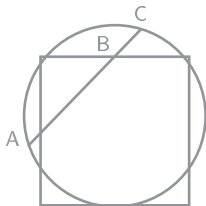


Inżynieria oprogramowania

Radosław Klimek

2015-23



<http://home.agh.edu.pl/rklimek>

1 Podejście zwinne oraz metodyka Scrum

1 Podejście zwinne oraz metodyka Scrum

Podejście zwinne oraz metodyka Scrum

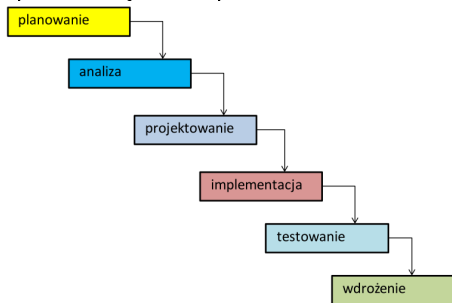


Edgar DEGAS: *Lecja tańca w operze*

Metody tradycyjne – podejście kaskadowe

Kaskadowe podejście do zarządzania projektami IT (ang. *Waterfall*), wymusza bardzo długi czas oczekiwania od złożenia zamówienia przez klienta do wdrożenia oprogramowania.

Sterowanie projektem przez wykonywanie czynności w określonych i zaplanowanych etapach.



Model kaskadowy znajduje zastosowanie w projektach, gdzie kluczem do sukcesu jest ustrukturyzowanie prac zespołu, wysoka przewidywalność oraz odpowiedni nacisk na kontrolę.

Metody tradycyjne – wady

- Problemy w komunikacji zespołu,
- niejasne wymagania klienta,
- przekraczająca terminy implementacja,
- testowanie zazwyczaj jest pomijane,
- defekty wykrywane są na końcu, a ich usuwanie jest drogie,
- niska elastyczność,
- ulegające zmianie warunki biznesowe.

Programowanie zwinne

Definicja

Programowanie zwinne – grupa metod wytwarzania oprogramowania opartego na programowaniu iteracyjno-przyrostowym, powstałe jako alternatywa do tradycyjnych metod typu kaskadowego.

Programowanie zwinne zostało zaproponowane w tzw. **Manifest Agile** (ang. **Manifesto for Agile Software Development**) w 2001 roku. I coraz bardziej zyskuje na popularności.

Manifest Agile

Treść Manifestu

”Wytwarzając oprogramowanie i pomagając innym w tym zakresie, odkrywamy lepsze sposoby wykonywania tej pracy. W wyniku tych doświadczeń przedkładamy:

- **Ludzi i interakcje** ponad procesy i narzędzia.
- **Działające oprogramowanie** ponad obszerną dokumentację.
- **Współpracę z klientem** ponad formalne ustalenia.
- **Reagowanie na zmiany** ponad podążanie za planem.

Doceniamy to, co wymieniono po prawej stronie, jednak bardziej cenimy to, co po lewej.”

Założenia i cele manifestu Agile

- Osiągnięcie satysfakcji klienta poprzez szybkość wytwarzania oprogramowania,
- działające oprogramowanie jest dostarczane okresowo (raczej tygodniowo niż miesięcznie),
- podstawową miarą postępu jest działające oprogramowanie,
- późne zmiany w specyfikacji nie mają destrukcyjnego wpływu na proces wytwarzania oprogramowania,
- bliska, dzienna współpraca pomiędzy biznesem a deweloperem.

Założenia i cele manifestu Agile (cd.)

- bezpośredni kontakt, jako najlepsza forma komunikacji w zespole i poza nim,
- ciągła uwaga nastawiona na aspekty techniczne oraz dobry projekt,
- prostota,
- samozarządzalność zespołów,
- regularna adaptacja do zmieniających się wymagań.

Metodyki zwinne – charakterystyka

- Metodyki zwinne biorą pod uwagę realia w jakich pracujemy – szybkie tempo, dużą konkurencyjność, ciągle zmieniające się otoczenie i nieprzewidywalność zarówno wewnętrzną, jak i przede wszystkim zewnętrzną.
- Główną podstawą metodyk zwinnych jest zaobserwowanie faktu, że oczekiwania odbiorcy (klienta) oraz warunki rynkowe często ewoluują podczas trwania projektu.
- Metodyka ta polega na organizacji programistów w nieduże, samodzielne zespoły programistyczne, w których członkowie sami szukają rozwiązań problemów, które pojawiają się w trakcie pracy.

Metodyki zwinne – charakterystyka (cd.)

- Nie występuje tu problem komunikacji. Metoda nastawiona jest na bezpośrednią komunikację pomiędzy członkami zespołu. Dzięki temu zbędne jest tworzenie rozbudowanej dokumentacji kodu.
- Skład zespołów jest zazwyczaj wielofunkcyjny oraz samzarządzalny, bez zastosowania jakiegokolwiek hierarchii korporacyjnej.
- Członkowie zespołu biorą odpowiedzialność za zadania postawione w każdej iteracji. Sami decydują jak osiągnąć postawione cele.

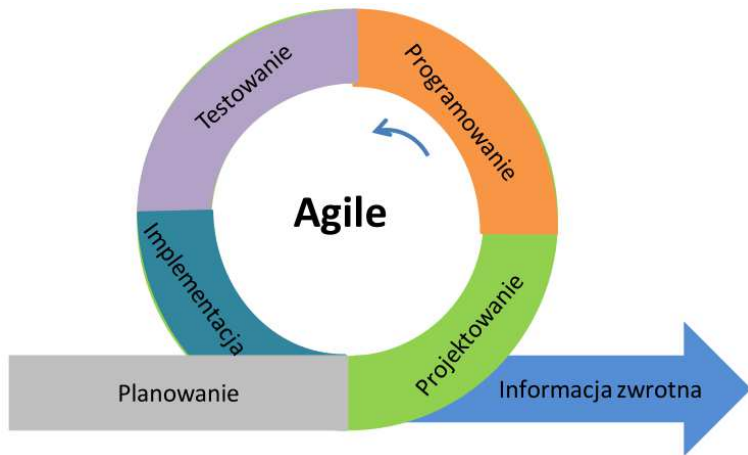
Etapy programowania zwinnego

Wyróżniamy następujące etapy składające się na pełny cykl:

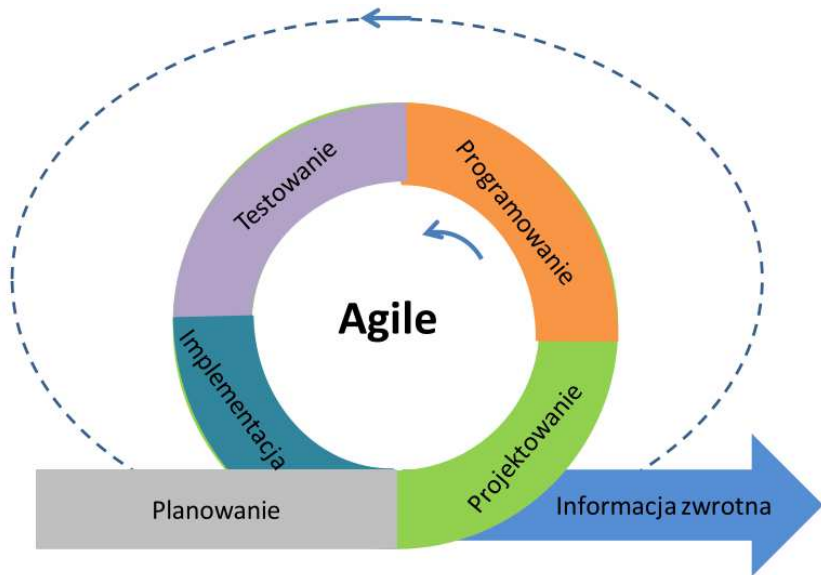
- Planowanie (ang. *plan*),
- projektowanie (ang. *design*),
- programowanie (ang. *development*),
- testowanie (ang. *testing*),
- implementacja (ang. *release*),
- informacja zwrotna (ang. *feedback*).

Powyższe etapy tworzą cykl powtarzany, aż do czasu zakończenia danego projektu.

Etapy programowania zwinnego (cd.)



Etapy programowania zwinnego (cd.cd.)



Etapy programowania zwinnego – uwagi

- Kolejne cykle mają służyć skorygowaniu przygotowanego zadania na bazie informacji od klienta (użytkownika). Służą też elastycznemu wprowadzaniu ewentualnych zmian wymagań ze strony klienta jeżeli takie się pojawiły w formie informacji zwrotnej (ang. *feedback*).
- Stosuje się je, aby zapobiec ryzykownej sytuacji błędu ostatecznej wersji produktu, jest to mniej kosztowne niż testowanie kolejnych niepełnych wersji produktu.
- Kolejne cykle nie mają natomiast na celu poprawiania błędów danego zadania w nieskończoność. Taka sytuacja wynika często z pominięcia lub niedokładnego przeprowadzenia etapu planowania (w tym zbierania i analizy wymagań od klienta).

Co to jest Scrum?

Jednym ze sposobów realizacji metodyki pracy Agile, i zarazem najpopularniejszym, jest **metodyka Scrum**.

Co to jest Scrum?

Jednym ze sposobów realizacji metodyki pracy Agile, i zarazem najpopularniejszym, jest **metodyka Scrum**.

- Metodyka Scrum, lub krótko Scrum, jako pojęcie jest często używany zamiennie ze słowem Agile i nie do końca jest to błąd.
- Dobrze stosowany Scrum może być drogą do osiągnięcia założeń podejścia zwinnego do wytwarzania oprogramowania, czyli Agile.
- Prawidłowo stosowany Scrum, spełnia cztery podstawowe założenia manifestu Agile i tzw. 12 zasad Agile.

Młyn – scrum

W rugby scrum czyli młyn jest metodą wznowienia gry polegającą na tym, że gracze pakują się ściśle razem z opuszczonymi głowami, a następnie próbują zdobyć piłkę.



Historia Scrum

- W 1986 roku Hirotaka Takeuchi i Ikujiro Nonaka wprowadzili termin Scrum w kontekście rozwoju produktu w artykule „Nowa gra rozwoju nowych produktów” („The New New Product Development Game”) na łamach *Harvard Business Review*.
- Pokazano tu nowe podejście do rozwoju produktu komercyjnego, którego celem jest zwiększenie szybkości i elastyczności.
- Koncepcja powstała w oparciu o studia przypadków z firm (Honda, Canon, Fuit-Xerox, Brother, Epson, 3M, HP oraz Xerox) produkujących samochody, kserokopiarki i drukarki.
- Autorzy nazywali to podejściem holistycznym lub rugby, ponieważ cały proces jest wykonywany przez jeden wielofunkcyjny zespół w wielu nakładających się fazach, podczas których zespół „próbuje pokonać dystans jako jednostka, podając piłkę tam i z powrotem”.

Historia Scrum (cd.)

- 1993 – został utworzony pierwszy zespół scrumowy i opracowano dokładne procedury Scrum. Ken Schwaber wykorzystywał tą metodę w swojej firmie Advanced Development Methods, a Jeff Sutherland, John Scumniotales i Jeff McKenna opracowali podobne podejście w Easel Corporation.
- 1995 – Sutherland i Schwaber wspólnie przedstawili artykuł opisujący ramy postępowania zgodnego z metodyką scrum (*the scrum framework*) w ramach OOPSLA'95 (Object-Oriented Programming, Systems, Languages & Applications '95) w Austin w Teksasie.
- 2001 – Schwaber i Miki Beedl opisują metodykę Scrum w książce pt. „Agile Software Development with Scrum”.
- 2009 – metodyka Scrum jest oficjalnie zdefiniowana w publicznym dokumencie o nazwie „The Scrum Guide”.

Scrum – pierwsze podsumowanie

- Metodyka Scrum to jedna z najpopularniejszych metodyk zwinnych w zarządzaniu projektami IT, oparta na zasadach Agile.
- To metodyka, która daje możliwość rozwiązywania złożonych problemów, adaptacji produktu, do wymagań klienta.
- Scrum umożliwia wydajne i innowacyjne kreowanie produktu, o możliwie jak najwyższej jakości dla klienta, ze względu na iteracyjny (przyrostowy) proces kontroli.
- Metodyka Scrum przez twórców metody definiowana jest, jako ramy procesu, w którym można wdrożyć różnorodne podprocesy oraz techniki w celu zarządzania złożonym rozwojem produktu.
- Dzięki metodyce scrum rozpoczęcie pracy wymaga określenia "wizji" produktu, bez konieczności dokładnego wyznaczenia szczegółowych cech formy, w jakiej ma zostać dostarczony klientowi.

Scrum – pierwsze podsumowanie (cd.)

- Scrum opiera się na empirycznej teorii sterowania procesem.
- Empiryzm zakłada, że wiedza pochodzi z doświadczenia i podejmowania decyzji opartych na tym, co jest już znane.
- Scrum stosuje podejście iteracyjne i przyrostowe w celu optymalizacji przewidywalności i kontroli ryzyka.
- Może mieć zastosowanie w realizacji projektów w oparciu o metodyki zwinne zgodne z manifestem Agile.

Metodyka Scrum – definicje

Definicja

Scrum to iteracyjne i przyrostowe ramy postępowania zgodne ze Scrum Guide. Może mieć zastosowanie w realizacji projektów w oparciu o metodyki zwinne zgodne z manifestem Agile.

Metodyka Scrum – definicje

Definicja

Scrum to iteracyjne i przyrostowe ramy postępowania zgodne ze Scrum Guide. Może mieć zastosowanie w realizacji projektów w oparciu o metodyki zwinne zgodne z manifestem Agile.

Definicja (Scrum Guide, XI 2020)

Scrum to uproszczone ramy postępowania, które pomagają poszczególnym osobom, zespołom i organizacjom wytwarzać wartość poprzez adaptacyjne rozwiązywanie złożonych problemów.

Metodyka Scrum – definicje

Definicja

Scrum to iteracyjne i przyrostowe ramy postępowania zgodne ze Scrum Guide. Może mieć zastosowanie w realizacji projektów w oparciu o metodyki zwinne zgodne z manifestem Agile.

Definicja (Scrum Guide, XI 2020)

Scrum to uproszczone ramy postępowania, które pomagają poszczególnym osobom, zespołom i organizacjom wytwarzać wartość poprzez adaptacyjne rozwiązywanie złożonych problemów.

- Sama metodyka Scrum nie jest skomplikowana, najtrudniejsza w Scrumie jest integracja podanych informacji i wdrożenie metody w praktyce.
- Podstawą jest dokładne zrozumienie idei, poszczególnych etapów, a także roli każdego uczestnika projektu.
- Dla optymalnego działania niezbędne jest również bezwzględne przestrzeganie reguł, które znajdują się w oficjalnym **Scrum Guide**.

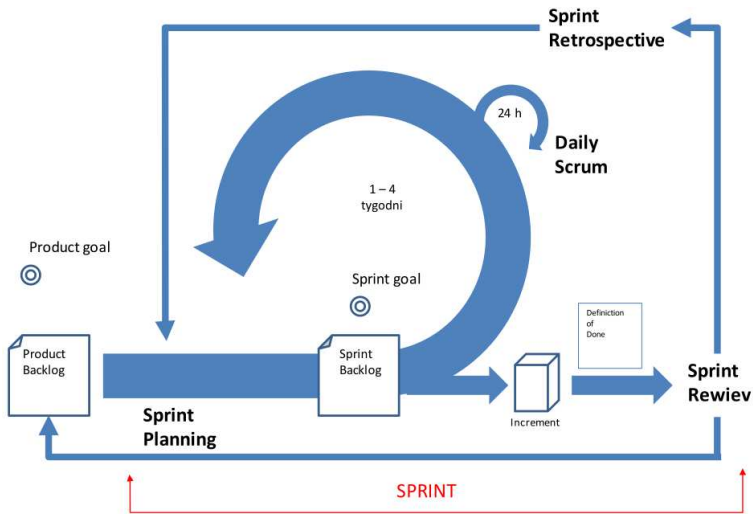
Co to jest scrum

- Scrum sam w sobie jest swego rodzaju strukturą, szkieletem lub ramą (ang. *framework*) zarządzania projektami w ramach zwinnego zarządzania.
- Nie narzuca on konkretnych praktyk ani technik tworzenia oprogramowania i zarządzania zespołami.
- Zgodnie z definicją ze Scrum Guide'a w obręb każdego scruma wchodzi: **zespoły scrumowe** oraz powiązane z nimi **role**, **artefakty**, **wydarzenia** i **reguły**.

Z czego składa się każdy scrum

- **Role:**
Zespół scrumowy = Właściciel produktu + Mistrz scruma + Zespół deweloperski
- **Artefakty:**
Rejestr produktu + Rejestr sprintu + Przyrost
- **Zobowiązania artefaktów:**
Cel produktu + Cel sprintu + Definicja ukończenia
- **Zdarzenia:**
Sprint = Planowanie sprintu + Dzienny sprint + Przegląd sprintu + Retrospektywa sprintu
- **Reguły:**
Przejrzystość + Inspekcja + Adaptacja

Struktura scruma



Zespół scrumowy



Product Owner



Scrum Master



Developers

Reguły Scruma

Zdefiniowano trzy ważne reguły, które stanowią filary każdego scruma, które mają zapewnić jego poprawne działanie w duchu Agile i powinny być stosowane przez cały zespół na każdym etapie pracy. Reguły te stanowią podstawę kontroli i zarządzania całym procesem:

- 1 **przejrzystość** (ang. *transparency*),
- 2 **adaptacja** (ang. *adaptation*),
- 3 **inspekcja** (ang. *inspection*).

Reguły scruma – omówienie

- **Przejrzystość** – istotne aspekty procesu muszą być zrozumiałe i przejrzyste dla osób odpowiedzialnych za tworzenie produktu lub powiązane z tym procesem. Konieczne jest ustandaryzowanie istotnych kwestii, dzięki czemu cały proces, jego kontrola oraz cel końcowy są „przejrzyste”.
- **Inspekcja** – kontrole postępu prac, powinny być przeprowadzane przez wysoko wykwalifikowanego inspektora tak często jak tylko to możliwe (inspekcje nie powinny utrudniać pracy zespołowi scrum) aby wykryć ewentualne ulepszenia lub też zapobiec błędom jak najwcześniejszy etapie.
- **Adaptacja** – w przypadku zarejestrowania podczas rutynowych kontroli (inspekcji), jakiegokolwiek czynnika, który zaburza proces, zmniejsza efektywność produkcji lub wartość produktu, reguła adaptacji pozwala zespołowi na natychmiastową zmianę sposobu pracy.

Artefakty każdego scruma

Definicja

Artefakty scruma reprezentują pracę lub wartość niezbędną do utrzymania transparentności oraz umożliwiają dokonywanie inspekcji i adaptacji.

- Wszystkie artefakty muszą być zaplanowane tak, aby kluczowe informacje były jasne i zrozumiałe dla wszystkich.
- Bez artefaktów proces scrumowy nie może istnieć.

Artefakty scruma:

- Rejestr produktu,
- Rejestr sprintu,
- Przyrost.

Rejestr produktu

Definicja

Rejestr produktu (ang. *Product Backlog*) – jest to lista zawierająca wszystkie działania niezbędne do wytworzenia danego produktu. Jest to coś co w tradycyjnych projektach nazywaliśmy wymaganiami użytkownika lub specyfikacją wymagań.

- Najczęściej lista jest uporządkowana w kolejności od najważniejszych do najmniej istotnych zadań.
- Za uporządkowanie, zawartość oraz zarządzanie rejestrem produktu odpowiedzialny jest właściciel produktu.
- Istnieje tak długo jak istnieje produkt.
- Rejestr produktu jest dynamiczny.
- Sprawdzenie i dostosowanie rejestru produktu może odbyć się w dowolnym momencie.
- Zdarzeniem, które służy potwierdzeniu adekwatności rejestru produktu jest przegląd sprintu.

Rejestr sprintu

Definicja

Rejestr sprintu (ang. *Sprint Backlog*) – lista zadań odnosząca się do jednego sprintu.

- Zawiera w sobie elementy rejestru sprintu wybranych przez zespół do realizacji w danym sprincie.
- Powstaje podczas planowania sprintu i jest artefaktem, z którego zespół deweloperski korzysta na co dzień podczas wykonywania pracy zaplanowanej na sprint.
- Informuje o tym czym zajmuje się zespół danego dnia.
- Właścicielami rejestru sprintu są deweloperzy.
- O jego kształcie i zawartości decydują deweloperzy.

Przyrost

Definicja

Przyrost (ang. *Increment*) – w pełni ukończona część produktu (przyrost). Jest to oceniony i przetestowany wynik działania pojedynczego sprintu, który przeszedł etap przeglądu. A także suma wszystkich poprzednich ukończonych elementów.

Artefakty – zobowiązania

- **Cel produktu** (ang. *Product Goal*) – buduje kontekst dla całego rejestru produktu. Odpowiada na pytanie “czemu” tworzymy i rozwijamy dany produkt. Element ten określa do czego dąży zespół scrumowy oraz po czym pozna, że to osiągnął.
- **Cel sprintu** (ang. *Sprint Goal*) – zapewnia spójność (jest jeden), wzmacnia skupienie i zachęca do współpracy deweloperów. Jest zobowiązaniem Developerów, ale pozwala na elastyczność w kwestii realizacji elementów ze rejestrów sprintu (nie wszystkie z nich muszą zostać ukończone, żeby udało się osiągnąć cel sprintu).
- **Definicja ukończenia** (ang. *Definition of Done*) – przyrost powstaje w momencie, w którym element rejestr produktu zostaje ukończony. Żeby wszystkie zaangażowane strony wiedziały, kiedy taka sytuacja ma miejsce, kluczowe jest stworzenie oraz przestrzeganie definicji ukończenia.

Zespół scruma

Zespół scrumowy (ang. *Scrum Team*) to działający w sposób autonomiczny i wielozadaniowy zespół osób (zazwyczaj w skład zespołu scrumowego wchodzi mniej niż 10 osób), pracujących razem nad jednym projektem, osoby te mają wszystkie kompetencje niezbędne do wykonania produktu zgodnie z oczekiwaniami klienta.

W skład zespołu scrumowego wchodzi:

- Właściciel produktu,
- Zespół deweloperów,
- Mistrz scruma.

Zespół scruma



Product Owner



Scrum Master



Developers

Zespół scruma – właściciel produktu

Właściciel produktu (ang. *Product Owner*) – to najwyższej usytuowana osoba w procesie produkcji, reprezentująca klienta.

- zarządza kosztami projektu,
- zarządza rejestrem produktu,
- odpowiada za zapewnienie możliwie jak największej wartości produktu,
- konsultowanie z klientem wizji produktu oraz opracowywanie jego specyfikacji,
- kontroluje proces powstawania produktu i koordynuje pracę zespołu odpowiedzialnego za rozwój,
- właściciel produktu może być członkiem zespołu, jednak nie jest zalecane, aby jednocześnie był mistrzem scrum,
- jedna osoba może być właścicielem produktu dla więcej niż jednego zespołu scrumowego, jednak liczba tych zespołów nie powinna być zbyt duża.

Zespół scruma – deweloperzy

Deweloperzy (ang. *Development Team*) – grupa osób, składająca się z od trzech do dziewięciu osób, mających wkład w tworzenie produktu.

- Mają odpowiednie kompetencje do tworzenia produktu,
- zakres ich wiedzy i umiejętności pozwala na pełne spełnienie wymagań produktu,
- są odpowiedzialni za rozwój produktu,
- w sposób samodzielny organizują czas oraz typują zadania konieczne do realizacji,
- w zespole deweloperskim nie ma tytułów innych niż deweloper,
- nie istnieją żadne podzespoły w zespole deweloperskim,
- odpowiedzialność za wykonaną pracę ponosi cały zespół.

Zespół scruma – deweloperzy

Tak więc każdy zespół ma:

- jednego właściciela scruma,
- jednego mistrza scruma,
- jeden rejestr produktu.

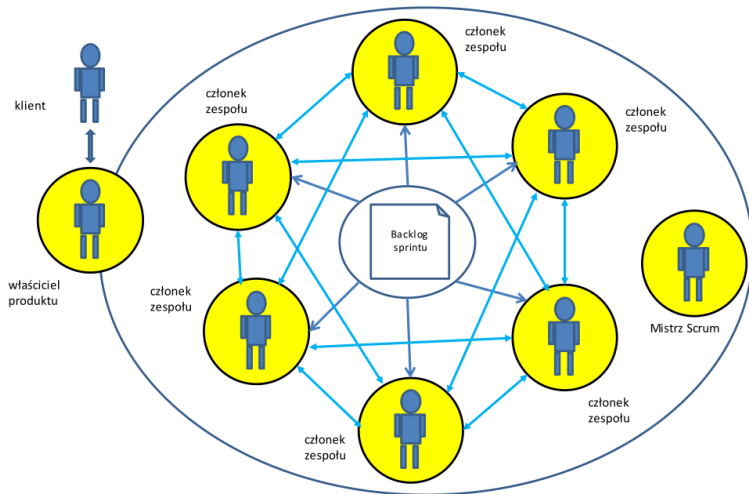
Zespół scruma – mistrz scruma

Definicja

Mistrz scruma (ang. *Scrum Master*) koordynuje stosowanie metodyki scrum przez zespół i właściciela produktu w pracy nad konkretnym projektem.

- nie jest to kierownik zespołu,
- kontroluje pracę zespołu odpowiedzialnego za rozwój,
- odpowiada za poprawną implementację procesu i metod,
- dba o to, by odpowiednio stosował się on do metodyki scrum,
- usuwa wszelkich przeszkody uniemożliwiających zespołowi wykonanie zadania,
- prowadzi treningi i szkolenia z zakresu scrum,
- organizuje spotkania zespołu.

Zespół scruma – relacje



Proces produkcyjny scruma

Proces produkcyjny w metodzie scrum został oparty na idei **Time Boxes**, czyli oparcia planu pracy na sprecyzowanych przedziałach czasu zwanych przebiegami lub sprintami.

Metodyka Scrum zakłada podział procesu tworzenia oprogramowania na pięć zdarzeń:

- 1 Sprint,
- 2 Planowanie sprintu,
- 3 Codzienny scrum,
- 4 Przegląd sprintu,
- 5 Retrospektywa sprintu.

Sprint

Definicja

Sprint (ang. *Sprint*) to ustalony, powtarzalny okres czasu, w czasie którego dostarczany jest gotowy pakiet rozwiązań, przetestowanych i możliwych do zweryfikowania przez właściciela produktu.

Przyjmuje się, że sprint nie powinien być krótszy niż tydzień i nie dłuższy niż miesiąc. Długość sprintu zależy od:

- dostępności klienta,
- wielkości zespołu,
- specyfiki zadań (niektórych zadań nie da się podzielić na mniejsze),
- doświadczenia zespołu w prowadzeniu projektów wg. scrum.

Jeżeli sprint jest za krótki zespół nie zdąży wytworzyć testowalnego produktu. Jeżeli sprint jest za długi tracimy sens założeń Agile oraz spada efektywność pracy.

Sprint (cd.)

Praca zespołu opiera się na:

- wzajemnym wsparciu,
- wzajemnej komunikacji,
- wzajemnym doradzaniu.

Planowanie sprintu

Definicja

Planowanie sprintu (ang. *Sprint Planning*) – jest to pierwsze zdarzenie od którego rozpoczyna się każdy sprint. Planowany jest kolejny inkrement rozwijanego produktu.

W planowaniu sprintu biorą udział:

- Właściciel produktu,
- Zespół deweloperów,
- Mistrz scruma,
- ewentualnie eksperci spoza zespołu.

Planowanie sprintu

Produktem planowania jest **Rejestr sprintu** (ang. *Backlog Sprint*) oraz **Cel sprintu** (ang. *Sprint Goal*).

- Właściciel produktu razem z całym zespołem ustala, które elementy z rejestru produktu przejdą do rejestru sprintu.
- Podczas planowania sprintu powstaje pierwsza wersja rejestru sprintu, jest on modyfikowany w kolejnych dniach pracy zespołu deweloperskiego, pokazuje co już wydarzyło się w sprincie.
- Maksymalny czas trwania planowania to 8 godzin dla sprintu trwającego 4 tygodnie. Przyjmuje się 2 godziny na tydzień sprintu.

Codzienny scrum

Definicja

Codzienny scrum (ang. *Daily Scrum*) – codzienne spotkanie deweloperów, odbywające się przez cały okres sprintu. Potocznie nazywane także **StandUp** – bo odbywa się na stojąco, co ma za zadanie skrócenie czasu jego trwania.

Uczestnicy:

- Zespół deweloperski – w tym zdarzeniu biorą udział wszyscy developerzy, aby wspólnie ułożyć plan na najbliższy dzień roboczy.
- Mistrz scruma – obecność mistrza scruma nie jest wymagana. W początkowej fazie jego zadaniem jest facylitacja spotkania.

Codzienny scrum

Główne zasady codziennego scruma:

- czas trwania maksymalnie 15 minut,
- w tym samym miejscu,
- o tej samej porze,
- pełna obecność zespołu,
- aktualizacja Rejestru sprintu,
- żadnych rozmów pobocznych.

Codzienny scrum – pytania

Podczas codziennego scruma każdy członek zespołu odpowiada na pytania:

- co zrobiłem od ostatniego codziennego scruma?
- jakie są przeszkody?
- co będę robił przez kolejne 24 godziny?

Celem spotkania jest:

- poprawa komunikacji,
- synchronizacja wiedzy,
- zaplanowanie bieżącej pracy,
- rozpoznanie potencjalnych problemów.

Przegląd sprintu

Definicja

Przegląd sprintu (ang. *Sprint Review*) – to wydarzenie, które odbywa się w ostatni dzień każdego sprintu, na którym zespół prezentuje przyrost produktu.

Założenia:

- Czas trwania maksymalnie 4 godziny (dla czterotygodniowego sprintu).
- Dla krótszych sprintów czas przeglądu jest proporcjonalnie mniejszy.
- Przygotowanie prezentacji 1 godzina.

Przegląd sprintu – uczestnicy

W spotkaniu uczestniczy:

- właściciel produktu,
- deweloperzy,
- mistrz scruma,
- przedstawiciele biznesu,
- klienci,
- management,
- deweloperzy z innych projektów.

Przegląd sprintu – uczestnicy (cd.)

- **Deweloperzy** – przekazują co wydarzyło się w sprincie. Omawiają działający przyrost (inspekcja).
- **Właściciel produktu** – jest gospodarzem i narratorem podczas wydarzenia, dba o uczestnictwo odpowiednich osób.
- **Mistrz scruma** – zajmuje się facylitacją całego wydarzenia, jak również dba o utrzymanie odpowiedniego prawidłowego przebiegu wydarzenia z zachowaniem wszystkich jego elementów.
- **Przedstawiciele biznesu, management** oraz **klienci**, czyli osoby dla których budujemy dany produkt (interesariusze) – przekazują swój feedback na temat produktu i dzielą się informacjami napływającymi z otoczenia produktu.
- **Zespół scrumowy** wraz z **interesariuszami** – dokonuje adaptacji rejestru produktu w celu osiągnięcia maksymalizacji wartości i odzwierciedlenia potencjalnych zmian priorytetów, mogących mieć odzwierciedlenie w planowaniu następnego sprintu.

Przegląd sprintu – cele

Cele przeglądu sprintu:

- Prezentacja rzeczywistego, działającego produktu dla klienta.
- Prezentowane są jedynie skończone funkcjonalności.
- Sprawdzenie jakości wyprodukowanego produktu.
- Prace niez zaakceptowane przez klienta wracają do rejestru produktu.
- Identyfikacja ulepszeń, dodanie ich do rejestru produktu.

Retrospekcja sprintu

Definicja

Retrospekcja sprintu (ang. *Sprint Retrospective*) – jest to spotkanie odbywające się po przeglądzie sprintu a przed kolejnym **planowaniem**.

Uczestniczą w nim tylko:

- właściciel produktu,
- deweloperzy,
- mistrz scruma.

i wszyscy uczestnicy spotkania mają takie same zadanie: szukają sposobu jak pracować lepiej.

Czas trwania: 3 godziny dla sprintu 4 tygodniowego. 45 minut na każdy tydzień sprintu.

Retrospekcja sprintu – cele

Celem tego spotkania jest:

- sprawdzenie, co działało się w ostatnim sprincie, biorąc pod uwagę ludzi, relacje, proces i narzędzia,
- zidentyfikowanie i uporządkowanie istotnych elementów, które sprawdziły się w działaniu oraz tych, które kwalifikują się do usprawnienia,
- stworzenie planu wprowadzania w życie usprawnień sposobu wykonywania pracy przez zespół scrumowy już w kolejnym sprincie.

Zalety metodyki Scrum

Wdrożenie metodyki zwinnej może mieć wiele korzyści, nieosiągalnych w przypadku realizacji projektu tradycyjnymi sposobami. Przede wszystkim, metodyka Scrum pozwala na:

- bieżące nanoszenie zmian w projekcie w czasie jego realizacji – zarówno do istniejących, jak i przyszłych funkcjonalności,
- pełen monitoring powstającego produktu – każdy sprint kończy się wypuszczeniem nowej wersji produktu do klienta, którą ten testuje i nanosi na niej poprawki do zrealizowania w kolejnym sprincie,

Zalety metodyki Scrum (cd.)

- Szybszą i skuteczniejszą realizację projektu za sprawą bieżących korekt, a co za tym idzie – lepsze dopasowanie produktu do oczekiwań klienta,
- Zmianę zakresu projektu, a nawet jego zamknięcie w dowolnym momencie – na przykład w sytuacji okrojonego budżetu,
- Zminimalizowanie wydatków związanych z formalnościami – nie ma wymogu tworzenia obszernej dokumentacji projektowej,
- Zwiększenie morale zespołu – dzięki samoorganizacji uczestnicy projektu mają większe pole do eksperymentowania, a ich relacje opierają się na koleżeństwie, co przekłada się na lepszą wydajność

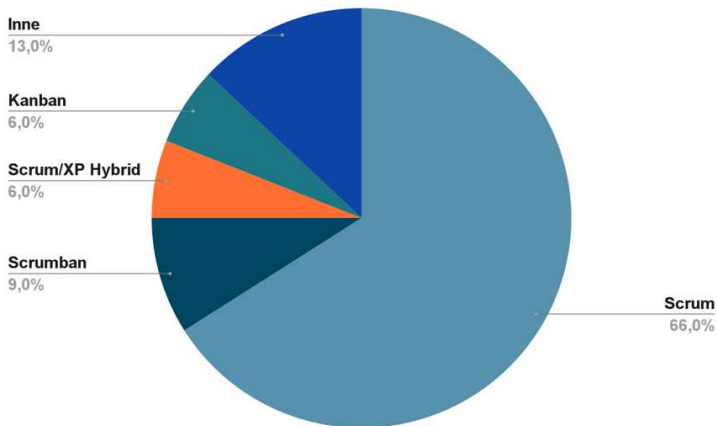
Inne rodzaje metodyk zwinnych

Są inne sposoby na realizację Agile Manifesto, takie jak:

- Extreme Programming (XP),
- Kanban,
- DSDM,
- Crystal,
- Lean Startup.

Jak widać istnieje wiele metod Agile. Jednak raporty takie jak **State of Agile z Version One** czy listy osób, które uzyskały certyfikaty, pokazują jednoznacznie, że to właśnie metodyka Scrum zdominowała świat Agile.

Inne metodyki – adaptacja Scrum



<https://digital.ai/resource-center/analyst-reports/state-of-ag>

Scrum – jak w praktyce?

- 94% programistów używa Scrum,
- Wielkość zespołu to średnio 7 osób.
- Czas sprintu to średnio 2,5 tygodnia.
- Czas pracy nad projektem to średnio 12 tygodni.

<https://www.scrumalliance.org/ScrumRedesignDEVSite/media/Scrum>

Po co nam Scrum?

- Wzrost wydajności o 250%.
- O 42% lepsze utrzymanie produktywności na stałym poziomie.
- Lepszy kontakt z klientem.
- Lepsze samopoczucie pracowników.

<https://docs.broadcom.com/doc/the-impact-of-agile-quantified>