. *Facilitated Workshop*) interesariuszom efektywne dzielenie się wiedzą i doświadczeniem.

- **Brak kompromisów dotyczących jakości.** Poziom jakości wytwarzanego produktu uzgadniany jest na początku projektu i nie ulega zmianie podczas jego realizacji. Dla wprowadzenia w życie tej zasady zespół powinien zapewnić niezmiennosć jakości podczas realizacji projektu, projektować, dokumentować i testować (odpowiednio), zapewniać jakość w cyklu życia projektu poprzez regularne przeglądy, testować rozwiązanie wcześniej i w sposób ciągły (testowanie biznesowe i techniczne). Podstawowymi technikami realizacji tej zasady są przyznawanie pierwszeństwa (metoda MoSCoW) oraz ramy czasowe, sprawiające, iż testowanie nie jest obciążone niepotrzebnym ryzykiem.
- **Budowa przyrostowa na solidnym fundamencie firmowym.** Dla realizacji tej zasady zespół powinien dostarczać korzyści biznesowe najwcześniej, jak to tylko możliwe, w sposób ciągły potwierdzać prawidłowość rozwiązania podczas jego wytwarzania, formalnie ponownie szacować priorytety i wykonalność z każdym dostarczonym przyrostem. Do implementacji tej zasady zespół wykorzystuje cykl życia projektu w metodyce ATERN (fazy Feasibility oraz Foundations).
- **Rozwój iteracyjny.** Dla dostarczenia rozwiązania spełniającego wymagania biznesowe metodyka ATERN korzysta z iteracyjnego wytwarzania produktów projektu. Koncepcja iteracji jest wbudowana w cykl życia projektu, aż do najniższego poziomu ram czasowych. Dla realizacji tej zasady zespół powinien projektować w sposób wystarczający do wytwarzania produktów, stosować podejście iteracyjne do wytwarzania wszystkich produktów, korzystać z informacji zwrotnej od klienta w każdej iteracji, akceptować wyłanianie się bardziej szczegółowych rozwiązań, później niż wcześniej uwzględniać zmiany, ponieważ prawdziwe rozwiązanie nie powstaje bez nich (według ATERN zmiany są naturalne, nieuniknione i przyczyniają się do ewolucji rozwiązania), być kreatywnym, eksperymentować, uczyć się i rozwijać.
- **Ciągła i jasna komunikacja.** Brak komunikacji jest często przyczyną niepowodzeń projektów. Metodyka ATERN dostarcza technik, które zwiększają efektywność komunikacji zarówno wewnątrz, jak i na zewnątrz projektu. Dla realizacji tej zasady zespół powinien przeprowadzać codzienne

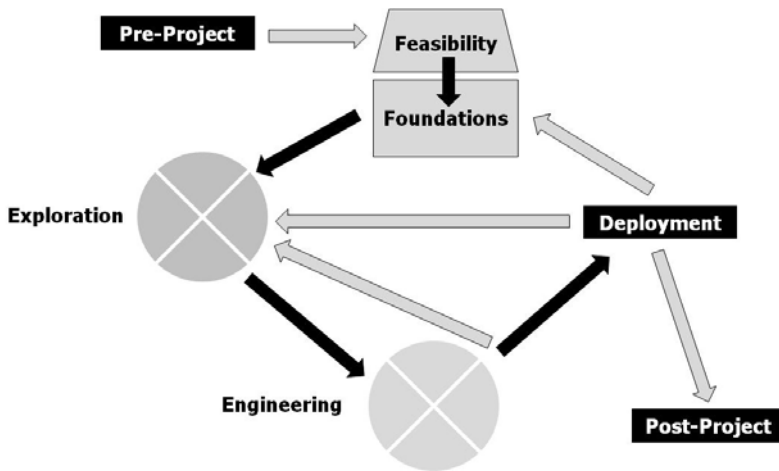
sesje na stojąco (ang. *Team Stand-up Session*), uczestniczyć w warsztatach ułatwiających, korzystać z modelowania i prototypowania, jako technik komunikacji, prezentować elementy rozwijanego rozwiązania wcześniej i często, utrzymywać odpowiednią, niezbyt obszerną dokumentację i na czas, zarządzać oczekiwaniami interesariuszy w cyklu życia projektu, popierać nieformalną komunikację twarzą-w-twarz na wszystkich jej poziomach. Metodyka ATERN podkreśla ogromną wartość interakcji ludzkich poprzez korzystanie z wyżej wymienionych technik, jak również dokładnie zdefiniowanych ról i odpowiedzialności.

- **Demonstrowanie kontroli.** Projekt powinien być pod kontrolą w każdej fazie swojego cyklu życia. Zespół projektowy powinien być proaktywny szczególnie podczas monitorowania i kontroli postępu prac w projekcie w odniesieniu do produktów fazy Foundations, zwłaszcza uzasadnienia biznesowego. Dla realizacji tej zasady zespół powinien korzystać z odpowiedniego poziomu formalizacji dla śledzenia i raportowania postępu prac, starać się, by plany i postęp prac był widoczny dla wszystkich interesariuszy, mierzyć postęp poprzez skupienie się bardziej na dostarczaniu produktów, niż wykonanych działaniach, zarządzać proaktywnie, szacować kontynuowane prace i możliwości ich wykonania na bazie celów biznesowych. Dobrze zdefiniowane ramy czasowe posiadające zdefiniowane na stałe punkty przeglądów, jak również przygotowanie planów w fazie Foundations oraz planów ram czasowych (ang. *Timebox Plans*) powinny ułatwić zespołowi projektowemu realizację niniejszej zasady.

3.2 Przykładowy cykl życia projektu w metodyce ATERN

Zgodnie z powyższymi zasadami, cykl życia projektu w metodyce ATERN jest iteracyjny i przyrostowy. Rozwiązanie jest dostarczane klientowi biznesowemu w seriach przyrostów, które powodują ciągły wzrost funkcjonalności dostarczanego rozwiązania. Pilne potrzeby klienta biznesowego dotyczące wymagań powinny być realizowane wcześniej, podczas gdy mniej ważne – później. Iteracyjna natura tej metodyki daje możliwość uczestnictwa przedstawicielom klienta w zespole projektowym podczas konstrukcji rozwiązania. Mogą oni na bieżąco komentować powstające rozwiązanie, zgłaszać wnioski o wprowadzenie zmian podczas realizacji iteracji w danym przyroście.

Metodyka ATERN integruje cykl życia projektu z cyklem życia rozwijanego produktu, co pokazano przykładowo na rysunku 3.



Rysunek 3. Przykładowy cykl życia projektu w metodyce ATERN [3]

3.2.1 Faza Pre-Project

W wielu organizacjach projekty są elementami programów lub portfeli projektów. Faza Pre-Project formalizuje propozycje projektów i umieszcza je w kontekście innych, aktualnie wykonywanych lub planowanych do wykonania prac w organizacji. Jej podstawowymi celami są [3]:

- przygotowanie opisu problemu biznesowego,
- identyfikacja ról Business Sponsor oraz Business Visionary,
- potwierdzenie, że projekt jest zgodny ze strategią biznesową organizacji,
- przygotowanie zakresu, planu i zasobów dla fazy Feasibility.

3.2.2 Faza Feasibility

Dobre praktyki zarządzania projektem powinny zapewniać osiągnięcie wraz z produktem końcowym prognozowanych korzyści. Faza Feasibility umożliwia podjęcie decyzji, czy proponowany projekt jest możliwy do wykonania z perspektywy biznesowej i technicznej, przy uwzględnieniu zagregowanego (wysokopoziomowego) dochodzenia do finalnego rozwiązania, jak również kosztów i ram czasowych.

Podstawowe cele fazy Feasibility to [3]:

- ustalenie, czy istnieje możliwe do wykonania rozwiązanie problemu biznesowego zdefiniowanego w Terms of Reference podczas fazy Pre-Project,
- identyfikacja korzyści, jakie zostaną osiągnięte w wyniku dostarczenia rozwiązania,
- zarys możliwych podejść do rozwiązania, z uwzględnieniem strategii jego obsługi oraz sposobu zarządzania projektem,

- opis organizacji oraz aspektów nadzoru nad projektem,
- wstępne oszacowanie czasu oraz kosztów całkowitych projektu,
- przygotowanie planu oraz zasobów dla fazy Foundations.

Warunkami wstępnymi niezbędnymi do realizacji fazy Feasibility są:

- zatwierdzenie produktu biznesowego Terms of Reference dla projektu,
- dostępność wymaganych zasobów dla przeprowadzenia studium wykonalności,
- rola Business Visionary dysponuje wystarczającym czasem do zdefiniowania wizji biznesowej projektu.

3.2.3 Faza Foundations

Celem fazy Foundations jest ustalenie trwałych, firmowych fundamentów dla projektu. Dla zdefiniowania jasnego i elastycznego projektu należy połączyć perspektywę biznesu, rozwiązania i zarządzania.

Podstawowymi celami fazy Foundations są [3]:

- specyfikacja bazowych wymagań wysokiego poziomu dla projektu i opis ich priorytetów oraz związków z potrzebami biznesowymi,
- opis procesów biznesowych obsługiwanych przez proponowane rozwiązanie (opcjonalnie),
- identyfikacja informacji tworzonych i aktualizowanych przez proponowane rozwiązanie,
- opis strategii dla wszystkich aspektów wdrożenia proponowanego rozwiązania,
- uszczegółowienie uzasadnienia biznesowego dla projektu,
- uruchomienie projektowania architektury rozwiązania oraz identyfikacja jego fizycznych i infrastrukturalnych elementów,
- definiowanie technicznych standardów implementacji,
- odpowiedź na pytanie, w jaki sposób zapewniona będzie jakość,
- ustalenie odpowiedniego nadzoru oraz organizacji dla projektu,
- opis cyklu życia rozwijanego rozwiązania poprzez zastosowanie technik zarządzania projektem, jak również metod demonstrowania i komunikowania postępu prac,
- harmonogram bazy dla rozwoju oraz wdrożenia rozwiązania,
- identyfikacja, oszacowanie i zarządzanie ryzykiem związanym z projektem.

Warunkiem wstępnym, niezbędnym do realizacji fazy Foundations jest zgoda na Feasibility Assessment (jeśli taki dokument został utworzony).

3.2.4 Faza Exploration

Faza Exploration razem z fazą Engineering jest używana do iteracyjnego i przyrostowego dochodzenia do szczegółowych wymagań biznesowych oraz transferu ich na wykonalne rozwiązanie. Od wstępnego rozwiązania tworzonego w fazie Exploration nie oczekuje się, żeby było gotowe do produkcji. Zespół skupia się na zdemonstrowaniu tego, że dostarczy, co potrzeba, i co jest zgodne z ogólnymi potrzebami biznesowymi. Produkt końcowy z fazy Exploration będzie doskonały dalej podczas fazy Engineering dla zapewnienia akceptacji kryteriów takich, jak: wydajność (ang. *performance*), pojemność (ang. *capacity*), bezpieczeństwo (ang. *security*), zdolność do obsługi (ang. *supportability*) oraz zdolność do utrzymania (ang. *maintainability*). Podstawowymi celami fazy Exploration są [3]:

- opracowanie bazowych wymagań zebranych i zapisanych w dokumencie biznesowym PRL (ang. *Prioritised Requirement List*) utworzonym podczas fazy Foundations,
- bardzo dokładne zbadanie potrzeb biznesowych i dostarczenie szczegółowych wymagań dla rozwijanego rozwiązania,
- tworzenie funkcjonalnego rozwiązania, które demonstruje spełnienie potrzeb biznesu,
- wczesne przekazanie organizacji rozwiązania, z którego, być może, będzie ona korzystać, obsługiwać oraz utrzymywać,
- (jeśli potrzeba) ewolucja produktu technicznego Solution Foundations (składniki: Business Area Definition i System Architecture Definition) pochodzącego z fazy Foundations w modelu, który opisuje, w jaki sposób rozwiązanie pracuje i w jaki sposób rozwiązanie wpływa na procesy i systemy biznesowe.

3.2.5 Faza Engineering

W fazie tej, traktowanej łącznie z fazą Exploration, następuje iteracyjny i przyrostowy rozwój, osiągniętego w fazie Exploration, wstępnego rozwiązania w celu przygotowania go do wdrożenia. Wysiłek zespołu projektowego jest skupiony głównie na adresowaniu wymagań poza funkcjonalnych. Dodatkowo zaangażowanie przedstawicieli biznesu dostarcza ciągłej walidacji celów biznesowych z perspektywy funkcjonalnej.

Podstawowymi celami fazy Engineering są [3]:

- szczegółowa analiza produktu technicznego Evolving Solution z fazy Exploration w celu zapewnienia uzgodnionych kryteriów akceptacji,
- rozszerzenie i analiza każdego produktu wymaganego do właściwego działania oraz obsługi w działaniu.

Warunkami wstępnymi, niezbędnymi do realizacji fazy Engineering są:

- produkt techniczny (dokumentacja) Evolving Solution z fazy Exploration został zatwierdzony (rola Business Visionary potwierdza, że cechy produktu demonstrowane w Evolving Solution są zgodne z wizją finalnego rozwiązania biznesowego),
- środowisko (fizyczne i techniczne) jest obecne i odpowiednie dla rozwoju rozwiązania,
- wymagany personel i interesariusze są zaangażowani (jeśli to wymagane).

3.2.6 Faza Deployment

Podstawowym celem fazy Deployment jest wytworzenie produktu, który może być wdrożony u klienta. Jeśli jest on sprzedawany i przekazywany przez wykonawcę na zewnątrz, wtedy faza Deployment jest wykorzystywana do przygotowania produktu do takiego przekazania. Poza tym gra rolę kluczowego punktu przeglądu (recenzji) produktu do wdrożenia lub jego ewentualnego przyszłego rozwoju.

Podstawowymi celami fazy Deployment są [3]:

- potwierdzenie ciągłej wydajności i zdolności do życia projektu oraz dokonanie aktualizacji planów (jeśli wymagane),
- wdrożenie rozwiązania (lub jego kolejnego przyrostu) w środowisku biznesowym,
- szkolenia użytkowników końcowych (jeśli wymagane) i dostarczenie niezbędnej dokumentacji do obsługi rozwiązania w środowisku biznesowym,
- szkolenia lub dostarczenie niezbędnej dokumentacji dla personelu operacyjnej obsługi (odpowiedzialni za techniczne aspekty obsługi i utrzymania rozwiązania),
- oszacowanie, czy wdrożone rozwiązanie jest w stanie dostarczać korzyści zapisane w uzasadnieniu biznesowym,
- po końcowym wdrożeniu: formalne zamknięcie projektu, zdiagnozowanie wykonania całego projektu z punktu widzenia technicznego i procesowego oraz przegląd wykonania projektu z punktu widzenia biznesowego.

Warunkiem wstępnym wymaganym do realizacji fazy Deployment jest zatwierdzenie do wdrożenia produktu technicznego Deployed Solution.

3.2.7 Faza Post-Project

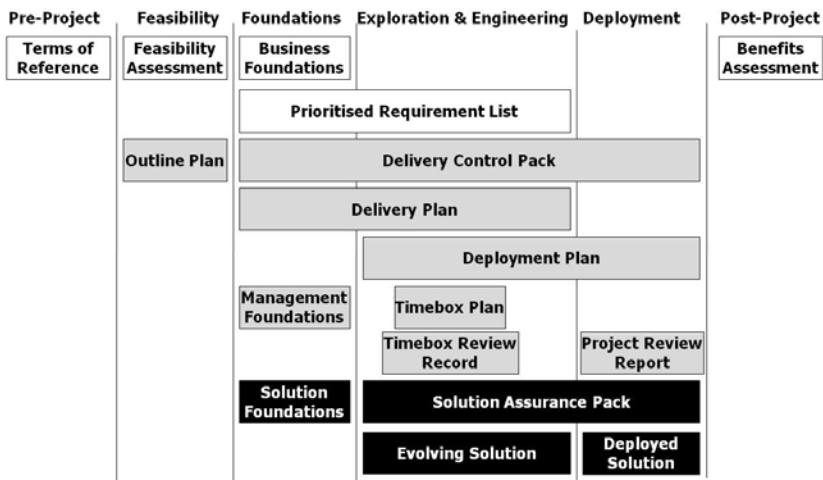
Faza Post-Project ma miejsce po ostatnim planowanym wdrożeniu rozwiązania. Jej celem jest zastanowienie się nad wydajnością projektu w warunkach aktualnie osiągniętych wartości biznesowych. Oszacowanie to powinno być przeprowadzone najwcześniej, jak to tylko możliwe, zwykle trzy do sześciu miesięcy po zamknięciu

projektu. Podstawowym celem fazy Post-Project jest określenie, czy korzyści zapisane w uzasadnieniu biznesowym zostały osiągnięte poprzez użytkowanie wdrożonego rozwiązania [3]. Warunkiem wstępnym, wymaganym do realizacji fazy Post-Project, jest pomyślne wdrożenie rozwiązania u klienta.

3.3 Podstawowe produkty w cyklu życia projektu w metodyce ATERN

W każdej fazie cyklu tworzenia projektu w metodyce ATERN zdefiniowane są produkty. Nie wszystkie z nich są wymagane w każdym projekcie, a formalizm związany z każdym z nich zależy od projektu oraz organizacji, w której projekt jest realizowany. Niektóre produkty są charakterystyczne dla konkretnej fazy cyklu życia projektu, pozostałe są rozwijane podczas wielu faz.

Produkt biznesowy Feasibility Assessment, powstały w fazie Feasibility, jest uszczegółowiony w fazie Foundations do Business Foundations. Produkt Business Foundations stanowi podstawę do zdefiniowania w fazach Exploration & Engineering produktu biznesowego PRL (ang. *Prioritised Requirement List*). Podobnie informacje zawarte w produkcie zarządczym Outline Plan, powstałym w fazie Feasibility, są niezbędne do stworzenia w fazach Foundation oraz Exploration & Engineering produktu Delivery Plan. Produkt Delivery Plan stanowi podstawę do tworzenia planów ram czasowych (ang. *Timebox Plans*) i planów wdrożenia kolejnych przyrostów (ang. *Deployment Plans*), co przedstawia rysunek 4.



Rysunek 4. Podstawowe produkty w cyklu życia projektu w metodyce ATERN [3]. Kolor biały – produkty biznesowe, kolor szary – produkty zarządcze, kolor czarny – produkty techniczne.

3.3.1 Faza Pre-Project

Produktem fazy Pre-Project jest produkt biznesowy Terms of Reference. Zawiera zazwyczaj definicję celów biznesowych projektu na wysokim poziomie agregacji. Powinien zawierać krótki opis (zarys) potrzeb biznesowych, definicję celów biznesowych i zakresu projektu dla spełnienia tych potrzeb.

3.3.2 Faza Feasibility

Produktami fazy Feasibility są produkt biznesowy Feasibility Assessment oraz produkt zarządczy Outline Plan.

Feasibility Assessment (wizja biznesowa) to przegląd projektu na wysokim poziomie agregacji (perspektywa biznesowa i techniczna). Zawiera identyfikację i opis ryzyka, jak również propozycję strategii jego redukcji. Powinien składać się z następujących elementów: zarys uzasadnienia biznesowego (ang. *Outline Business Case*), szkic proponowanego rozwiązania (ang. *Outline Solution*) oraz, opcjonalnie, prototyp pokazujący możliwości wykonania proponowanego rozwiązania (ang. *Feasibility Prototype*).

Outline Plan powstaje na podstawie odpowiedzi uzyskanych w PAQ (ang. *Project Approach Questionnaire*). Dostarcza dokładnego planu prac, które mają być wykonane w fazie Foundations. Powinien składać się z następujących elementów: zarys prawdopodobnego podejścia do projektu (na podstawie PAQ), prawdopodobny profil zasobów niezbędnych do wykonania prac w projekcie, opis narzędzi wymaganych do obsługi procesu iteracyjnego wytwarzania, struktura organizacyjna projektu i procesy niezbędne do jego nadzoru, analiza ryzyka i problemów wpływających na wykonalność projektu, zarys harmonogramu projektu (harmonogram kamieni milowych dotyczących dostarczania i wdrażania rozwiązania), szczegółowy plan dla fazy Foundations.

3.3.3 Faza Foundations

W fazie tej tworzone są: produkt biznesowy Business Foundations, produkt zarządczy Management Foundations oraz produkt techniczny Solution Foundations. Rozpoczynają się tutaj również prace nad produktem biznesowym PRL (ang. *Prioritised Requirement List*) oraz produktami zarządczymi Delivery Control Pack i Delivery Plan.

Produkt biznesowy Business Foundations dostarcza informacji biznesowej niezbędnej do sukcesu projektu (powinna być zrozumiana przez wszystkich interesariuszy przed rozpoczęciem prac nad rozwojem produktu). Powstaje poprzez rozwój wizji biznesowej zapisanej w Feasibility Assessment (faza Feasibility). Jeśli projekt nie jest elementem programu, czy portfela, wtedy zawiera uzasadnienie biznesowe

projektu (doprecyzowanie zarysu uzasadnienia biznesowego zawartego w Feasibility Assessment).

PRL opisuje wymagania dotyczące produktu, jakie powinny być osiągnięte, aby uzasadnienie biznesowe zostało spełnione. Jest to bazowy zbiór wymagań na wysokim poziomie agregacji, który powstaje na koniec fazy Foundations i uszczegóławiany jest w fazach Exploration oraz Engineering (iteracyjny rozwój produktu) z uwzględnieniem wprowadzanych zmian.

Solution Foundations dostarcza informacji technicznych na temat rozwiązania (podstawa sukcesu projektu). Powinien być zrozumiany przez wszystkich wewnętrznych interesariuszy projektu przed rozpoczęciem prac na produktem projektu. Ma zawierać następujące elementy: BAD (ang. *Business Area Definition*), SAD (ang. *System Architecture Definition*), DAD (ang. *Development Approach Definition*) oraz Solution Prototype (opcjonalnie).

Management Foundations opisuje zasadnicze aspekty organizacyjne i kontrolne projektu, jak również precyzyjnie definiuje, w jaki sposób będzie on zarządzany. Poza tym zawiera opis sposobu wykorzystania praktyk ATERN niezbędnych do zarządzania projektem. Przeglądane i rozważane są odpowiedzi uzyskane w Project Approach Questionnaire (faza Feasibility). Produkt ten stanowi rozwinięcie i doprecyzowanie dokumentu zarządczego Outline Plan z fazy Feasibility.

Delivery Plan zawiera opracowany i uszczegółowiony harmonogram opisany w produkcie zarządczym Outline Plan (faza Feasibility). Harmonogram ram czasowych oraz innych działań niezbędnych dla dostarczenia produktów projektu dotyczy wszystkich aspektów ich rozwoju i wdrożenia (przyrostowa natura projektu).

Delivery Control Pack mieści w sobie raporty, dokumenty i rejestry (logi) dotyczące ciągłego monitorowania statusu projektu. Podstawowymi jego elementami są: Risk Log (śledzenie różnych rodzajów ryzyka oraz reagowanie na nie), Change Control Records (śledzenie zmian wprowadzanych do zakresu projektu), Periodic Reports (raporty dotyczące postępu prac w projekcie), Issues Log (rejestr problemów), Communication Log (rejestr komunikacji), Burn-Down Chart (wykres „spalania”) oraz Project Dashboard (tablica projektu).

3.3.4 Fazy Exploration & Engineering

Rezultatami faz Exploration oraz Engineering są: produkty zarządcze Timebox Plan i Timebox Review Record oraz produkt techniczny Evolving Solution. W fazach tych kontynuowane są prace nad produktem biznesowym PRL oraz produktami zarządczymi Delivery Control Pack i Delivery Plan, jak również rozpoczyna się praca nad produktem zarządczym Deployment Plan i produktem technicznym Solution Assurance Pack.

Kompozycja i natura Evolving Solution jest wyłącznie zależna od celów projektu oraz fazy realizacji projektu. Na początku może być wstępnym szkicem procesu biznesowego wykonanym na tablicy flip chart. Na końcu projektu może być to w pełni rozwinięty i udokumentowany proces biznesowy obsługiwany przez oprogramowanie oraz całą używaną dokumentację niezbędną do jego utrzymania. Na bazie Evolving Solution tworzony jest w fazie Deployment produkt techniczny Deployed Solution.

Solution Assurance Pack jest zbiorem elementów potwierdzających kompletność komponentów zawartych w Evolving Solution i zawiera: Solution Review Record, Business Testing Pack oraz Technical Testing Pack.

Deployment Plan (czasami składnik Delivery Plan) jest szczegółowym planem dla fazy Deployment. Zawiera opis wszystkich aspektów działań niezbędnych do dostarczenia produktów projektu z perspektywy biznesowej i technicznej. Działania biznesowe mogą zawierać opisy, jak będą implementowane nowe procesy biznesowe, jak nowa struktura organizacyjna będzie skomunikowana oraz mechanizmy i logikę związaną z edukacją dotyczącą wpływu każdej zmiany biznesowej. Z kolei działania techniczne mogą zawierać opis, jak wykorzystywany będzie sprzęt i oprogramowanie. Komponentem tego produktu jest Benefits Realisation Plan definiujący i dostarczający harmonogramu działań dla osiągnięcia korzyści zapisanych w uzasadnieniu biznesowym a osiągniętych poprzez wdrożenie rozwiązania (Deployed Solution).

Timebox Plan zawiera opracowanie celów biznesowych realizowanych w każdym ramieniu czasowym zapisanych w Delivery Plan. Powinien zawierać: definicję produktu (produktów) dla każdego ramienia czasowego, określenie kamieni milowych definiujących np. daty przeglądów technicznych lub przeglądów dokonywanych przez użytkowników w każdym ramieniu czasowym, uzgodnione priorytety MoSCoW dla produktów i działań w ramach czasowych, definicję wszystkich zasobów niezbędnych dla wykonania prac.

Timebox Review Record jest wytwarzany lub aktualizowany w punktach przeglądów w każdym ramieniu czasowym. Zawiera informację o tym, co zostało osiągnięte, a także informację zwrotną, która może wpłynąć na dalszą realizację danego ramienia czasowego. Po zakończeniu realizacji ramienia czasowego wszystkie nierozwiązane problemy są rozważane w kontekście Delivery Plan oraz przyszłych Timebox Plans. Formalnie Timebox Review Record musi istnieć w jakiejś fizycznej formie. Generalnie powinien zawierać informację na temat tego, co zostało wykonane a co nie, biorąc pod uwagę Timebox Plan, jak również: zapisy formalnej akceptacji produktów przez przedstawicieli biznesu, oszacowanie priorytetów wszystkich, niewykonanych prac oraz planowany termin ich wykonania, oszacowanie efektywności

procesów kontroli ram czasowych oraz technik iteracyjnego rozwoju zgodnie z zasadami metodyki ATERN.

3.3.5 Faza Deployment

Produktami fazy Deployment są: produkt zarządczy Project Review Report, produkt techniczny Deployed Solution. W fazie tej kontynuowane są również prace nad produktami zarządczymi Delivery Control Pack, Deployment Plan, jak również produktem technicznym Solution Assurance Pack.

Deployed Solution ma zawierać w pełni zintegrowane i wzajemnie spójne komponenty finalnego rozwiązania, jak również materiały wymagane do jego obsługi.

Project Review Report jest produktem rozwijanym i aktualizowanym na końcu każdego przyrostu. Może zawierać jeden lub większą liczbę następujących elementów: Increment Review Record, Benefits Enablement Summary oraz End of Project Assessment.

3.3.6 Faza Post-Project

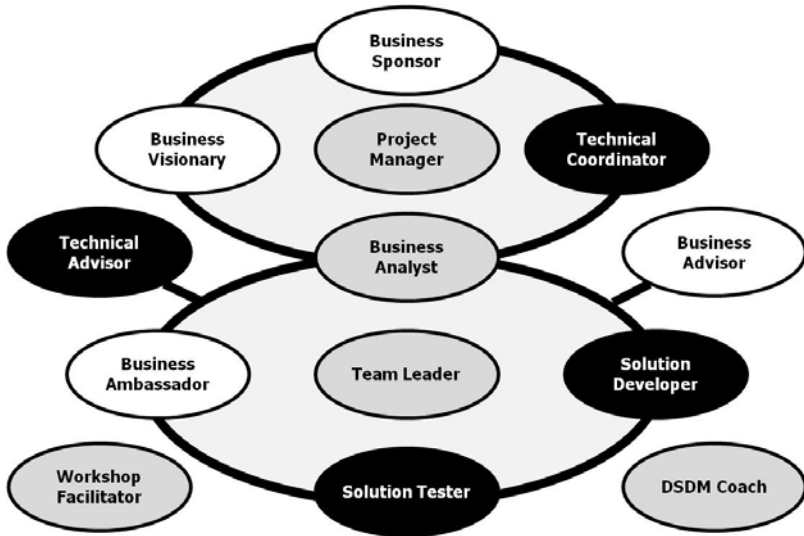
Produktem fazy Post-Project jest produkt biznesowy Benefits Assessment. Zawiera opis sposobu naliczania korzyści podczas korzystania z wdrożonego rozwiązania (Deployed Solution).

3.4 Role i odpowiedzialności w metodyce ATERN

Według metodyki ATERN, ludzie pracujący efektywnie razem są podstawą sukcesu projektu. Dlatego w strukturze organizacyjnej projektu wyróżniono role wraz z przypisanymi do nich obowiązkami – zarówno po stronie klienta, jak i dostawcy. W strukturze organizacyjnej projektu wskazuje się następujące poziomy:

- projektu (ang. *Project Level*),
- zespołu projektowego (ang. *Team Level*) oraz
- wsparcie (ang. *Others*).

Na poziomie projektu występują następujące role: Business Sponsor, Business Visionary, Business Analyst, Technical Coordinator oraz Project Manager. Z kolei na poziomie zespołu projektowego reprezentowane są role: Business Analyst, Business Ambassador, Solution Developer, Solution Tester oraz Team Leader. Wsparcie stanowią: Workshop Facilitator oraz DSDM Coach. Strukturę organizacyjną projektu realizowanego w metodyce ATERN przedstawiono na rysunku 5.



Rysunek 5. Podstawowe role w metodyce ATERN [3]. Kolor biały – role biznesowe, kolor szary – role zarządcze, kolor czarny – role techniczne.

Business Sponsor to najważniejsza – z punktu widzenia biznesowego – jednoosobowa rola w projekcie. Osoba ją pełniąca odpowiedzialna jest za proponowane podejście do dostarczenia produktu końcowego projektu, jak również za uzasadnienie biznesowe projektu w czasie jego trwania. Business Sponsor jest właścicielem dostarczonego w wyniku realizacji projektu produktu oraz odpowiada za osiągnięcie korzyści związanych z jego wdrożeniem. Musi posiadać wysoką pozycję w organizacji, ponieważ powinien być zdolny do rozwiązywania wszelkich problemów biznesowych. Poza tym:

- jest właścicielem uzasadnienia biznesowego projektu,
- zapewnia ciągłą zgodność projektu z uzasadnieniem biznesowym,
- ręczy za dostępność środków finansowych i innych zasobów,
- odpowiada za szybki i efektywny proces decyzyjny dla przekazywanych mu problemów,
- szybko reaguje na wszystkie pojawiające się problemy.

Business Visionary to osoba bardziej zaangażowana w projekt niż Business Sponsor. Jest odpowiedzialna za interpretację potrzeb biznesowych artykułowanych przez rolę Business Sponsor, przekazuje je do zespołu projektowego oraz zapewnia ich szczegółową reprezentację w uzasadnieniu biznesowym projektu. Pozostaje zaangażowana przez cały czas trwania projektu, dostarczając zespołowi projektowemu

kierunków strategicznych i gwarantując, że dostarczony produkt pozwoli osiągnąć korzyści zapisane w uzasadnieniu biznesowym projektu. Poza tym:

- jest właścicielem każdej zmiany biznesowej z perspektywy organizacyjnej,
- definiuje wizję biznesową projektu,
- komunikuje i promuje wizję biznesową projektu wśród wszystkich zainteresowanych,
- monitoruje postęp projektu z punktu widzenia realizacji wizji biznesowej,
- współpracuje z zespołem projektowym podczas zbierania wymagań, projektowania rozwiązania i uczestniczy w sesjach przeglądów,
- zatwierdza zmiany dotyczące wymagań zapisanych w PRL na wysokim poziomie agregacji,
- zapewnia współpracę interesariuszy projektu w obszarze biznesu,
- dba o zasoby biznesowe i ich dostępność w razie potrzeby,
- zapewnia transfer wizji biznesowej do praktyki roboczej,
- pełni rolę arbitra w przypadku nieporozumień pomiędzy członkami zespołu projektowego.

Project Manager jest odpowiedzialny za wszystkie aspekty dostawy produktu. Zapewnia zarządzanie zespołem projektowym na poziomie projektu i odpowiada za zarządzanie środowiskiem pracy, w którym tworzony jest produkt projektu. Poza tym koordynuje wszystkie aspekty zarządzania projektem na poziomie projektu. Zgodnie z metodyką ATERN, Project Manager pozostawia szczegółowe planowanie aktualnie realizowanych dostaw produktów projektu roli lidera zespołu (ang. *Team Leader*) oraz członkom zespołu projektowego (ang. *Solution Development Team*). Jego odpowiedzialność dotyczy całego cyklu życia projektu i obejmuje zarówno biznesowe, jak i techniczne aspekty dostaw produktów projektu. Poza tym:

- komunikuje się z kierownictwem najwyższego szczebla i nadzorem projektu (Business Sponsor, Komitet Sterujący itd.),
- zajmuje się planowaniem i przygotowywaniem harmonogramu na poziomie projektu, pozostawiając szczegóły liderowi zespołu i członkom zespołu projektowego,
- monitoruje postęp prac w projekcie, porównując go z planami bazowymi,
- zarządza ryzykiem i problemami w momencie ich pojawienia się, przekazując je w razie potrzeby do kierownictwa biznesowego lub zespołów technicznych,
- zarządza ogólną konfiguracją na poziomie projektu,
- motywuje zespół projektowy do osiągnięcia celów projektu,
- zarządza zaangażowaniem biznesu w zespołach projektowych,
- zapewnia dodatkowych specjalistów technicznych (jeśli potrzeba),
- obsługuje problemy przekazywane przez zespół projektowy,

- pełni rolę doradcy zespołu projektowego dla obsługi różnych sytuacji.

Technical Coordinator dba o spójność projektu z technicznego punktu widzenia i spełnianie wymagań technicznych standardów jakości. Wykonuje te same funkcje z perspektywy technicznej, które z punktu widzenia biznesowego realizuje Business Visionary. Poza tym:

- akceptuje i kontroluje techniczną architekturę rozwiązania,
- definiuje środowisko techniczne projektu,
- doradza i koordynuje aktywność techniczną każdego zespołu projektowego,
- identyfikuje, analizuje ryzyko techniczne i przekazuje przygotowaną analizę roli Project Manager,
- zapewnia spełnienie wymagań poza funkcjonalnych,
- dba o dotrzymanie standardów i stosowanie dobrych praktyk technicznych,
- kontroluje techniczną konfigurację produktu lub rozwiązania,
- zarządza technicznymi aspektami przekazania rozwiązania do użytkownika w środowisku biznesowym,
- rozwiązuje techniczne różnice pomiędzy członkami zespołu projektowego.

Team Leader wspólnie z członkami zespołu projektowego planuje i koordynuje wszystkie szczegółowe aspekty dostawy produktów projektu. Jest to bardziej rola lidera zespołu, niż rola zarządzająca. Dlatego osoba przypisana do tej roli powinna posiadać odpowiednie umiejętności „miękkie” (interpersonalne, przywódcze), ponieważ jej zadaniem jest przewodzenie zespołowi w cyklu życia projektu.

Business Ambassador pochodzi bezpośrednio z obszaru biznesowego i dostarcza informacji biznesowych z perspektywy osoby, która będzie korzystać z powstającego produktu projektu. Podejmuje wszystkie decyzje dotyczące dopasowania rozwiązania do uzgodnionych celów biznesowych. Pracując razem z członkami zespołu projektowego, Business Ambassador uczestniczy w ewolucji produktu (rozwiązania). Jest również odpowiedzialny za codzienną komunikację z osobami obsadzonymi w rolach na poziomie projektu oraz z przedstawicielami biznesu.

Business Analyst jest całkowicie zintegrowany z zespołem projektowym i skupiony na relacjach pomiędzy osobami pełniącymi role biznesowe i techniczne. Dbą o to, aby potrzeby biznesu były dokładnie analizowane i odzwierciedlone w informacjach przekazywanych zespołowi projektowemu niezbędnych dla wygenerowania produktu (rozwiązania).

Solution Developer interpretuje wymagania biznesowe i tłumaczy je na możliwe do wdrożenia rozwiązanie spełniające wymagania funkcjonalne i poza funkcjonalne. Osoba pełniąca tę rolę powinna być przypisana do projektu na cały czas jego realizacji, ponieważ w przeciwnym przypadku do ram czasowych projektu wprowadzane jest znaczące ryzyko.

Solution Tester jest całkowicie zintegrowany z zespołem projektowym i odpowiedzialny za testowanie powstającego produktu (rozwiązania) zgodnie z Technical Testing Strategy w całym cyklu życia projektu.

Business Advisor, współpracujący z rolą Business Ambassador, jest odpowiedzialny za dostarczanie specyficznych informacji dotyczących rozwoju i testowania produktu (rozwiązania). Może być zwykłym użytkownikiem lub beneficjentem produktu końcowego projektu i powinien dostarczać rad i wskazówek, które powstający produkt ma uwzględnić.

Technical Advisor obsługuje zespół projektowy dostarczając specyficznych informacji technicznych niezbędnych do operacyjnego zarządzania zmianą, operacyjnej obsługi, ciągłego utrzymania rozwiązania. Przekazywanymi przez tę rolę informacjami technicznymi są:

- wymagania, sesje projektowania i przeglądów,
- codzienne decyzje operacyjne,
- scenariusze operacyjne pomagające definiować i testować rozwiązanie,
- zapewnienie, że rozwiązanie rozwija się prawidłowo,
- operacyjna akceptacja testowania,
- rozwój dokumentacji,
- szkolenia zespołów operacyjnych.

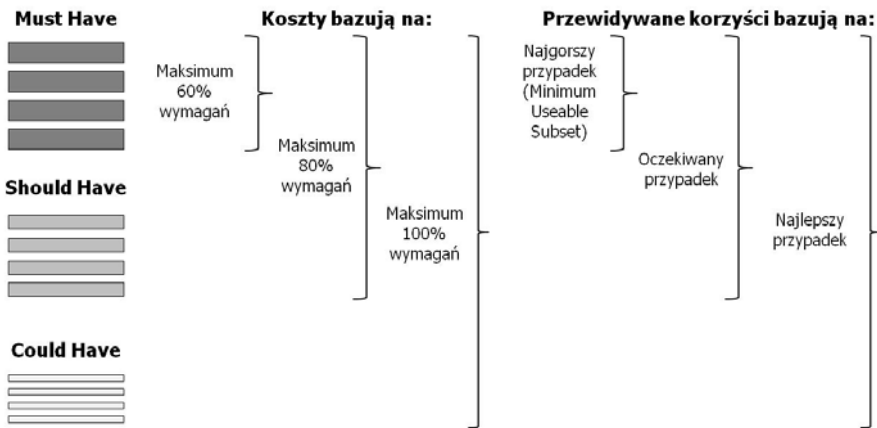
Workshop Facilitator zarządza warsztatami wspomagającymi zespoły projektowe, będąc jednocześnie katalizatorem dla ich przygotowania i komunikacji. Jest odpowiedzialny za kontekst warsztatu, a nie za jego zawartość. Powinien być niezależny od otoczenia projektowego, co pozwala na „świeże” spojrzenie, i co jest niezbędne do osiągnięcia celów warsztatów.

DSDM Coach jest kluczową rolą w przypadku, kiedy zespół projektowy nie ma doświadczenia dotyczącego metodyki ATERN. W takich przypadkach DSDM Coach powinien przekazać zespołowi projektowemu istotę podejścia ATERN odnosząc ją do kontekstu i ograniczeń w organizacji, w której realizowany jest projekt. Wymaga się, żeby DSDM Coach posiadał certyfikat z zakresu metodyki ATERN.

Kompetencje członków zespołu projektowego niejednokrotnie nie wystarczają do efektywnej realizacji projektu. Dlatego w metodyce ATERN istnieje możliwość zaalokowania do projektu dodatkowych osób dla wykonania prac technicznych. Decyzję o ich zaalokowaniu, na poziomie projektu podejmuje Project Manager, podczas gdy na poziomie zespołu projektowego Project Leader. Osoby te pełnią role specjalistów (ang. *Specialist*), które powinny być zintegrowane z rolami pełnionymi przez członków zespołu projektowego.

3.5 Technika priorytetyzacji MoSCoW

W metodyce ATERN technika priorytetyzacji może być zastosowana do: wymagań, zadań, produktów, studiów przypadków, opowieści użytkownika (ang. *User Stories*), kryteriów akceptacji i testów. MoSCoW jest techniką uniwersalną, umożliwiającą zrozumienie i definiowanie priorytetów. Użycie słów kluczowych Must, Should, Could oraz Won't Have dostarcza zespołowi projektowemu jasnych wskazówek dotyczących priorytetów. Technikę MoSCoW stosowaną w metodyce ATERN przedstawiono na rysunku 6.



Rysunek 6. Priorytetyzacja MoSCoW w metodyce ATERN [3]

Słowo kluczowe Must Have (ang. **MUST** – **Minimum Useable SubseT**) definiuje minimalny zbiór wymagań (wymagania funkcjonalne i poza funkcjonalne), które projekt gwarantuje dostarczyć w produkcie końcowym. Brak możliwości dostarczenia wymagań posiadających priorytet Must Have spowodowałby niespełnienie uzasadnienia biznesowego dla projektu, czyli niepowodzenie projektu. Zazwyczaj wymagania oznaczone priorytetem Must Have obejmują do 60% wszystkich wymagań zapisanych w PRL i definiują najgorszy z punktu widzenia biznesowego, ale dopuszczalny przypadek.

Wymagania oznaczone priorytetem Should Have są ważne z punktu widzenia biznesowego, ale nie są niezbędne dla funkcjonalności produktu końcowego projektu. Mogą być pominięte, a produkt końcowy nie straci uzgodnionej z klientem biznesowym funkcjonalności. Obejmują zazwyczaj do 20% wszystkich wymagań zapisanych w PRL i mogą być włączone do zakresu projektu dla utrzymania go na

ścieżce planu bazowego (zmienny zakres). Ich obejście może być trudne oraz kosztowne.

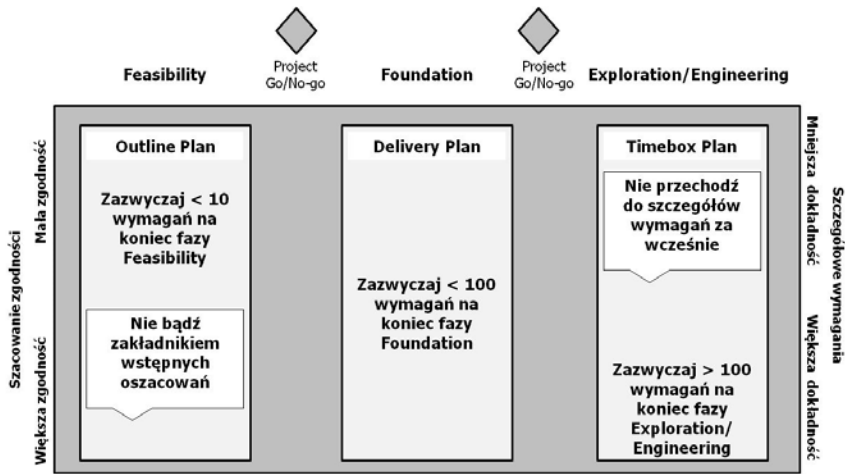
Wymagania oznaczone priorytetem *Could Have* są pożądane z punktu widzenia biznesowego, ale mniej ważne dla funkcjonalności produktu końcowego projektu. Ich pominięcie ma mniejszy wpływ na funkcjonalność produktu końcowego, niż pominięcie wymagań posiadających priorytet *Should Have*. Mogą być włączone do zakresu projektu bez powodowania problemów. Zazwyczaj stanowią do 20% wszystkich wymagań zapisanych w PRL. Ich obejście jest proste i tanie.

Wymagania oznaczone priorytetem *Won't Have*, zgodnie z uzgodnieniami zespołu projektowego z przedstawicielami biznesu, nie będą zaimplementowane w produkcie końcowym projektu (mogą być dostarczone następnym razem). Znajdują się zatem poza zakresem projektu w ustalonych, stałych ramach czasowych jego realizacji. Są zapisane w PRL dla zdefiniowania zakresu projektu i uniknięcia ponownego wprowadzania „tylnymi drzwiami” w późniejszym terminie. Ułatwiają zarządzanie oczekiwaniami biznesowymi, choć nie znajdują się w dostarczonym przez zespół projektowy produkcie końcowym projektu.

Najlepsze praktyki sugerują, że implementacja w produkcie końcowym projektu wymagań oznaczonych priorytetami *Must*, *Should* i *Could Have* (do 100% wymagań zapisanych w PRL) to najlepszy przypadek, podczas gdy dostarczenie w produkcie końcowym projektu jedynie wymagań oznaczonych priorytetami *Must* i *Should Have* (do 80% wymagań zapisanych w PRL) reprezentuje oczekiwany przez klienta biznesowego przypadek (rys. 6). Najgorszym, ale dopuszczalnym przypadkiem jest implementacja w produkcie końcowym projektu jedynie wymagań oznaczonych priorytetami *Must Have* (60% wymagań zapisanych w PRL).

Zastosowanie techniki wysuwania na pierwszy (priorytetyzacji) plan *MoSCoW* w metodyce *ATERN* wyjaśnia istotę podejścia zwinnego do zarządzania projektem. Podczas gdy czas, koszty i jakość są stałe, zmianie ulega zakres projektu. Reguła 60:20:20 stanowi tutaj wskazówkę. Uzgodnienie, jaki procent wymagań, kosztów i korzyści stanowi w danym projekcie, odpowiednio najgorszy, oczekiwany i najlepszy przypadek powinno być zapisane w uzasadnieniu biznesowym projektu (faza *Foundations*).

Na rysunku 7 przedstawiono ewolucję wymagań (funkcjonalne i poza funkcjonalne) w cyklu życia projektu w metodyce *ATERN*.



Rysunek 7. Ewolucja wymagań w cyklu życia projektu w metodyce ATERN [3]

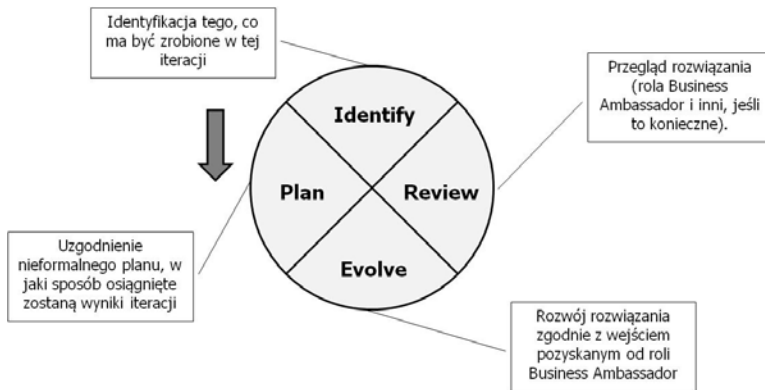
3.6 Technika rozwoju iteracyjnego

Rozwój iteracyjny to proces, w wyniku którego dostawa produktu wynika z powtarzanych cykli działań rozwojowych, gdzie każde uruchomienie cyklu zbliża funkcjonalność otrzymanego produktu częściowego do funkcjonalności produktu końcowego. Zwykle wymaga to od zespołu projektowego powtarzania następujących kroków:

- identyfikacja (ang. *Identify*) tego, co powinno być osiągnięte w danej iteracji,
- planowanie (ang. *Plan*), w jaki sposób osiągnięte zostaną wyniki iteracji,
- rozwój (ang. *Evolve*) produktu (rozwiązania) zgodnie z planem,
- przegląd (ang. *Review*) produktu (rozwiązania) poprzez określenie, czy wymagana jest jeszcze następna iteracja.

Ideę iteracyjnego rozwoju przedstawiono na rysunku 8 na następnej stronie.

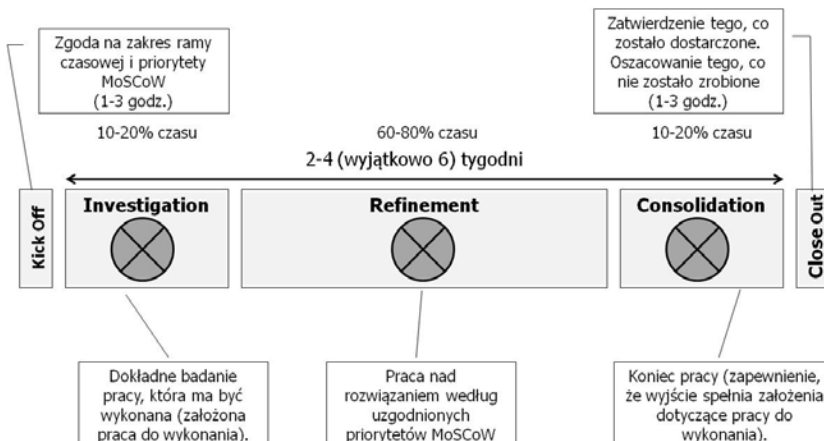
W metodyce ATERN każde uruchomienie cyklu pociąga za sobą analizę, projektowanie, budowę oraz przegląd rozwijającego się produktu (rozwiązania), zgodnie z technikami wymaganymi dla organizacji i zarządzania pracami w projekcie. Iteracyjny rozwój jest tu kluczową techniką: od idei (pomysłu) do produktu końcowego projektu. Proces iteracyjnego rozwoju musi być stale kontrolowany, a jego kontrola powinna być widoczna dla wszystkich zainteresowanych procesem lub rozwijanym produktem. Metodyka ATERN proponuje kontrolę poprzez cykl trzech iteracji, a mianowicie: Investigation, Refinement oraz Consolidation. Innymi słowy poprzez implementację kontroli ram czasowych do iteracyjnego rozwoju (ang. *Iterative Development*).



Rysunek 8. Iteracyjny rozwój w metodyce ATERN [3]

3.7 Technika ram czasowych

Ramy czasowe (ang. *Timeboxing*) są następną, kluczową techniką stosowaną w metodyce ATERN. Stanowią dobrze zdefiniowany proces dedykowany kontroli wytwarzania produktów na najniższym poziomie (poziom ramy czasowej) w cyklu iteracyjnym z kilkoma punktami przeglądów. Przeglądy pozwalają zapewnić jakość oraz wydajność wytwarzania produktów na najniższym poziomie. Poprzez dostarczanie na czas produktów na najniższym poziomie (poziom ramy czasowej), zapewnione jest terminowe wytwarzanie produktów na najwyższym poziomie (poziom projektu). Ideę ram czasowych przedstawiono na rysunku 9.



Rysunek 9. Idea ram czasowych (ang. *Timeboxing*) w metodyce ATERN [3]

Każdą ramę czasową rozpoczyna sesja wstępna (ang. *Kick-off meeting*) i kończy sesja zamykająca (ang. *Close-out meeting*). Rama czasowa składa się z trzech podstawowych etapów (iteracji), a więc: Investigation, Refinement oraz Consolidation odzwierciedlających cykl iteracyjnego rozwoju (ang. *Iterative Development*).

Sesja wstępna zespołu projektowego trwa krótko (1-3 godziny), jest niezbędna dla zrozumienia celów aktualnie rozpoczynanej ramy czasowej i zaakceptowania ich przez zespół projektowy, jako realistycznych, a więc możliwych do osiągnięcia w ramach rozpatrywanej ramy czasowej.

Investigation to wstępne badanie (10-20% czasu ramy czasowej) szczegółów wszystkich produktów, które mają być dostarczone na tym etapie, w tym uzgodnienie produktów oraz miar ilościowych, które pozwolą wykazać sukces jej realizacji.

Refinement to całkowity rozwój (wytworzenie) i testowanie uzgodnionych produktów ramy czasowej, zgodnie z ich ustalonymi priorytetami (60-80% czasu ramy czasowej).

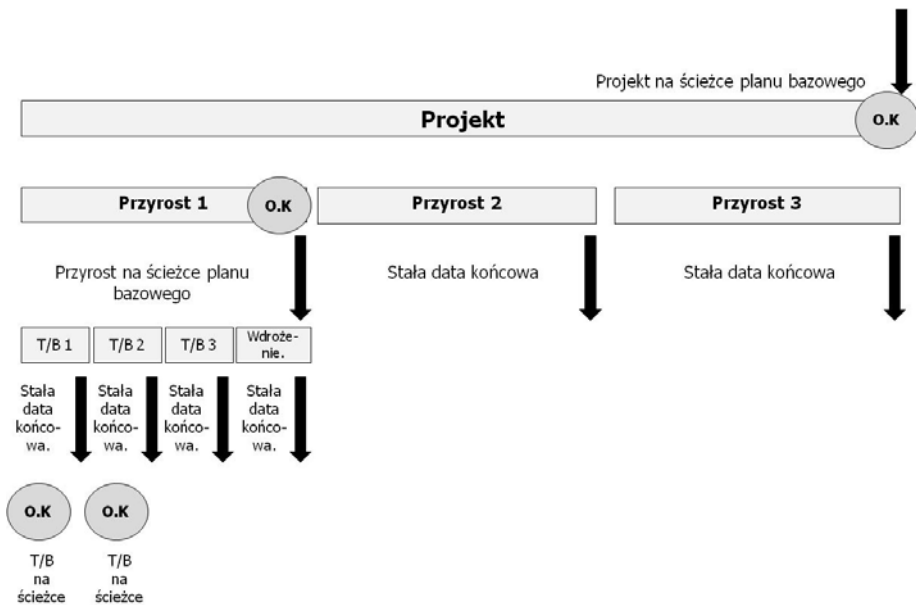
Consolidation to zapewnienie, że wytworzone produkty spełniają uzgodnione kryteria akceptacji (10-20% czasu ramy czasowej). Wszystkie produkty, które nie spełniają uzgodnionych kryteriów akceptacji, do końca danej ramy czasowej nie będą dostarczone.

Sesja zamykająca to formalna akceptacja przez osoby pełniące w zespole projektowym role Business Visionary oraz Technical Coordinator wszystkich produktów wytworzonych w ramach zakończonej ramy czasowej (1-3 godziny).

3.8 Planowanie i tworzenie harmonogramu ram czasowych

Podstawową funkcją planu dostarczania jest przygotowanie harmonogramu planowanych przyrostów (ang. *Increments*) i wewnątrz nich planowanych ram czasowych (ang. *Timebox*), które składają się na harmonogram projektu. Harmonogram powinien odzwierciedlać prawdopodobną liczbę i czasy trwania ram czasowych w danym przyroście. Zastosowanie techniki ram czasowych razem z techniką przyznawania pierwszeństwa (priorytetyzacji) MoSCoW zapewnia brak opóźnień w realizacji ram czasowych i dostarczenie produktu spełniającego cele biznesowe w tych ramach czasowych. Jeśli każda rama czasowa wchodząca w skład danego przyrostu znajduje się na ścieżce planu bazowego, wtedy ten przyrost wchodzący w skład projektu znajduje się również na ścieżce planu bazowego a więc projekt, jako całość, znajduje się na ścieżce planu bazowego projektu.

Opisywaną sytuację przedstawiono na rysunku 10 na następnej stronie.



Rysunek 10. Planowanie i tworzenie harmonogramu ram czasowych w metodycie ATERN [3]

Podsumowanie

W podejściu tradycyjnym do zarządzania projektami zakres pozostaje stały, podczas gdy czas i koszty podlegają zmianom. Natomiast w podejściu zwinnym za stałe uznawane są: czas, koszty i jakość, a zakres podlega zmianom. Jakość jest tu niezmienna, ponieważ kryteria akceptacji produktu projektu zostały uzgodnione i zatwierdzone przez klienta biznesowego we wczesnych fazach cyklu życia projektu, tak samo, jak czas i budżet.

Idea Manifestu Agile, jak również zdefiniowane w nim zasady, zostały zastosowane w podejściu zwinnym do zarządzania projektami wytwarzania oprogramowania reprezentowanym przez metodykę ATERN. Jej podstawowymi komponentami są [4]:

- filozofia ATERN zgodna z podejściem zwinnym,
- zasady ATERN zgodne z ogólnymi zasadami podejścia zwinnego zdefiniowanymi przez autorów Manifestu Agile,
- ludzie w ATERN, a więc jasno określone role w zespole projektowym wraz z przypisanym do każdej z nich zakresem obowiązków,

- procesy w cyklu życia projektu w ATERN, gdzie: odpowiednikiem faz Pre-Project i Feasibility jest faza tworzenia wizji, odpowiednikiem faz Foundations (plan na wysokim poziomie agregacji) oraz Exploration & Engineering (plany szczegółowe dla ram czasowych) jest faza planowania adaptacyjnego, odpowiednikami faz Exploration & Engineering oraz Deployment są fazy eksploracji i adaptacji (dostosowywanie się do częstych zmian wymagań), podczas gdy odpowiednikiem faz Deployment oraz Post-Project jest faza zamknięcia.
- produkty zdefiniowane w każdej fazie cyklu życia projektu realizowanego w metodyce ATERN zostały opisane w rozdziale 3.

Do technik (praktyk) rekomendowanych przez metodykę ATERN zalicza się: warsztaty ułatwiające (ang. *Facilitated Workshops*), priorytetyzacja MoSCoW, iteracyjny rozwój (ang. *Iterative Development*), modelowanie (ang. *Modelling*) oraz ramy czasowe (ang. *Timeboxing*). Metodyka ATERN dostarcza dobrych praktyk umożliwiających zarządzanie zakresem (wymaganiami) projektu, szacowanie, pomiary postępu realizacji prac w projekcie oraz ich wydajności, zapewniania jakości w cyklu życia projektu, planowania na wszystkich poziomach szczegółowości oraz zarządzania ryzykiem i kontroli realizacji projektu.

Komponenty metodyki ATERN pozwalają realizować projekty zgodnie z podejściem zwinnym zdefiniowanym w Manifeście Agile. Zastosowanie zmiennego zakresu projektu przy stałym czasie, koszcie i jakości, pozwala zredukować ryzyko niepowodzenia współczesnych projektów wytwarzania oprogramowania, a tym samym uniknąć wielu zagrożeń ery e-gospodarki.

Literatura

- [1] Highsmith J., *APM: Agile Project Management. Jak tworzyć innowacyjne produkty*, Mikom, Warszawa 2005.
- [2] Łabuda W., *Jak zrealizować udany projekt informatyczny*, materiały konferencyjne podsumowujące projekt „Program rozwoju oferty dydaktycznej i podnoszenia kompetencji wykładowców w Warszawskiej Wyższej Szkole Informatyki”, 2011.
- [3] *Agile Project Management Handbook*, version 1.1, Dynamic Systems Development Method Limited, 2013.
- [4] Łabuda W., *Podejście „zwinne” vs tradycyjne do realizacji projektów informatycznych*, seminarium zakładowe 2014/2015, Warszawa 2015.

The agile vs. traditional approach to the software development projects

Abstract

The article is an attempt to compare the traditional and agile approach to software development projects. In the introduction they discussed the basic concepts concerning the context in which the software development projects are being carried out. They presented the differences between the traditional and agile approach to software development projects and they briefly characterized the new software development project management methodologies. Next they focused on the adaptive approach for IT project management based on the Agile Manifesto. They characterized principles and practices as well as software development project life cycle for the adaptive (agile) approach. The article was finished with discussing the essence of the agile approach to software development projects management on the example of the ATERN methodology.

Key words – agile approach to software development projects, ATERN methodology