



© eVolpe Consulting Group

Metodyki wdrażania systemów IT

podejście tradycyjne i zwinne



Spis treści

Wstęp	3
Metodyki wdrożeniowe	5
System Overview	7
Waterfall	8
Budżet	9
Analiza przedwdrożeniowa	10
Prace implementacyjne	11
Testy i zarządzanie zmianą	12
Dokumentacja	13
Szkolenia i start produkcyjny	14
Agile	15
Manifest Zwinnego Programowania	16
Scrum	17
Time & Material	18
Product Discovery	19
Twój udział we wdrożeniu	20
Scrum – słowniczek pojęć	21
Sprint	23
Praca w Scrumie	24
Filary pracy w Scrumie	26
Definition of Done	27
Wnioski	28
Agile czy waterfall?	29

Wstęp

Tworzenie systemów informatycznych brzmi jak sztuka tajemna, raczej niedostępna dla tej części społeczeństwa, której nie było dane spędzić kilku lat na wydziale informatyki. Niemniej jednak mało kto z nas, przeciętnych homo internetus, jest w stanie wyobrazić sobie życie bez laptopa, smartfona, czy tabletu. Nie da się ukryć, iż z każdym rokiem przyspiesza również **postęp w zakresie rozwiązań softwarewych** obsługiwanych przez elektronikę użytkową. Technologia informacyjna rozwija się błyskawicznie i zmienia nasze życie już nie z pokolenia na pokolenie, czy z dekady na dekadę, a nawet jeszcze szybciej!

Biorąc pod uwagę raczej bezsprzeczny wniosek o opanowującym naszą codzienność trendzie informatyzacji i automatyzacji powtarzalnych procesów, warto zastanowić się, w jaki sposób skorzystamy z dobrodziejstwa oprogramowania informatycznego na gruncie biznesowym. Aby system przyniósł firmie wartość, niezbędne jest jego **efektywne wdrożenie**. W niniejszej publikacji przedstawiamy **porównanie dwóch najbardziej popularnych metodyk wdrożeniowych** dla rozwiązań systemowych w przedsiębiorstwach.

Metodyki wdrożeniowe

Metodyka wdrożeniowa to **sposób uporządkowania projektu informatycznego**, który umożliwia optymalne przeprowadzenie całego przedsięwzięcia. Ze względu na złożoność i długotrwałość procesu implementacji rozwiązań IT konieczne jest: prawidłowe zaplanowanie funkcji oprogramowania oraz właściwa organizacja pracy zespołu deweloperskiego. Kompleksowy projekt wdrożeniowy można podzielić na wiele faz. Na ogół dość wyraźnie odróżniony jest etap początkowy. Fazy związane z realizacją i jednoczesnym monitorowaniem projektu przeplatają się ze sobą. Zbiór wskazanych etapów nazywany jest cyklem życia projektu. W przypadku oprogramowania dla przedsiębiorstw niezwykle istotne jest, aby wszelkie aspekty techniczne współgrały z potrzebami biznesowymi zleceńodawcy. W tym celu na przestrzeni lat wyłoniły się dwa przewodnie podejścia metodologiczne: **tradycyjne** (inaczej: waterfall, metodyka kaskadowa) oraz **zwinne** (w tym np. framework o nazwie Scrum).

System Overview

Zanim zdecydujesz o wyborze metodyki, warto przeprowadzić warsztaty o nazwie **System Overview**. Jest to nieobowiązkowy etap działań analitycznych, oferowany przez eVolpe, który może być stosowany zarówno w przypadku dalszej pracy waterfallem jak i Scrumem.

Polega on na praktycznym i teoretycznym zapoznaniu kluczowych użytkowników ze standardową wersją systemu, prezentacji standardowej nomenklatury oraz typowych procesów obsługiwanych przez system, doradztwie w zakresie rozpoznania i sprecyzowania potrzeb po stronie klienta, empirycznym poznaniu systemu poprzez realizowanie ćwiczeń praktycznych.

Metodyki tradycyjne waterfall

Sednem metodyki waterfall jest szczegółowy plan zakładający sekwencyjną realizację poszczególnych etapów. W modelu kaskadowym fazy projektowe rozumiane są jako następujące po sobie, odrębne czynności.

Najbardziej popularny schemat postępowania zgodnie z metodyką tradycyjną prezentuje się następująco:



Budżet

Z względu na podstawową cechę metodyki kaskadowej, tj. **określony budżet**, kluczowym etapem projektu jest sporządzenie szczegółowej specyfikacji wymagań (ang. **Software Requirement Specification**), w tym listy koniecznych do zrealizowania prac mieszczących się w uzgodnionej obustronnie cenie (**fixed price**). Do porozumienia w kontekście zakresu wdrożenia, budżetu oraz mocy przerobowych zespołu deweloperskiego dochodzi podczas etapu analizy przedwdrożeńiowej. Warto przy tym pamiętać, iż coraz szybsze tempo rozwoju technologicznego skutkuje coraz wyższym współczynnikiem dezaktualizacji wymagań zawartych w takiej analizie. Oznacza to, iż połowa z przyjętych założeń biznesowych najprawdopodobniej przestanie obowiązywać po upływie około pół roku.

I. Ł. Węgrzyn, Jak pisać umowy IT dla projektów Agile – część 1, <https://www.linkedin.com/pulse/jak-pisac-umowy-dla-projektow-realizowanych-w-modelu-agile-wegrzyn>, Za: S. Atkinson, G. Benefield, Software Development: Why the Traditional Contract Model Is Not Fit for Purpose, HICSS, 2013, 2013 46th Hawaii International Conference on System Sciences.

Analiza przedwdrożeńiowa

W tradycyjnym ujęciu **analiza przedwdrożeńiowa** służy rozpoznaniu najważniejszych aspektów funkcjonowania przedsiębiorstwa. W ramach takiego spotkania omawiamy nadrzędne cele przyświecające implementacji systemu, główne potrzeby biznesowe poszczególnych grup beneficjentów, a także procesy zachodzące w przedsiębiorstwie. Dobrze zorganizowany etap początkowy pozwala na ujawnienie kluczowych potrzeb oraz sporządzenie pełnej specyfikacji i wyceny gotowego oprogramowania.

Czy wiesz, że...?

Zjawisko połowicznego rozkładu aktualności rozwiązań IT (podobne do erozji izotopów) zaobserwowali badacze na Uniwersytecie Missouri w Stanach Zjednoczonych. O ile w latach 80. ubiegłego stulecia *half-life* analizy wynosił 10-12 lat, w roku 2000 2-3 lata, to obecnie nie przekracza 6 miesięcy¹.

Prace implementacyjne

Faza prac implementacyjnych polega na stworzeniu kodu programistycznego, który zapewnia funkcjonowanie systemu zgodnie z założeniami przyjętymi podczas analizy. Początkowo atrakcyjna wizja posiadania odgórnie ustalonego budżetu, a także szczegółowej specyfikacji koniecznych do wykonania prac, może się jednak okazać poważnym ograniczeniem. Jak wynika z wieloletniej praktyki realizowania projektów informatycznych w eVolpe, koncepcja potrzeb biznesowych oraz bezpośrednio z nimi związanych funkcji oprogramowania potrafi zmieniać się bardzo dynamicznie, zwłaszcza kiedy abstrakcyjny wcześniej projekt zaczyna nabierać konkretnych kształtów.

Metodyka kaskadowa zakłada **testy systemu** przed jego ostatecznym uruchomieniem. Sprawdzana jest **zgodność osiągniętych rezultatów ze sporządzoną na początku specyfikacją**, a nie to, czy oprogramowanie odpowiada na **aktualne** potrzeby biznesowe. Podczas przeglądu akceptacyjnego weryfikujemy, czy założenia przyjęte w trakcie analizy zostały zrealizowane we właściwy sposób i czy funkcje systemu będące przedmiotem umowy działają poprawnie. Tematem pobocznym w przypadku waterfall są kwestie realizacji nieuwzględnionych w początkowym porozumieniu usprawnień. Istnieje co prawda możliwość przyjęcia aneksu do umowy, aczkolwiek w takim przypadku wszystkie dodatkowe prace wiązać się także z dodatkowym kosztem. **Paradoksalnie wzrasta wówczas ryzyko przekroczenia przyjętego na początku budżetu.** Wprawdzie istnieje możliwość podziału kompleksowego projektu na mniejsze etapy, jednak i w tym przypadku istnieje możliwość, iż wykonane w danym etapie prace nie będą w pełni odzwierciedlać aktualnych potrzeb. Sama procedura zarządzania zmianą również wprowadza dodatkowe usztywnienia we współpracy, a także dezorganizację i w konsekwencji spadek efektywności pracy zespołu deweloperskiego.

Dokumentacja

Sztywna struktura procesu jest jednocześnie zaletą, jak i najpoważniejszą wadą tradycyjnego podejścia. Wybór tego typu metodyki następuje najczęściej w przypadku przedsiębiorstw i organizacji, w których inwestor nie ma czasu pozostać w stałym kontakcie z firmą wdrożeniową. Niezwykle istotne w tym przypadku jest jednak to, by klient jasno określił swoje oczekiwania - jeszcze zanim prace implementacyjne zostaną uruchomione!

Jedną z charakterystycznych cech metodyki tradycyjnej jest **bardzo szczegółowa dokumentacja projektu**. Na pisemnym potwierdzeniu obustronnych ustaleń zależy zarówno firmie wdrożeniowej, jak i zleceniodawcy. Ma to bezpośredni związek z obowiązkiem utrzymania ustalonej ceny, terminu realizacji oraz dostarczeniem produktu zgodnego ze specyfikacją.

Szkolenia i start produkcyjny

Ważnym etapem podczas tradycyjnego wdrożenia jest **przeszkolenie użytkowników i administratorów**. W ten sposób zapewniamy najwyższy stopień znajomości funkcji dostarczonego oprogramowania. Faza szkoleń nie powinna być pomijana! Aby system przynosił założone korzyści biznesowe, konieczna jest pełna świadomość oferowanych udogodnień; zwłaszcza wśród pracowników firmy, którzy nie byli zaangażowani we wcześniejsze etapy wdrożenia.

Po zaakceptowaniu przygotowanego przez firmę wdrożeniową systemu następuje tzw. **start produkcyjny systemu**. Od tego momentu do zadań dostawcy należy **utrzymanie poprawnego działania instancji** w zatwierdzonym przez inwestora stanie oraz **dalszy jego rozwój** na podstawie nowego zlecenia.

Odpowiedź na pytanie „Czy wybrać metodykę waterfall?” powinna brzmieć „Tak” tylko i wyłącznie wtedy, gdy warunkiem koniecznym po stronie zleceniodawcy jest **zamknięty budżet** (przy jednoczesnej rezygnacji z dokonywania zmian w pierwotnych wymaganiach) lub **brak możliwości pełnego zaangażowania w proces wdrożenia**. Przyjęcie kaskadowej techniki realizowania projektu może zostać wymuszone, gdy przedmiotem umowy jest oprogramowanie, którego działanie regulują sztywne normy prawne (np. systemy księgowe). Tradycyjnym sposobem implementowane są zwykle systemy zamawiane przez instytucje państwowe. Ze względu na administracyjny wymóg prowadzenia szczegółowej dokumentacji oraz usztywniony budżet to właśnie waterfall najlepiej odpowiada ich specyficznym potrzebom.

Metodyka zwinna agile

Metodyki zwinne, w tym najbardziej popularny framework o nazwie Scrum, wywodzą się z opracowanego na początku XXI wieku ruchu agile (z ang. zwinny). W roku 2001 grupa kilkunastu praktyków wdrażania systemów informatycznych ze Stanów Zjednoczonych ogłosiła tzw. **Manifest Programowania Zwinnego**.

W tłumaczeniu na język polski brzmi on następująco:

Nowe metody programowania odkrywamy dzięki praktyce w programowaniu i wspieraniu w nim innych.
W wyniku naszej pracy zaczęliśmy bardziej cenić:
Ludzi i interakcje od procesów i narzędzi
Działające oprogramowanie od szczegółowej dokumentacji
Współpracę z klientem od negocjacji umów
Reagowanie na zmiany od realizacji zainicjowanego planu.
Oznacza to, że elementy wypisane po prawej są wartościowe, ale większą wartość mają dla nas te, które wypisano po lewej.

Należy pamiętać, iż przytoczony manifest zakłada **równoważność wszystkich przyjętych wartości**. Nie ma mowy o „zwinnosci” działania, gdy w pracy zespołu zabraknie któregoś z wyżej wymienionych elementów. Dostarczając działające, wytworzone we współpracy z klientem oprogramowanie, nie zachowamy się „zwinnie”, dopóki nie zapewnimy przyjaznego środowiska interakcji międzyludzkich oraz możliwości swobodnego reagowania na pojawiające się zmiany ustaleń. Podobnie zleceniodawca nie może wymagać nowoczesnego wdrożenia, jeżeli nie zakłada swojego aktywnego udziału w projekcie.

Scrum

Jednym z ważniejszych i szerzej rozpowszechnionych frameworków, zgodnych z duchem agile, jest **Scrum**. Jak precyzyjnie definicja, jest to **struktura procesu, a nie metodyka sama w sobie**. Scrum to zestaw podstawowych zasad, ról i spotkań tworzących ramy, w obrębie których każdy zespół samodzielnie decyduje, jak chce pracować.

Time & Material

Kolejne działania podejmowane w ramach wdrożenia realizowanego w Scrumie planowane są etapami i na podstawie bieżących doświadczeń użytkowników. Standardowo w ramach każdego z kilkutygodniowych iteracji (tzw. sprintów) dostarczany jest działający fragment (przyrost) systemu. Może być on na bieżąco testowany przez użytkowników końcowych i to nie tylko pod kątem poprawności działania, ale również praktycznego zastosowania podczas realizacji realnych procesów biznesowych. Dzięki temu możliwa jest cykliczna korekta rozwiązań oraz wprowadzanie zmian zgodnie z pojawiającymi się potrzebami. Zleceniodawca ma przy tym **pełną swobodę sterowania budżetem i zakresem planowanych do wykonania prac** poprzez nadawanie im odpowiednich priorytetów. Rozliczenie projektu odbywa się na zasadach **Time & Material**, co oznacza, że nakłady finansowe ponoszone przez inwestora dotyczą wyłącznie czasu i zasobów poświęconych przy pracy nad wdrożeniem.

W odróżnieniu od podejścia tradycyjnego, techniki zwinne nie wymagają pieczołowicie przygotowywanej dokumentacji, która na kilkuset stronach opisuje szczegółowe zasady działania abstrakcyjnego systemu. Podczas analizy przedwdrożeniowej określone są główne cele biznesowe oraz kluczowe funkcje systemu odpowiadające biznesowym potrzebom inwestora. Nie ma jednak potrzeby definiowania finalnego efektu prac już na samym początku wdrożenia – koncepcja poszczególnych funkcji realizowana jest z zastosowaniem zasady małych kroków, dzięki czemu z przyrostu na przyrost użytkownicy mogą poeksperymentować z różnymi opcjami i na bazie swoich doświadczeń wybrać idealną dla siebie.

Product Discovery

Ciekawym elementem analizy odróżniającym Scrum od podejścia tradycyjnego są kreatywne warsztaty **Product Discovery**. Ich głównym celem jest zdefiniowanie **co, dla kogo i jak** ma zostać wyprodukowane. W tego typu spotkaniach uczestniczą kluczowi, przyszli użytkownicy systemu, technicy oraz koordynatorzy wdrożenia. Czas trwania warsztatów zależy od złożoności wymagań i wynosi do kilku dni. Efektem **Product Discovery** powinna być ogólna wizja systemu zamknięta w pierwszej wersji Backlogu produktu, czyli listy prac do wykonania. Ważne, aby podczas warsztatów rozpoznano funkcjonalność systemu z perspektywy każdego z przyszłych jego beneficjentów.

Twój udział we wdrożeniu

Cechą charakterystyczną Scrum - którą niektórzy dość po-
chopnie uznają za wadę - jest **konieczność aktywnego udziału inwestorów w procesie wdrożenia na każdym z jego etapów**. W praktyce jednak zapobiega to nieporozumi-
eniom, buduje relacje pomiędzy zespołem deweloperskim i klientem, a także wspólną wizję tworzonego produktu. W efekcie zleceniodawca otrzymuje system maksymalnie do-
pasowany do indywidualnych potrzeb. Ponieważ po każdym ze sprintów istnieje możliwość przetestowania kole-
jnych usprawnień, a także decydowania które zagadnienia są aktualnie kluczowe i powinny zostać wykonane w pier-
wszej kolejności, zleceniodawca ma pewność, że przeznaczane na wdrożenie środki są optymalnie inwe-
stowane, a wartość biznesowa systemu w pełni maksymal-
izowana.

Scrum

słowniczek pojęć



Daily Stand-Up - spotkania służące weryfikacji statusu prac.

Product Backlog - uporządkowana według priorytetu lista wymagań wobec funkcji systemu.

Product Owner - osoba odpowiedzialna za określenie wizji produktu i nadzór nad jego rozwojem.

Refinement - spotkanie, podczas którego omawiane są nowe wymagania odnośnie systemu.

Retrospektywa - spotkania służące do podsumowania sprintu, tj. weryfikacji oraz identyfikacji i uporządkowaniu istotnych elementów, które sprawdziły się w dotychczasowym działaniu oraz takich, które kwalifikują się do usprawnienia w przyszłości.

Scrum Master - osoba odpowiedzialna za opiekę nad Zespołem Deweloperskim i kontrolę zgodności procesu.

Sprint - zwykle 2-tygodniowy okres, w którym realizowane są prace w celu osiągnięcia kolejnego działającego fragmentu systemu stanowiącego przyrost systemu.

Sprint Backlog - lista prac do wykonania w ramach jednego sprintu.

Sprint Planning - spotkanie, podczas którego ustalane są cele do osiągnięcia w konkretnym sprincie oraz zakres niezbędnych do tego prac i sposób ich realizacji.

Sprint Review - prezentacja zleceniodawcy gotowego rezultatu pracy wykonanej w danym sprincie.

Zespół Deweloperski - grupa osób odpowiedzialnych za dostarczenie produktu.

Sprint

Kompleksowy, ale podzielony na sprinty projekt umożliwia zespołowi deweloperskiemu skupienie uwagi na przygotowaniu i dokładnym, a także cyklicznym sprawdzeniu opracowywanych fragmentów kodu. Metodyki zwinne gwarantują w ten sposób niezawodność i najwyższą jakość finalnych produktów. Dzięki możliwości dostarczania poszczególnych części działającego oprogramowania z większą częstotliwością, Scrum zapewnia najwyższą zgodność produktu z oczekiwaniami i aktualnymi potrzebami.

Praca w Scrumie

Każdy sprint rozpoczyna się od tzw. **planningu**. To wtedy ustalany jest cel na najbliższe tygodnie oraz zakres niezbędnych do tego prac i sposób ich realizacji. W trakcie sprintu odbywają się jeszcze krótkie spotkania kontrolne, czyli Daily Stand-up. Przez około 15 minut przeprowadzana jest inspekcja wykonanych i zaplanowanych czynności. Możliwa jest również ewentualna adaptacja planu do zaistniałych nowych warunków. Nowym wymaganiom zgłaszanym przez zleceniodawcę dedykowane są spotkania o nazwie Refinement. W przypadku konieczności zmodyfikowania projektu, każdą zmianę należy najpierw odpowiednio doprecyzować, jasno określić beneficjenta oraz wartość biznesową, następnie podzielić na części i oszacować. Tak przygotowane zagadnienia mogą zostać dołączone do Backlogu produktu i przyjęte do realizacji w kolejnych sprintach. Po zakończonej iteracji następuje prezentacja zleceniodawcy prac wykonanych przez zespół deweloperski (tzw. Sprint Review). Dzięki osiągnięciom współczesnej technologii takie spotkania mogą być organizowane zdalnie. Jest to również odpowiedni moment na pierwszą dyskusję w kwestii nowych wymagań zainspirowanych obecnym kształtem systemu. Taka adaptacja nie jest możliwa przy projektach realizowanych zgodnie z metodyką waterfall.

Po zakończonym sprincie możliwe jest rozpoczęcie produkcyjnego użytkowania wykonanego przyrostu. Ostatnim spotkaniem w ramach danej iteracji jest tzw. Retrospektywa, w której uczestniczą Product Owner, Zespół Deweloperski oraz Scrum Master. Ma ona na celu podsumowanie sprintu w zakresie zarówno pozytywnych, jak i negatywnych wydarzeń, które miały w nim miejsce. Dzięki takiej analizie Zespół jest w stanie zminimalizować elementy, które negatywnie wpływają na jego efektywność oraz wprowadzić ulepszenia, dzięki którym praca w kolejnych sprintach przyniesie jeszcze większe korzyści dla projektu (inspekcja i adaptacja).

Liczba sprintów może, ale nie musi być z góry ustalona. To zleceniodawca decyduje, kiedy wspólnie opracowywany system jest już gotowy i spełnia jego oczekiwania. W takim momencie następuje przejście do fazy utrzymania projektu, którą regulują zasady zawarte w umowie serwisowej. Gdyby w późniejszym czasie zaistniała konieczność wprowadzenia dodatkowych ulepszeń produktu, można porozumieć się co do powrotu do działania w sprintach.

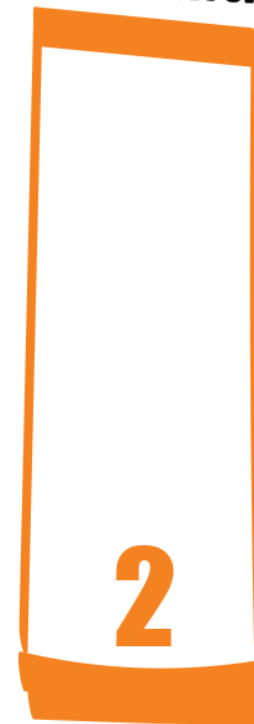
Powyżej opisane praktyczne zasady podziału obowiązków opierają się na trzech filarach pracy w Scrumie.

Filary pracy w Scrumie

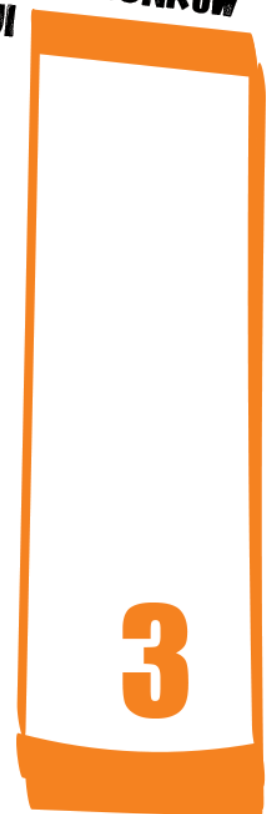
**ZACHOWAJ
PEŁNĄ TRANSPARENTNOŚĆ
DZIAŁANIA**



**WPROWADŹ
REGULY INSPEKCJI
REZULTATÓW**



**ZAPEWNIJ
MOŻLIWOŚĆ
ADAPTACJI
DO ZMIENNYCH
WARUNKÓW**



Definition of Done

Poszczególne elementy procesu muszą być jawne dla wszystkich zaangażowanych osób. Należy zadbać o pełne zrozumienie podejmowanych działań, np. poprzez ujednolicenie nazewnictwa, posługiwanie się określonym kanałem przekazywania informacji, ustalenie definicji ukończenia etapu (ang. **Definition of Done**) itp. Inspekcja i adaptacja służą wykrywaniu mankamentów oraz dokonywaniu ewentualnych korekt. Przy zachowaniu wszystkich trzech zasad zapewniony jest optymalny przebieg nawet najbardziej skomplikowanego wdrożenia.

Wspomniane filary wynikają z przyjętych w Scrumie wartości: **koncentracji, zaangażowania, otwartości, szacunku i odwagi**. Tylko stosując się do wszystkich opisanych powyżej reguł, możliwe jest stworzenie atmosfery wspólnego zaufania oraz osiągnięcie sukcesu w postaci efektywnego systemu.

Wnioski

Kwestia metodyki wdrożenia systemu informatycznego w firmie powinna zostać bardzo dokładnie rozpatrzona. Niezależnie od ostatecznej decyzji nie warto rozpoczynać procesu bez analizy wymagań. Oba opisane podejścia zakładają dość precyzyjne zbadanie otoczenia i warunków wewnętrznych determinujących architekturę zamawianego systemu. To, czy prace implementacyjne będą prowadzone z wykorzystaniem metodyki tradycyjnej, czy zwinnej, w większości przypadków zależy od podejścia zleceniodawcy do kwestii jego aktywnego udziału we wdrożeniu. Dysponując swoim czasem, musimy zdawać sobie sprawę, że Scrum wymaga stałej komunikacji i współpracy oraz bieżącej weryfikacji wykonanych prac. Dla niektórych firm tak duże zaangażowanie pracowników w proces wdrożenia stanowi pewną niedogodność. Inni uważają to jednak za główną zaletę podejścia agile. Z doświadczenia eVolpe wynika jednak, że zaangażowanie po stronie zleceniodawcy ma kluczowe znaczenie dla efektywnego wdrożenia; w konsekwencji zwrot czasu poświęconego przez zleceniodawcę jest niebywały. Pełna kontrola nad przebiegiem procesu zapewnia realizację przede wszystkim kluczowych funkcji. Działając w Scrumie przy pełnym zaangażowaniu i stałej kontroli budżetu raczej trudno "roztrwonić" go na funkcje, które okażą się nieużyteczne.

Agile czy waterfall?

Jeżeli chodzi o budżet, to wbrew pozorom nie on powinien być głównym determinantem wyboru metodyki. Podejście kaskadowe wymaga zamknięcia wdrożenia w sztywnej wycenie, dzięki czemu wydaje nam się, że przystępując do projektu mamy pewność ile pieniędzy wydamy. Jest to oczywiście złudne, ponieważ z uwagi na szybką dezaktualizację wymagań oraz możliwość praktycznego poznania systemu dopiero po zakończeniu wdrożenia, bardzo często niezbędne są do wykonania dodatkowe prace, a część wykonanych funkcji nie ma już zastosowania po zakończeniu prac. Wybierając Scruma możemy również określić maksymalną kwotę projektu i na bieżąco ustalać na co będziemy ją przeznaczać w kolejnych sprintach. Zwinne wdrożenie może zostać zakończone przez zlecniodawcę na dowolnym etapie, dlatego nie ma ryzyka przekroczenia zakładanego budżetu. Dodatkowo co kilka tygodni istnieje możliwość zgłoszenia zmian w projekcie i dopasowania systemu do nowych warunków, których nie sposób przewidzieć na początku drogi. Dzięki bieżącej priorytetyzacji zagadnień do wykonania praktycznie nie ma możliwości, żeby kluczowe funkcje nie zostały zrealizowane. W przypadku waterfall zgłoszenie jakiegokolwiek zmiany przysparza wielu problemów, wymaga podpisywania aneksów do umowy lub poruszania się w ramach formalnej procedury zarządzania zmianą, akceptacji nowych ustaleń co do zakresu prac i przekształcenia harmonogramu. Ma to swoje odzwierciedlenie także w naruszeniu zakładanego budżetu. W Scrumie zmiana jest czymś naturalnym i może być dokonywana na dowolnym etapie. Jeżeli tylko decydujesz się aktywnie brać udział w procesie i pozostawać podczas całego procesu wdrożenia w partnerskim kontakcie z firmą wdrożeniową, zdecydowanie lepszym rozwiązaniem jest wybór metodyki zwinnej.



eVolpe Consulting Group

Winogrody Business Center

Aleje Solidarności 46

61-696 Poznań

Tel.: +48 518 105 108

www.evolpe.pl

biuro@evolpe.pl