„Agile” jest **nie nakazowe** co oznacza, że samo w sobie nie określa dokładnie JAK uzyskać zwinność. W ciągu ostatnich 40 lat istniało wiele podejść (z lepszymi i gorszymi wynikami) do stworzenia konkretnego przewodnika lub **framework** który powie dokładnie, jak uzyskać zwinność w organizacji.Niektóre z tych ram zostały sformalizowane jeszcze przed Agile Manifesto (2001), ale w swoich najbardziej zaktualizowanych wersjach są zgodne z przesłaniem manifestu.

>

Scrum i jego’fundamenty (Przejrzystość, Inspekcja & Adaptacja)

Scrum jako kompletny framework dla rozwoju oprogramowania został po raz pierwszy opisany i sformalizowany przez dwóch amerykańskich praktyków inżynierii oprogramowania Jeffa Sutherlanda i Kena Schwabera w 1995 roku. (Ken Schwaber był jednym z inżynierów, którzy pierwotnie stworzyli “Agile Manifesto”).

Najnowsza aktualizacja Scrum Guide (dokumentu opisującego Scrum) miała miejsce pod koniec 2020 roku i zmieniła koncepcję Scrum jako frameworka, który może być używany **nie tylko** w tworzeniu oprogramowania. Wszystkie informacje na temat Scrum (Scrum Guide, szkolenia, certyfikacje, baza wiedzy itp.) można znaleźć na stronie www.scrum.org.

Scrum:

- oparty na empiryzmie, który twierdzi, że wiedza pochodzi z doświadczenia i podejmowania decyzji na podstawie tego, co jest obserwowane.
- oparty na myśleniu Lean, aby zredukować marnotrawstwo i skupić się na tym, co najważniejsze.
- stosuje iteracyjne, przyrostowe podejście w celu optymalizacji przewidywalności i kontroli ryzyka;
- angażuje grupy osób, które wspólnie posiadają **wszystkie umiejętności i wiedzę specjalistyczną** do wykonywania pracy i dzielenia się lub nabywania takich umiejętności w razie potrzeby.
- Łączy formalne ***Scrum Events*** dla inspekcji i adaptacji w ramach zdarzenia zawierającego **Sprint.**

.

- Zdarzenia te działają, ponieważ implementują empiryczne filary Scrum, takie jak **przejrzystość, inspekcja i adaptacja**

.

Przejrzystość:

- proces i praca muszą być widoczne zarówno dla osób wykonujących pracę, jak i osób odbierających pracę;

.

Inspekcja:

- ***Artefakty Scrum*** oraz postęp w kierunku uzgodnionych celów muszą być często i sumiennie kontrolowane aby wcześnie wykrywać problemy.

.

- inspekcja bez przejrzystości jest myląca i marnotrawna.
- inspekcja bez adaptacji jest bezcelowa (Scrum musi prowokować zmiany);

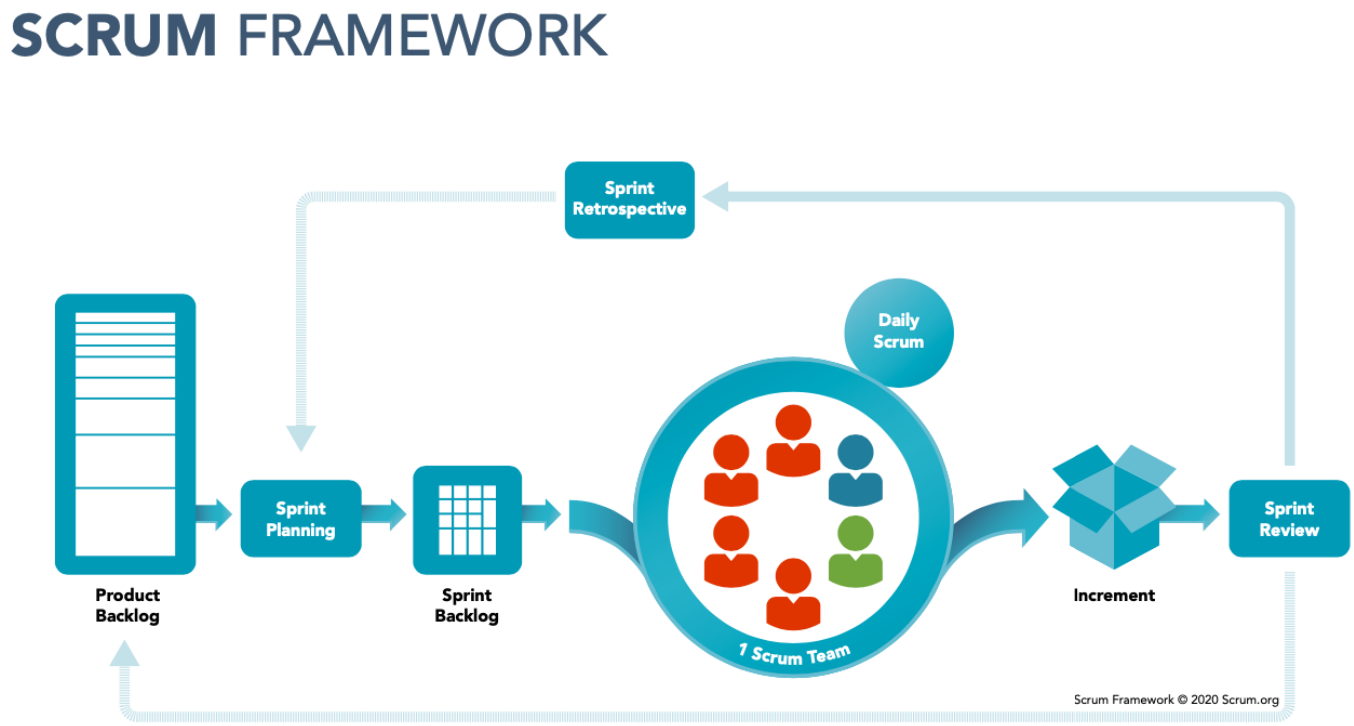
Adaptacja:

- Dostosowanie musi być wykonane tak szybko, jak to możliwe, aby zminimalizować dalsze odchylenia.
- Adaptacja staje się trudniejsza, gdy zaangażowani ludzie nie są upoważnieni lub nie zarządzają sobą.
- A ***Scrum Team*** oczekuje się, że dostosuje się w momencie, gdy dowie się czegoś nowego podczas inspekcji;

.

Ramy Scrum, zdarzenia, artefakty, zespół Scrum

& obowiązki ról



Zespół SCRUM = **Właściciel produktu** + **Scrum Master** + **Deweloperzy**

>

W skrócie, Scrum wymaga od **Scrum Mastera** wspierania środowiska, w którym:

>

1. A **Product Owner** zleca pracę nad złożonym problemem do **Product Backlog**

   .

2. Samoorganizujący się **Zespół Scrum** przekształca wybór pracy w **Increment** wartość podczas **Sprint**.
3. Zespół Scrum i jego **interesariusze** weryfikują wyniki i dostosowują się do następnego Sprintu.

4. Powtórz

   .

Jak widać, nie jest to bardzo skomplikowany proces, ale aby scrum działał poprawnie, muszą istnieć odpowiednie definicje (i **zrozumienie tych definicji**) a) Zdarzeń Scrum, b) Artefaktów Scrum i c) Ról & Obowiązków Zespołu Scrum, d) Obowiązków Interesariuszy

a) Zdarzenia Scrum:

- tworzenie regularności i minimalizowanie potrzeby spotkań niezdefiniowanych w Scrum;
- są ograniczone czasowo, aby nie pozwolić na marnowanie czasu;
- **It's totally OK** organizować inne, niezwiązane ze Scrumem wydarzenia/spotkania, o ile służą one swojemu celowi i nie są stratą czasu;

Sprint – (Zespół Scrumowy):
Wydarzenie Scrumowe, które jest ograniczone czasowo do jednego miesiąca lub krócej, które służy jako kontener dla innych wydarzeń i działań Scrumowych. Sprinty są wykonywane kolejno, bez przerw pośrednich.

>

Planowanie Sprintu – (Zespół Scrumowy):
Zdarzenie Scrumowe trwające maksymalnie 4-8 godzin (lub krócej). Podczas niego Zespół Scrumowy (właściciel produktu + programiści + mistrz Scrum) sprawdza pracę z Backlogu Produktu, która jest najbardziej wartościowa do wykonania w następnej kolejności i projektuje tę pracę w Backlogu Sprintu.

Daily Scrum – (Deweloperzy):
15-minutowe wydarzenie odbywające się każdego dnia. Podczas niego deweloperzy planują pracę na następne 24 godziny i sprawdzają, co zostało zrobione w ciągu ostatnich 24 godzin, aby lepiej przewidzieć, czy Cel Sprintu jest osiągalny. Codzienny Scrum odbywa się o tej samej porze i w tym samym miejscu każdego dnia, aby zmniejszyć złożoność.

Przegląd Sprintu – (Zespół Scrumowy + Interesariusze):
Maksymalnie 4-godzinne wydarzenie podsumowujące prace rozwojowe Sprintu. W

jego trakcie Interesariusze dokonują inspekcji Przyrostu produktu powstałego w wyniku Sprintu i oceniają wpływ wykonanej pracy na ogólny postęp oraz aktualizują Rejestr Produktu w celu zmaksymalizowania wartości następnego okresu/sprintu.

Przegląd Sprintu – (Zespół Scrumowy + Interesariusze):
Maksymalnie 4-godzinne wydarzenie kończące prace rozwojowe.

Retrospektywa Sprintu – (Zespół Scrumowy):
Maksymalnie 3-godzinne wydarzenie kończące Sprint. Podczas niego Zespół Scrumowy dokonuje inspekcji minionego Sprintu i planuje usprawnienia, które mają zostać wprowadzone podczas następnego Sprintu.

Retrospektywa – (Zespół Scrumowy)

b) Artefakty Sprintu:

.

- **reprezentują pracę lub wartość, aby zapewnić przejrzystość i możliwości inspekcji i adaptacji;**
- **są specjalnie zaprojektowane, aby zmaksymalizować przejrzystość kluczowych informacji, tak aby każdy miał takie samo zrozumienie stanu artefaktu;**
- **każdy artefakt jest własnością kogoś, kto jest za niego odpowiedzialny:**

Backlog Produktu – (którego właścicielem jest Właściciel Produktu):

- **Zawiera uporządkowaną listę prac do wykonania w**

celu stworzenia, utrzymania i utrzymania produktu. Zarządzana przez Właściciela Produktu, który jest odpowiedzialny za jasne określenie Celu Produktu.

Sprint Backlog – (należący do deweloperów):

- **Artefakt scrumowy, który zapewnia przegląd prac rozwojowych w celu realizacji celu sprintu, zazwyczaj prognozy funkcjonalności i pracy potrzebnej do dostarczenia tej funkcjonalności. Zarządzany przez deweloperów.**

Increment – (własność Zespołu Scrumowego):

- **Definiuje kompletną i wartościową pracę wytworzoną przez Zespół Deweloperski podczas jednego Sprintu. Praca nie może być uznana za część Przyrostu, jeśli nie spełnia Definicji Ukończenia. Suma wszystkich Przyrostów tworzy produkt**

.

c) Role i obowiązki Zespołu Scrumowego

- **określenie, kto jest odpowiedzialny za co i dlaczego;**

- **W ramach Zespołu Scrum nie ma podzespołów ani hierarchii. Jest to spójna jednostka profesjonalistów skupionych na jednym celu w danym czasie, czyli na Celu Produktu.**

Deweloperzy:

- **Deweloperzy to osoby w Zespole Scrumowym, które są zaangażowane w tworzenie dowolnego aspektu użytecznego Przyrostu w każdym Sprincie;**
- **specyficzne umiejętności potrzebne Deweloperom są często szerokie i będą się różnić w zależności od domeny pracy**
- **Tworzenie planu dla Sprintu, Backlogu Sprintu;**
- **zastosowanie Definicji ukończenia;**
- **Dostosowując swój plan każdego dnia do Celu Sprintu**

  .

- **Wzajemne rozliczanie się jako profesjonaliści.**

Właściciel produktu

- **Cała organizacja (w tym interesariusze) musi szanować**
- **Decyzje Właściciela Produktu!**
- **Właściciel Produktu to jedna osoba, a nie komitet.**
- **Właściciel Produktu może reprezentować potrzeby**

wielu interesariuszy w Backlogu Produktu.

- **Osoby chcące zmienić Backlog Produktu mogą to zrobić próbując przekonać Właściciela Produktu.**

.

Obowiązki Właściciela Produktu

- **Odpowiedzialność za maksymalizację wartości produktu poprzez zarządzanie i wyrażanie oczekiwań biznesowych i funkcjonalnych dotyczących produktu dla programistów;**
- **Rozwijanie i wyraźne komunikowanie celu produktu;**
- **Tworzenie i wyraźne komunikowanie elementów Backlogu Produktu;**
- **Porządkowanie elementów rejestru produktowego;**
- **Zapewnienie, że Backlog Produktu jest przejrzysty, widoczny i zrozumiały.**
- **Właściciel Produktu może wykonać powyższą pracę lub może delegować odpowiedzialność na inne osoby. Niezależnie od tego, Właściciel Produktu pozostaje odpowiedzialny.**

Scrum Master:

- **odpowiedzialny za prowadzenie, coaching, nauczanie i pomaganie Zespołowi Scrum, Kluczowym Interesariuszom i ich otoczeniu we właściwym zrozumieniu i wykorzystaniu Scrum.**
- **tworzenie Definicji Wykonania i utrzymywanie jej.**

d) Role i obowiązki interesariuszy

- **osoba zewnętrzna w stosunku do Zespołu Scrumowego, posiadająca konkretne zainteresowanie i wiedzę na temat produktu, która jest wymagana do jego przyrostowego rozwoju (będzie to głównie Sponsor Produktu/Wynalazca, ale mogą to być również klienci w modelu B2C);**
- **Interesariusze muszą być reprezentowani przez Właściciela Produktu (lub muszą delegować jego obowiązki na kogoś);**
- **Muszą być aktywnie zaangażowani w prace Zespołu Scrumowego podczas Przeglądu Sprintu;**
- **Interesariusze muszą szanować Product Owner’s decyzje!**

.

Definition of Done (DoD)

- **formalny opis miar jakości, które są wymagane, aby pozycja zaległości produktu stała się**

Przyrostem

- **Jeśli element Rejestru Produktu nie spełnia DoD, nie może zostać wydany ani nawet zaprezentowany podczas Przeglądu Sprintu. Zamiast tego powraca do Rejestru Produktu do rozważenia w przyszłości.**

Wartości Scrum

Czy trudniej jest Scrum Masterowi wdrożyć pewne ramy “jak pracujemy” czy zmienić całą kulturę firmy? Kultura firmy jest czymś, co jest definiowane głównie na samym początku jej istnienia, dlatego najtrudniejszą częścią pracy Scrum Mastera podczas wdrażania scruma jest “wspieranie środowiska”. Oznacza to położenie silnych fundamentów w kulturze firmy, aby Scrum mógł w ogóle zaistnieć. Te fundamenty to: Odwaga, Skupienie, Zaangażowanie, Szacunek i Otwartość.



Podsumowanie: Plusy, minusy Scrum

Scrum zapewnia przejrzystą iterację strukturę dla:

- **rozwoju i utrzymania złożonych produktów (na przykład rozwój oprogramowania);**
- **płynne dostarczanie wartości do klienta końcowego;**
- **tworzenie pętli sprzężenia zwrotnego od klienta końcowego do zespołu produkcyjnego**
- **dostosowanie do zmian w wymaganiach produktowych**

.

Zalety Scrum:

>

- **Scrum precyzyjnie określa, jak ustawić strukturę iteracji dla rozwoju produktu. Jeśli firma nie ma żadnej struktury rozwoju produktu, Scrum jest najprawdopodobniej najłatwiejszym sposobem wprowadzenia do Agility (jeśli Agile jest wartościową koncepcją rozwoju takiego produktu);**

.

- **Scrum daje szansę na dostosowanie, jeśli zmienią się wymagania produktu Istnienie Sprintu podzielonego na pudełka czasowe ogranicza te najnowsze zmiany do uzgodnienia tylko podczas każdego planowania sprintu (powinniśmy unikać zmiany wymagań podczas trwającego sprintu!). Jest to bardzo pomocne, jeśli pracujemy z klientem, który ciągle zmienia swoje podejście do każdego elementu planowanego rozwoju, przez co ponosimy koszty ciągłego przełączania kontekstu & non-stop research. Przekonanie klienta do korzystania ze Scrum i utrzymywanie go to dobry sposób na ulepszenie produktu.**

Konsekwencje Scrum:

- **Scrum precyzyjnie określa, jak ustawić strukturę iteracji dlatego kuszące jest, aby po prostu ją zastosować i nie myśleć o konsekwencjach. Zastosowanie Scruma wszędzie może wydawać się łatwe, ale w rzeczywistości najtrudniejszą częścią jest zmiana zachowania ludzi, którzy używają tego frameworka. Ciągłe skupianie się tylko na niektórych aspektach Scruma może skutkować Kultem cargo.**

.

- **Scrum daje szansę na dostosowanie, jeśli zmienią się wymagania dotyczące produktu Ale co jeśli zbieranie informacji zwrotnych od klienta/sponsora projektu odbywa się tylko podczas przeglądu sprintu na koniec 4-tygodniowego sprintu?Jest wysoce prawdopodobne, że wymagania uległy zmianie, ale programiści nie mieli o tym pojęcia i dostarczyli przyrost, który ma pewną wartość, ale mógłby mieć znacznie większą! Adaptacja do zmian wymagań nie dzieje się magicznie po prostu przez zastosowanie scruma lub jakiegokolwiek innego frameworka. Jeśli wartości takie jak przejrzystość i otwartość nie są kultywowane w organizacji, adaptacja do jakiejkolwiek zmiany będzie prawie niemożliwa.**