

Scrum

Pierwsze kroki

Agenda

✓ Empiryzm

✓ Agile vs Waterfall

✓ Agile Manifesto

✓ Scrum

Agenda

- ✓ Role w Scrumie
- ✓ Zdarzenia w Scrum
- ✓ Artefakty w Scrum
- ✓ Śledzenie postępów prac

Proces
empiryczny

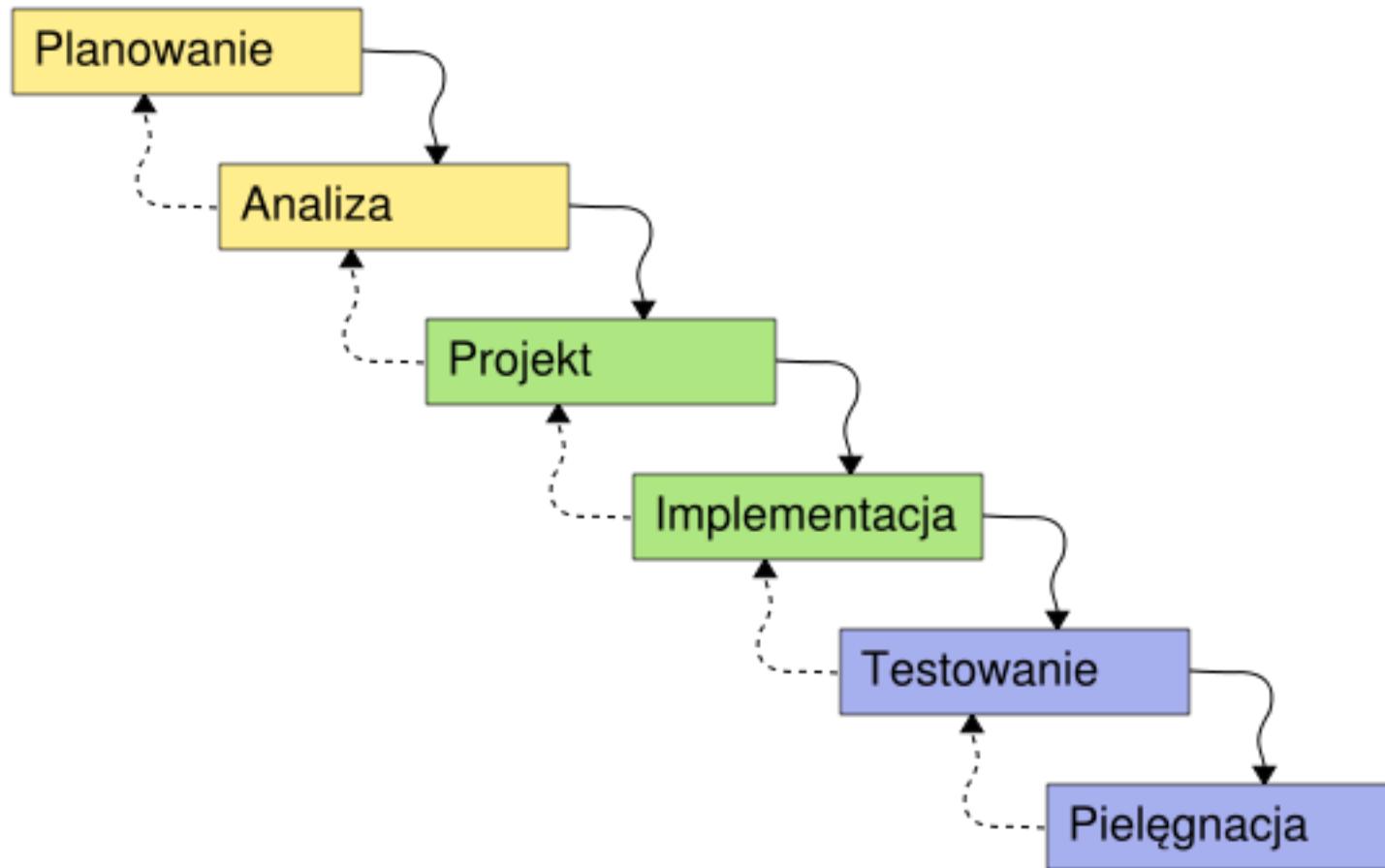
Podejście tradycyjne

- ✓ Podstawową tradycyjnych metodyk zarządzania projektami jest podejście predykcyjne – opierające się na predykcji
- ✓ Najpierw projektujemy kompletne rozwiązanie, a następnie przewidujemy jak będzie wyglądać realizacja

Waterfall – Kaskadowy proces



Kaskadowy model wytwarzania oprogramowania



Proces empiryczny

- ✓ Scrum opiera się na teorii empiryzmu.
- ✓ Teoria empiryzmu - wiedza wynika z doświadczenia i podejmowania decyzji w oparciu o to co poznane.
- ✓ Stosowane dla procesów
 - ✓ Kiedy nie wiemy jak dokładnie będzie wyglądał produkt kiedy zaczynamy
 - ✓ Chcemy kontrolować rezultat pracy i utrzymywać jakość na wysokim poziomie
 - ✓ Kroki nie zawsze są powtarzalne

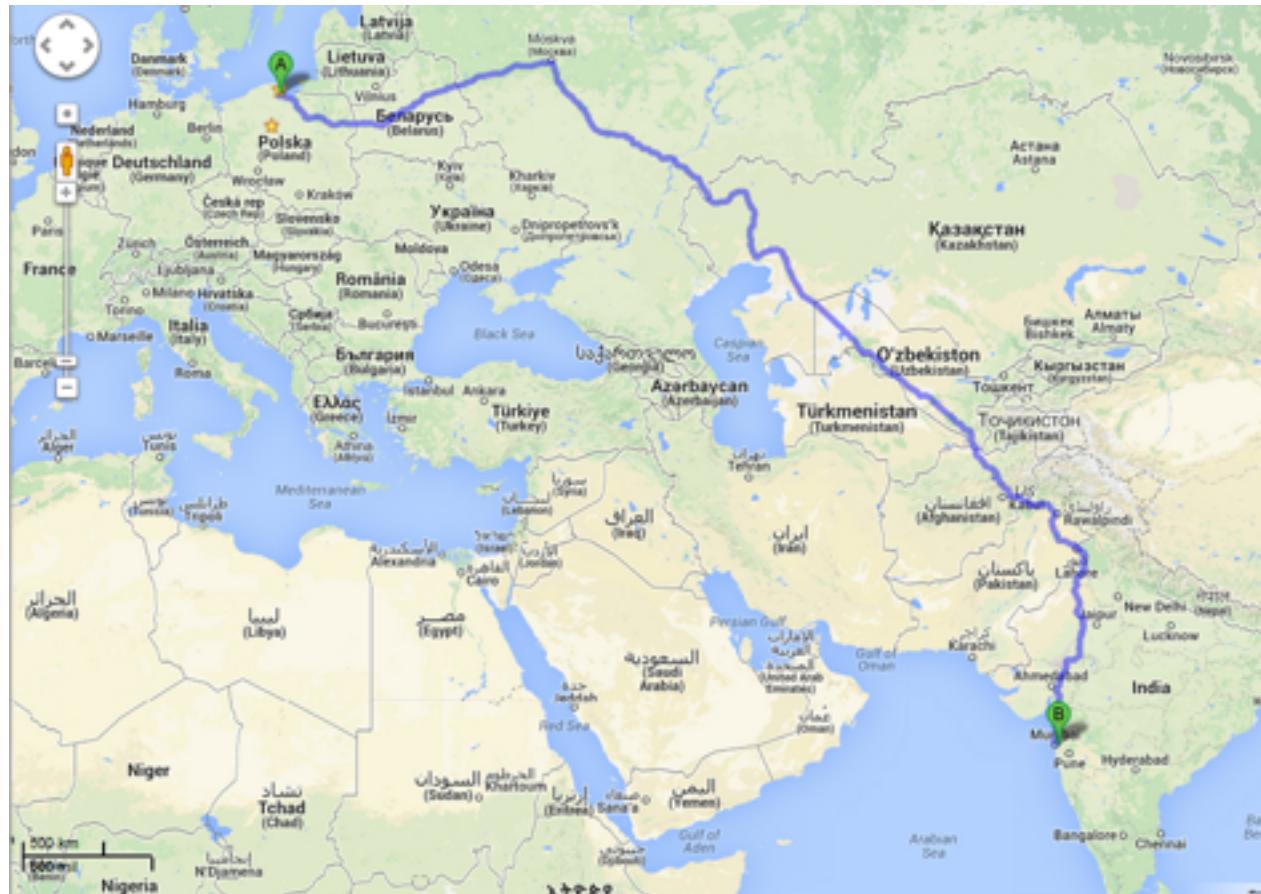
Fundamenty procesu empirycznego

- ✓ Przejrzystość (ang. transparency) – każdy element procesu musi być jawnym i zrozumiałym dla wszystkich
 - ✓ np. wszystkie nazwy powinny być zrozumiałe dla wszystkich uczestników procesu
 - ✓ osoby wykonujące pracę oraz osoby akceptujące wyniki tej pracy muszą posiadać wspólną definicję „Ukończonej pracy”
- ✓ Inspekcja – każdy element procesu musi być cyklicznie poddawany przeglądowi w celu zlokalizowania przeszkód, które powodują że proces staje się mniej optymalny
- ✓ Adaptacja – po zlokalizowaniu przeszkody w procesie należy tak szybko jak to możliwe wyeliminować ją

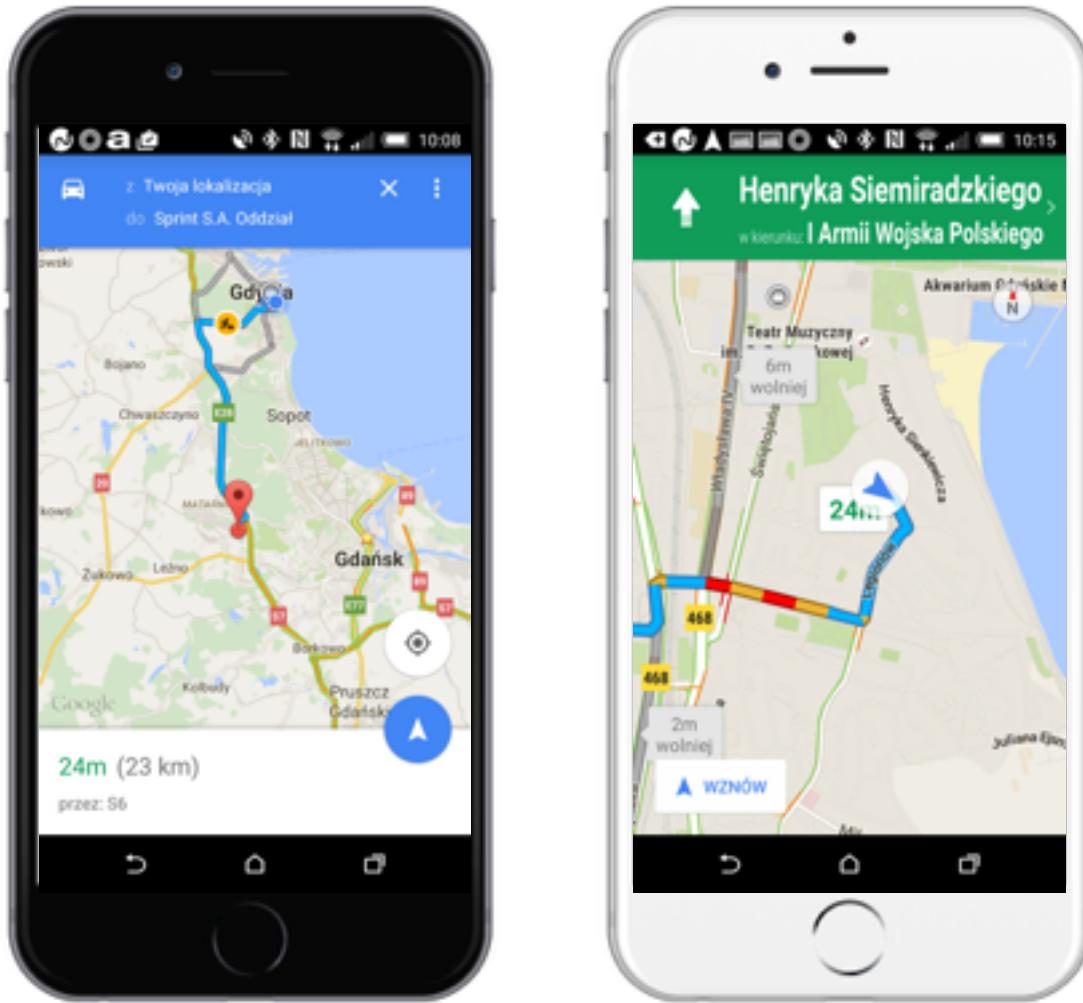
Proces adaptacyjny

- ✓ Dla procesów, których przebiegów nie można z góry przewidzieć i dokładnie zdefiniować
- ✓ Które generują nieprzewidywalne oraz niepowtarzalne wyniki
- ✓ Umożliwia szybkie radzenie sobie ze zmiana

Jak długo zajmie dojście samochodem do Mombaju w Indiach?

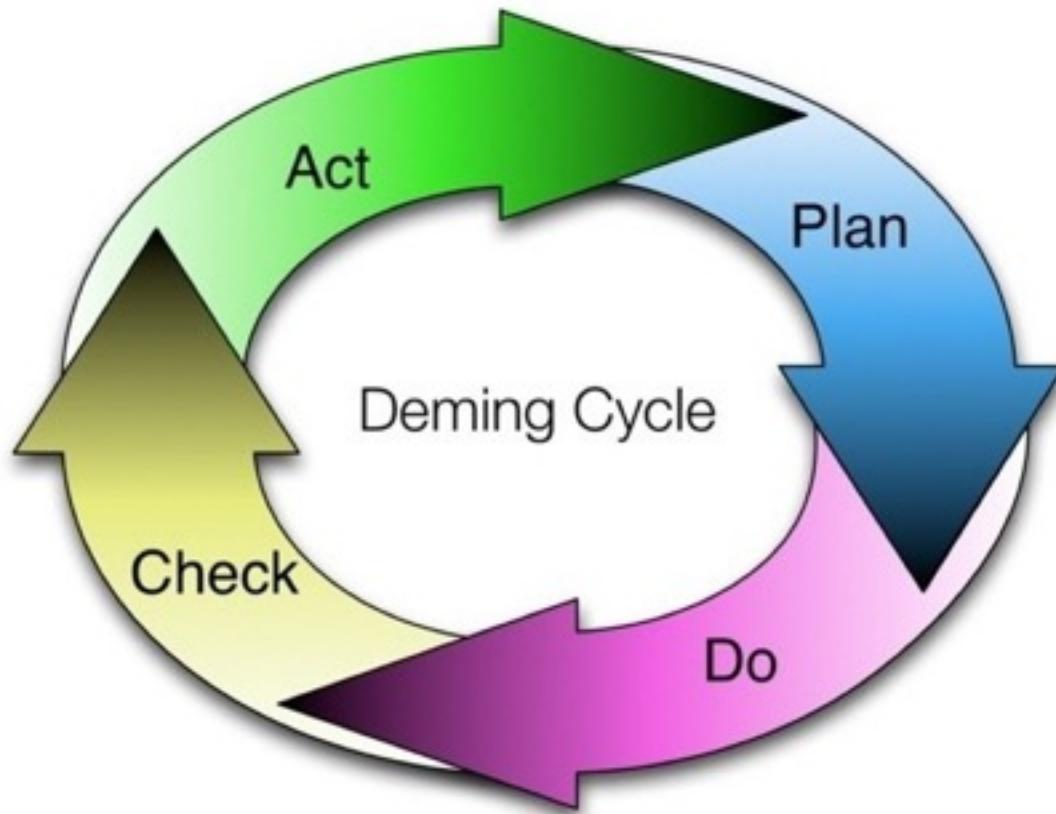


Adaptacyjność – nawigacja samochodowa



Cykl Deminga

Empirical Process



Adaptacyjność w projektach

Tworzenie produktu jest procesem

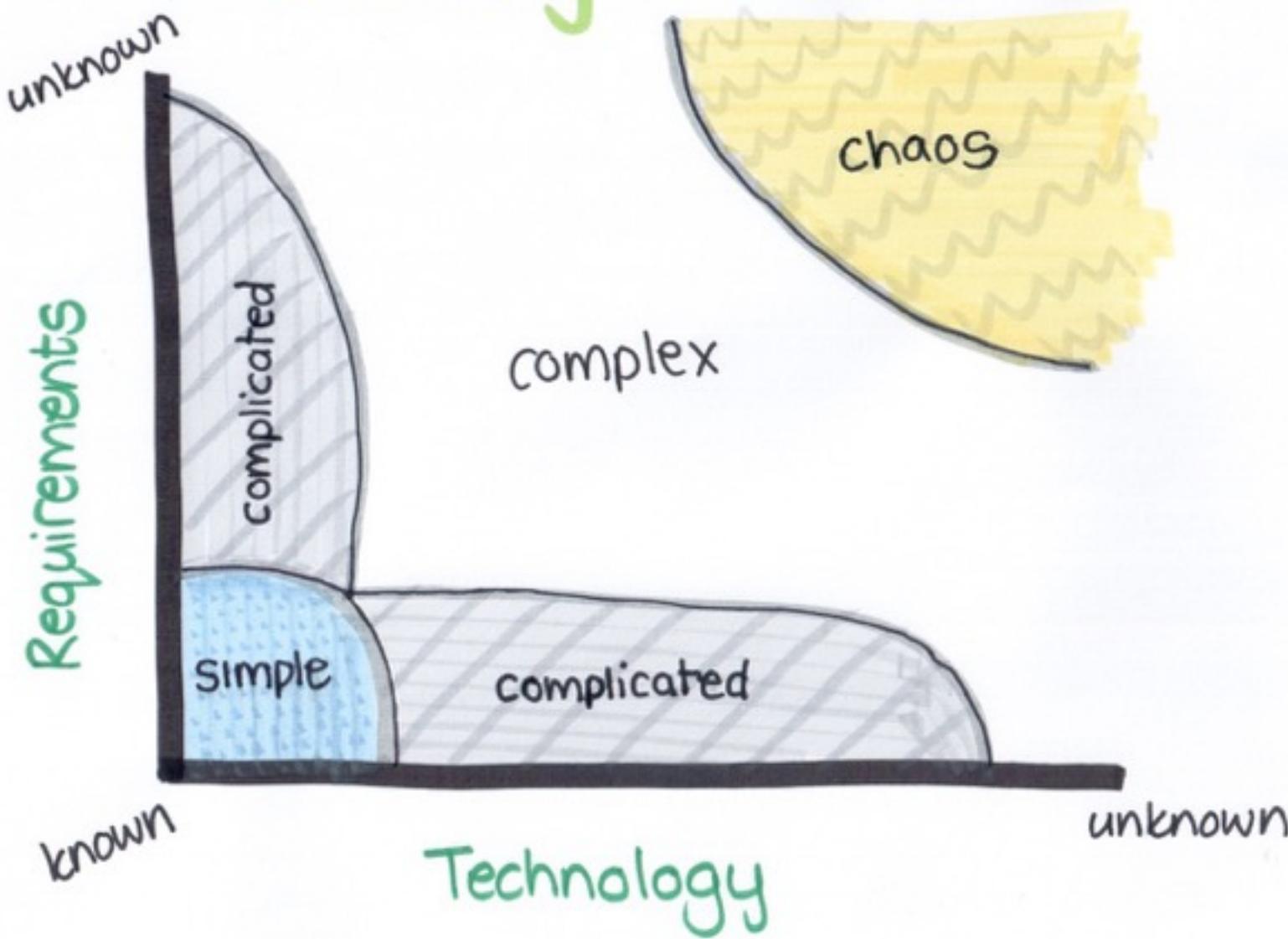
- ✓ twórczym (więc nie powtarzalnym)
- ✓ nie stabilnym (wymagania się zmieniają)
- ✓ o dużym stopniu złożoności

Adaptacyjność w projektach

Adaptacyjność stosowane jest na różnych poziomach

- ✓ codzienne spotkania zespołu
- ✓ wizualizacji pracy za pomocą tablic i wykresów
- ✓ częstym oddawaniu działającego oprogramowania
- ✓ częstym retrospekcjom

Stacey Matrix





www.projectcartoon.com

Jak opisał to klient



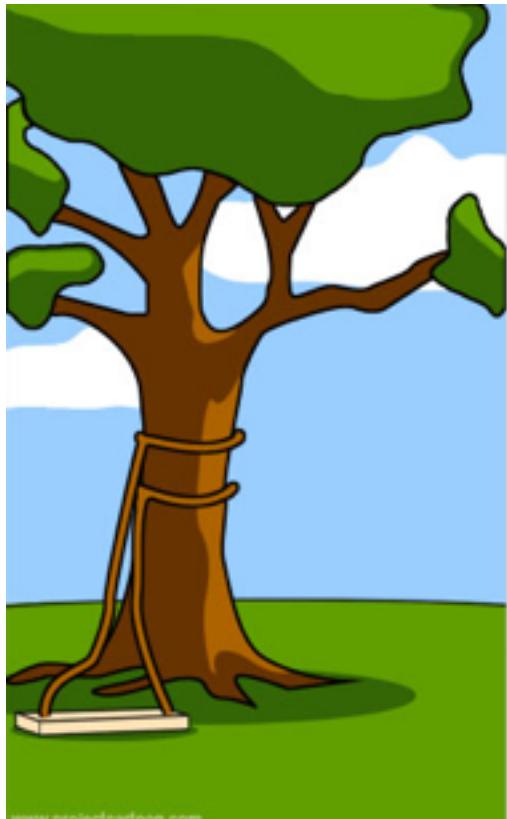
www.projectcartoon.com

Jak zrozumiał to kierownik projektu

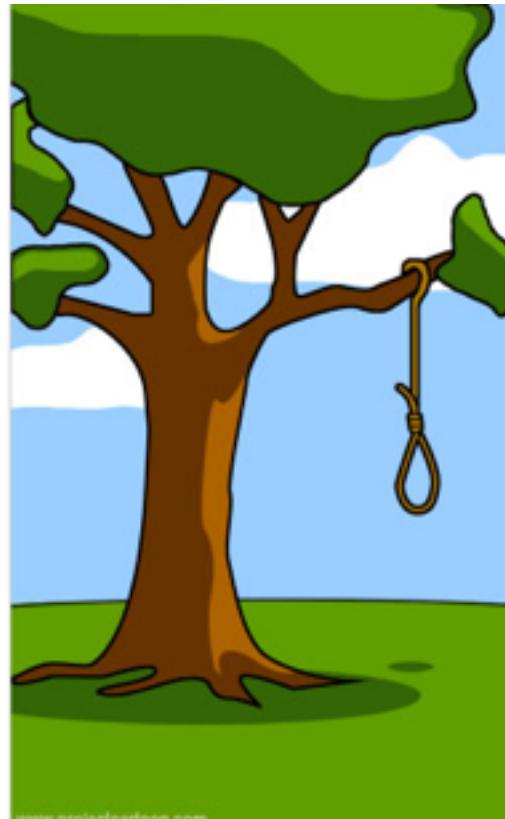


www.projectcartoon.com

Jak zaprojektował to analityk



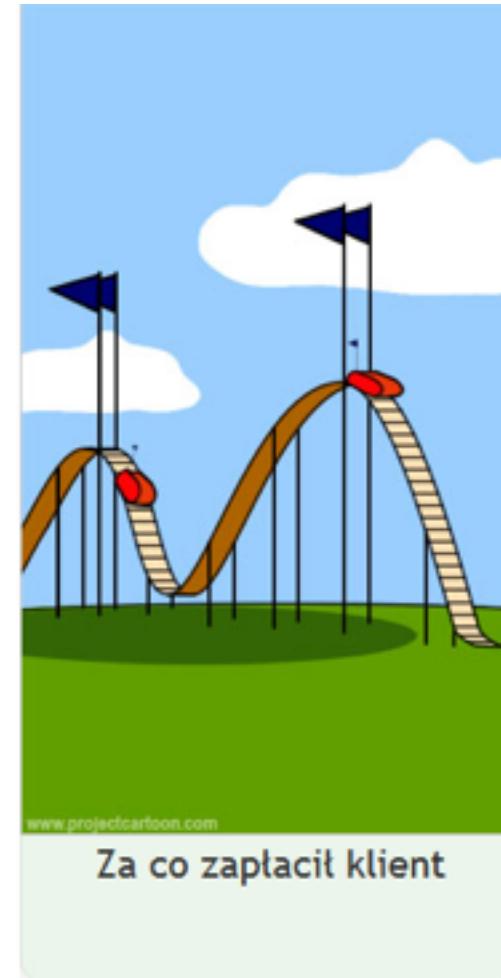
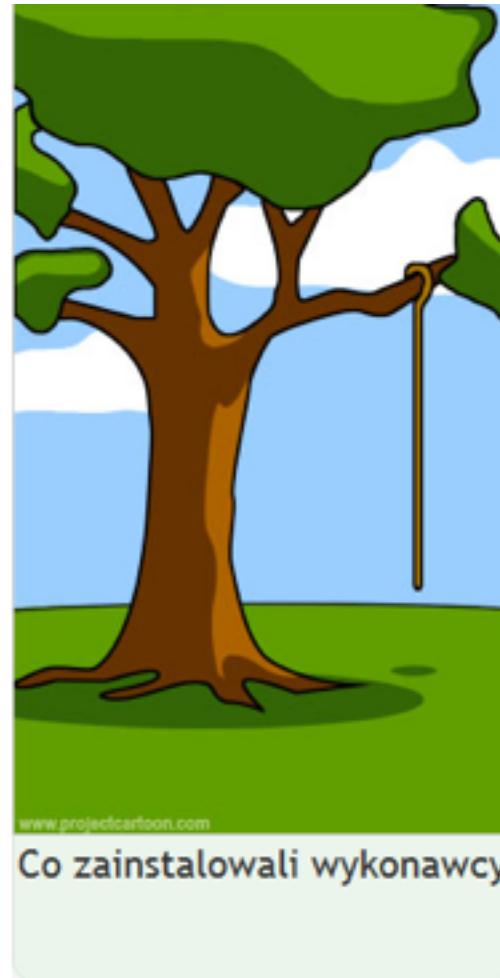
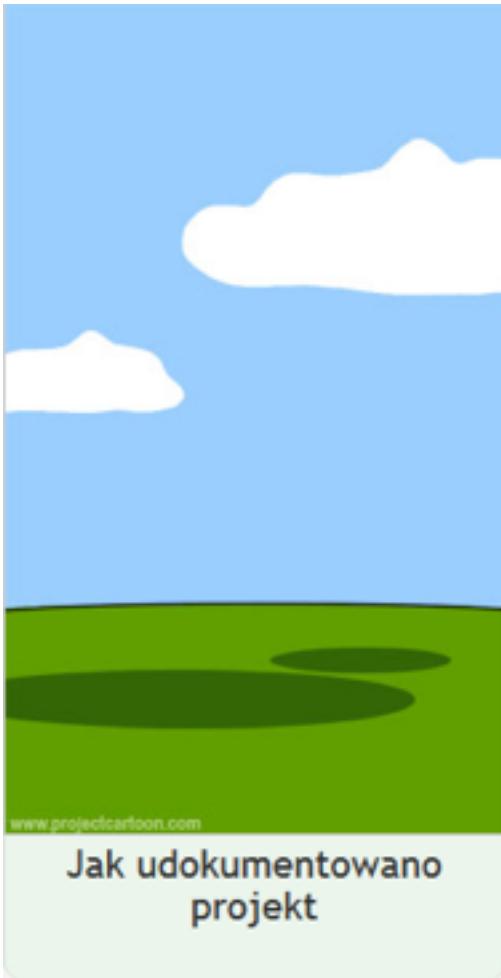
Jak napisał to programista

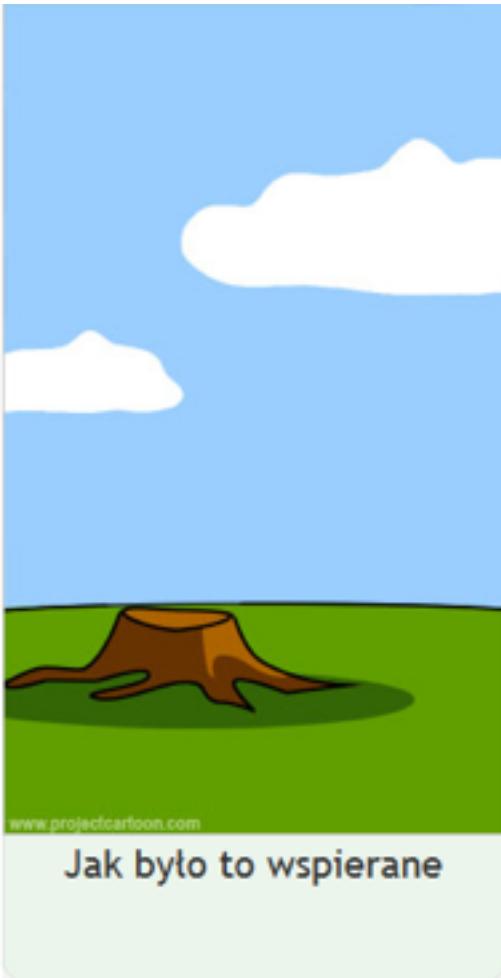


Co otrzymali betatesterzy



Jak opisał to doradca
biznesowy





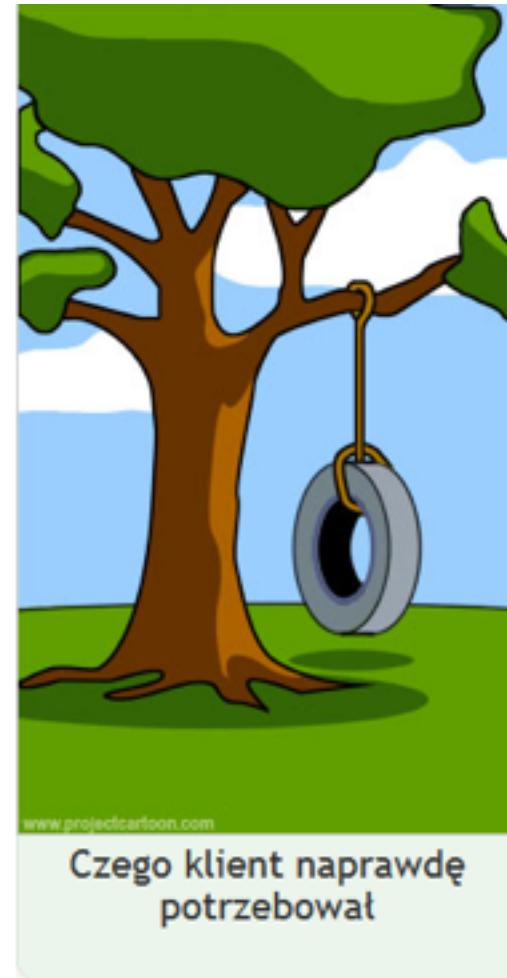
www.projectcartoon.com

Jak było to wspierane



www.projectcartoon.com

Co reklamował marketing



www.projectcartoon.com

Czego klient naprawdę potrzebował

Zwinność

Zwinność (agility) oznacza (1) zdolność łatwego i szybkiego poruszania się; lekkość, zwrotność, szybkość; (2) zdolność do szybkiego myślenia i wyciągania wniosków; (3) elastyczność i potencjał – w sferze umiejętności i możliwości – do szybkiego i sprawnego dostosowywania się do zachodzących zmian, a także...

...odwagę bycia wystarczająco szczerym by przyznać, iż budowanie oprogramowania jest procesem złożonym, który nie może zostać perfekcyjnie zaplanowany ze względu na zmieniające się otoczenie, potrzeby, wymagania i założenia.

Tradycyjne metody wytwarzania

WATERFALL

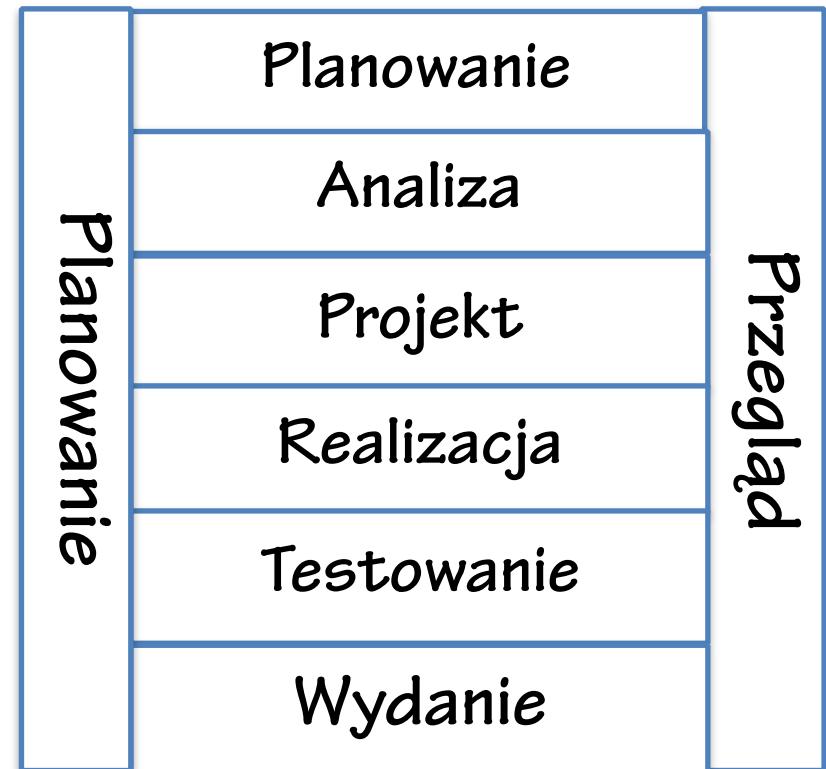


- ✓ planujemy cały projekt na początku
- ✓ nic nie nadaje się do użycia zanim projekt nie zostanie skończony

Scrum jest narzędziem dzięki któremu możemy stać się zwinnym

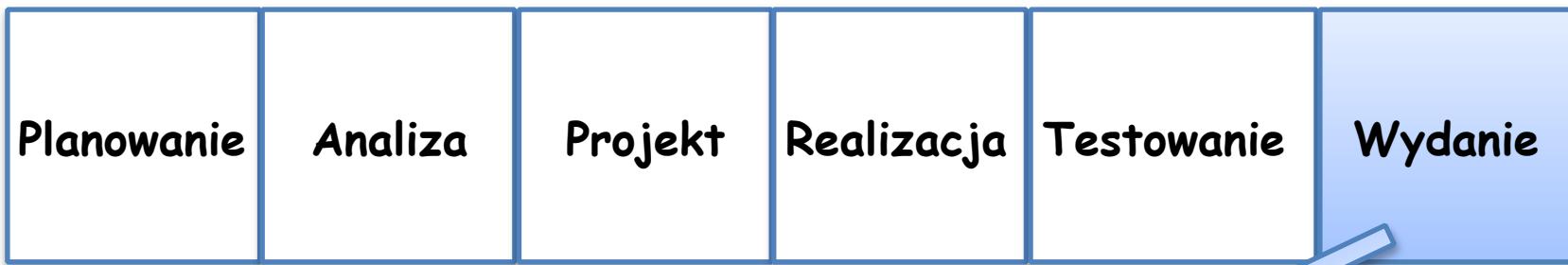
SCRUM

- ✓ krótkie iteracje o wysokiej wartości
- ✓ regularna inspekcja
- ✓ jest szansa na usprawnienia

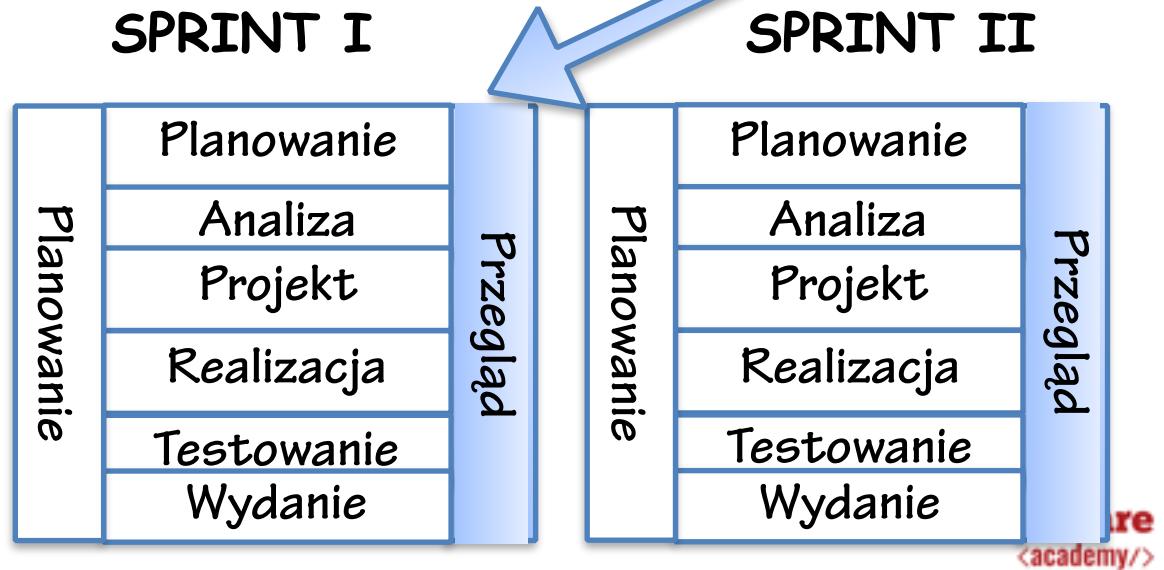


Ryzyko ograniczone jest do jednego Sprintu

WATERFALL

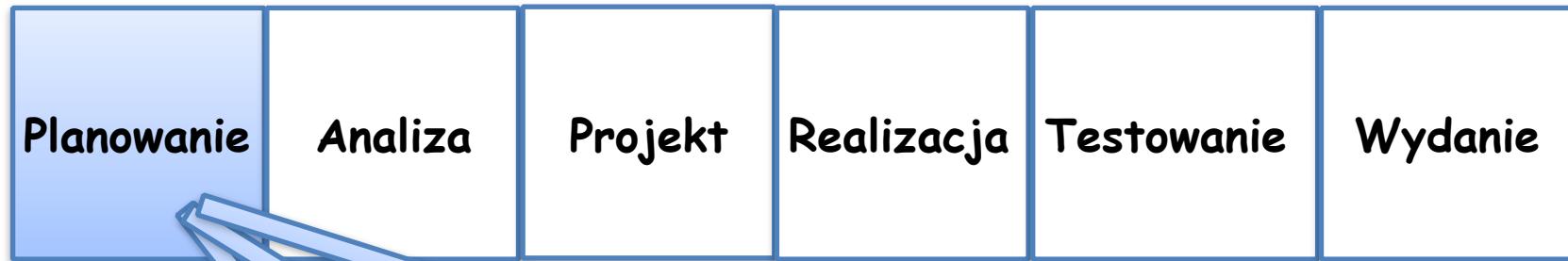


SCRUM



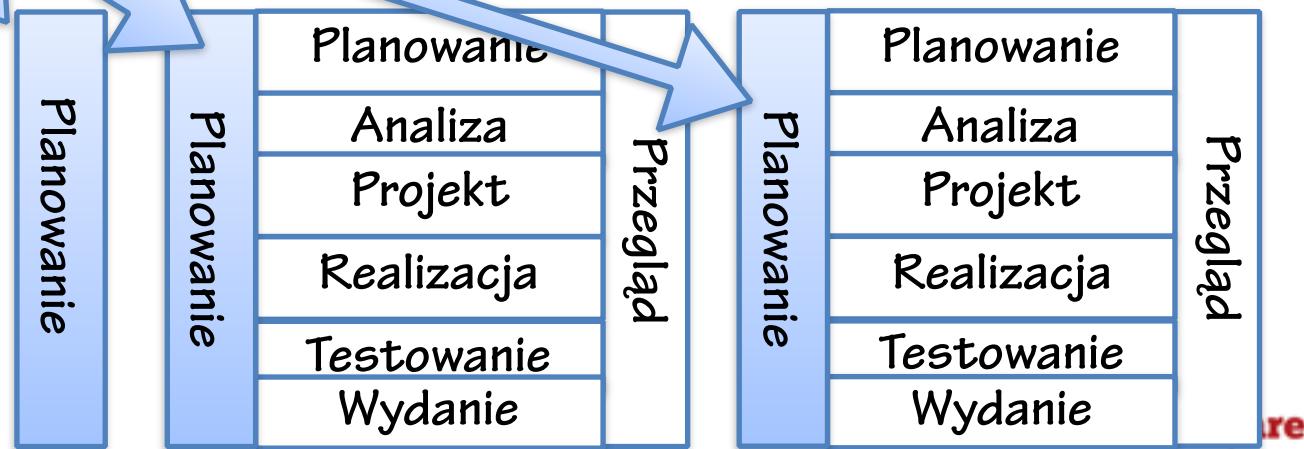
W Scrumie dostarczamy
szybciej niż w tradycyjnych metodach

WATERFALL



SCRUM

Planujemy trochę na
początku i przed
każdym Sprintem



Agile Manifesto

Agile - historia

- ✓ Manifest Agile podpisany w 2001 roku w Snowbird (kurort górski w stanie Utah)
- ✓ 17 osób , doświadczonych praktyków oprogramowani spotkało się by podyskutować na temat innych niż kaskadowe metody rozwoju oprogramowania
- ✓ Manifest jest dostępny pod adresem [http://
www.agilemanifesto.org/](http://www.agilemanifesto.org/)

Manifest Agile

Odkrywamy lepsze sposoby tworzenia oprogramowania poprzez wytwarzanie go oraz pomaganie w tym innym.

W wyniku zebranych doświadczeń nauczyliśmy się przedkładać:

Manifest Agile

Ludzi i interakcje pomiędzy nimi
ponad
procesy i narzędzia.

Manifest Agile

Działające oprogramowanie
ponad
Wyczerpującą dokumentację.

Manifest Agile

Współpracę z klientem
ponad
negocjowanie umów.

Manifest Agile

Reagowanie na zmianę

ponad

trzymanie się planu

12 zasad w Manifeście Agile

- ✓ Najważniejsze jest dla nas zadowolenie klienta poprzez wczesne i stałe dostarczanie mu działającego oprogramowania.
- ✓ Jesteśmy otwarci na zmiany wymagań, nawet jeśli prace są już zaawansowane. Procesy zwinne (agile) wykorzystują adaptację do zmiany by dać klientowi przewagę konkurencyjną.
- ✓ Dostarczamy działającego oprogramowania często, co kilka tygodni do kilku miesięcy przy czym preferujemy robienie tego jak najczęściej.
- ✓ Programiści i biznes (klienci) muszą ze sobą codziennie współpracować przez cały czas trwania projektu.

12 zasad w Manifeście Agile

- ✓ Budujemy projekty wokół silnie zmotywowanych ludzi. By osiągnąć cel obdarzamy ich zaufaniem oraz zapewniamy środowisko pracy i wsparcie, jakiego potrzebują.
- ✓ Najbardziej skuteczną metodą przekazywania informacji wewnętrz i na zewnątrz zespołu developerskiego są rozmowy twarzą w twarz.
- ✓ Działające oprogramowanie jest podstawową miarą postępów prac.
- ✓ Procesy zwinne (agile) promują pracę długodystansową. Jej tempo winno być takie, by sponsorzy, developerzy i użytkownicy mogli ją utrzymywać bez końca.

12 zasad w Manifeście Agile

- ✓ Stała dbałość o techniczną doskonałość i dobre projektowanie wspierają zwinność (agility).
- ✓ Prostota - sztuka maksymalizacji pracy, której wykonywać nie trzeba - ma zasadnicze znaczenie.

12 zasad w Manifeście Agile

- ✓ Najlepsze projekty architektury, wymagania i rozwiązania wyłaniają się z pracy samo-organizujących się zespołów.
- ✓ Regularnie co pewien czas zespół powinien się zastanawiać jak stać się jeszcze bardziej efektywnym, a następnie odpowiednio dostosowywać swój sposób pracy.

Agile znaczy zwinny

- ✓ Agile nie jest metodyką ani procesem
- ✓ Agile to filozofia zbierająca różne zasady i praktyki dotyczące rozwoju produktu.

Nie ma jednej definicji Agile

- ✓ Agile można porównać do hodowania drzewa. Najpierw ziarenko, które się rozrasta w przeciwnieństwie do budowania etapowego zaczynając od czubka i liści a potem dopiero korzeni.

AGILE

XP

SCRUM

KANBAN

TDD

Refactoring

BDD

Story
Points

User
stories

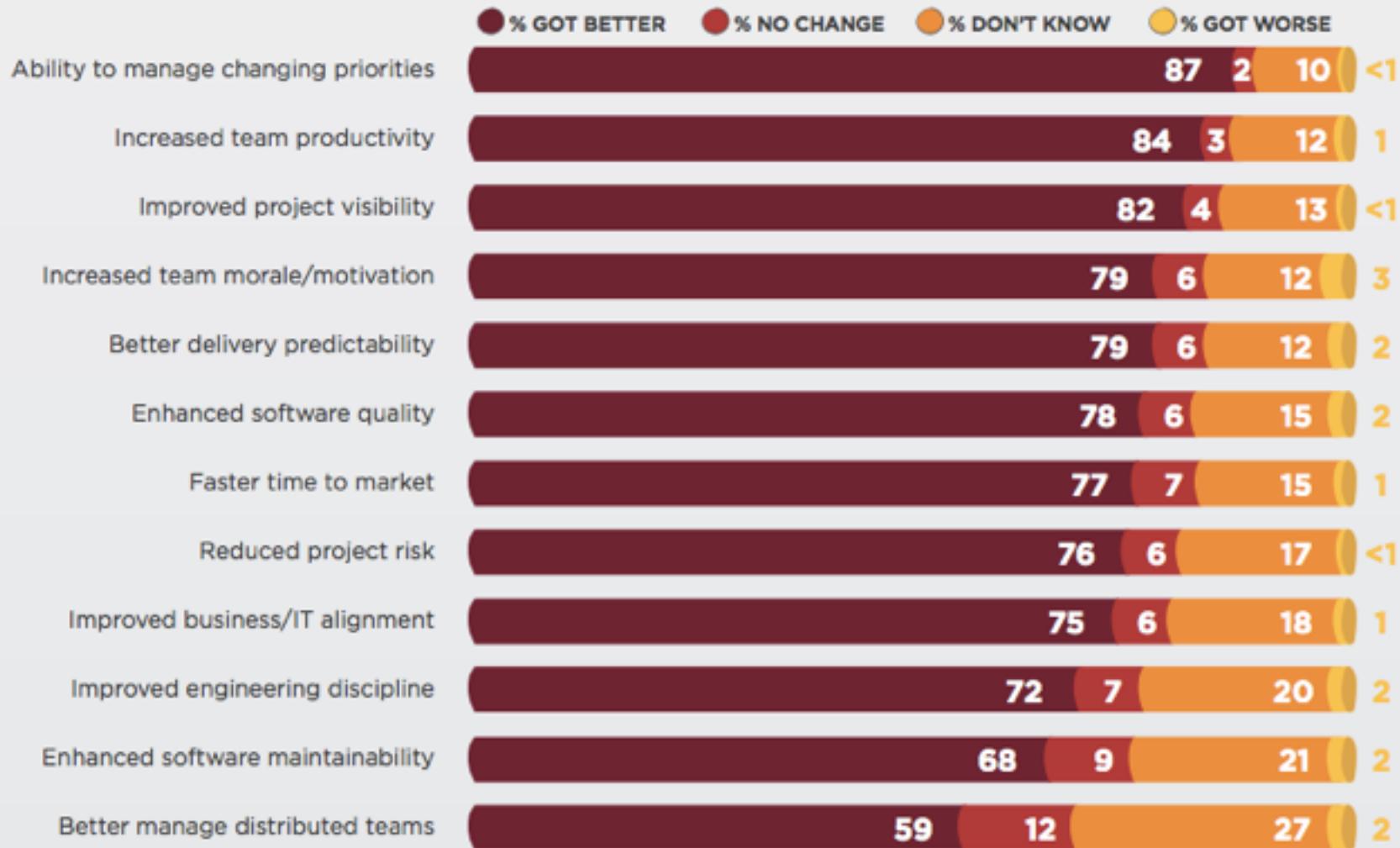
Pairing

ATDD

Metodyki lub
"Frameworki"

Praktyki

Czemu organizacje decydują się na Agile?



źródło: 9th Annual State of Agile Survey

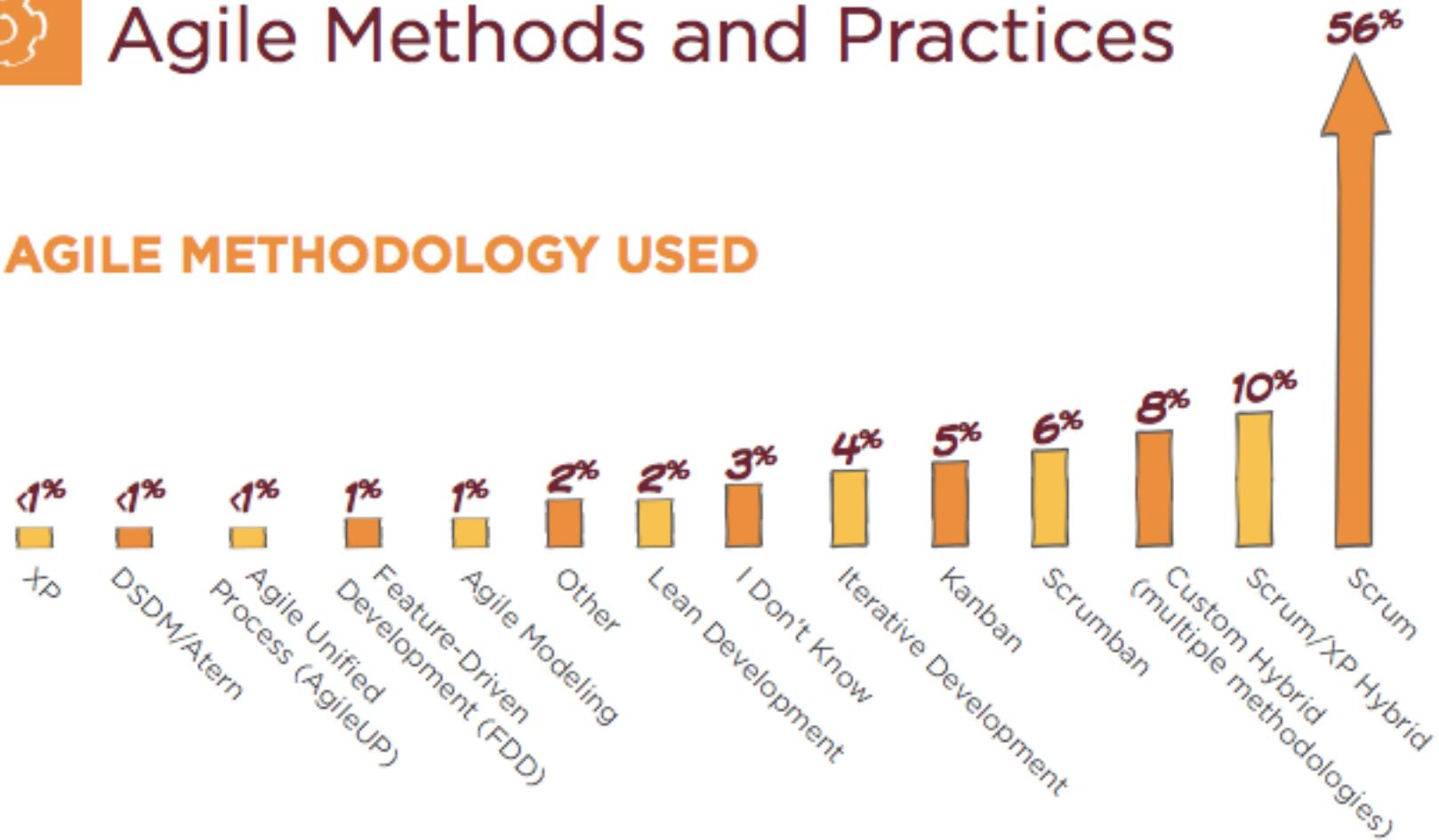
Popularne praktyki i metodyki



STATE OF AGILE

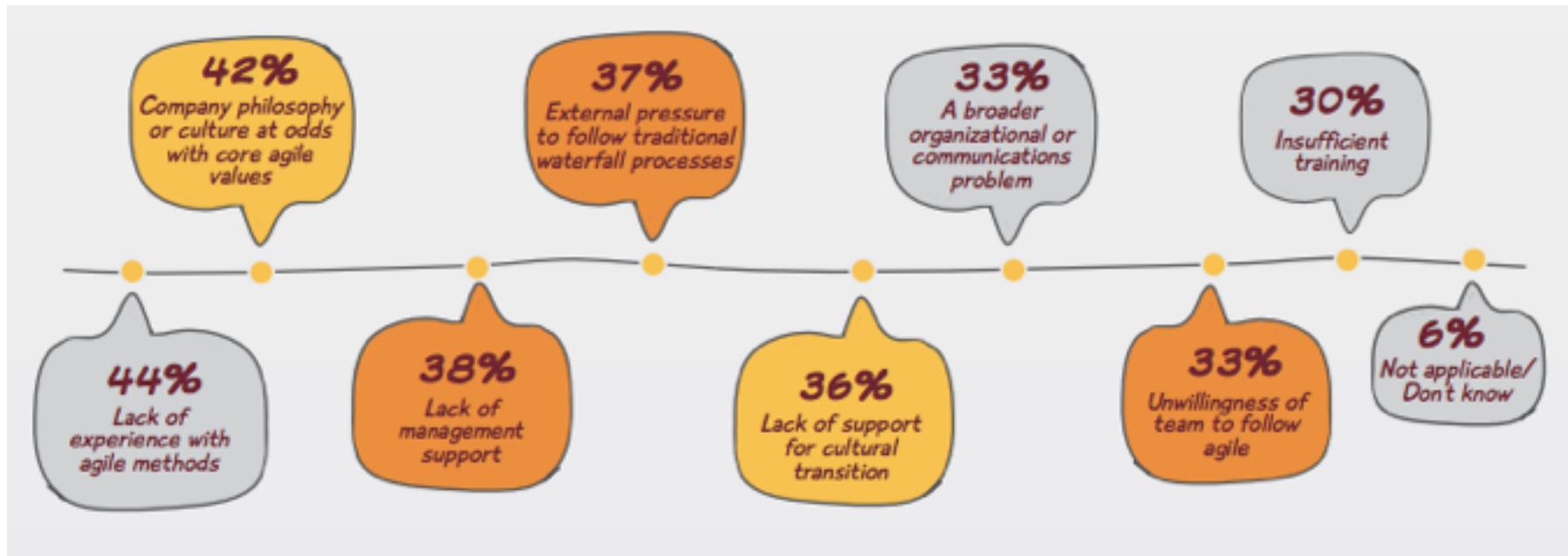
Agile Methods and Practices

AGILE METHODOLOGY USED



źródło: 9th Annual State of Agile Survey

Powody niepowodzeń we wdrożeniach Agile



źródło: 9th Annual State of Agile Survey

Scrum

Scrum

- ✓ Scrum został stworzony przez Kena Schwabera i Jeffa Sutherlanda na przełomie ubiegłego i bieżącego wieku



- ✓ Podstawowa metoda z rodziny Agile
- ✓ Metoda jest zdefiniowana w Scrum Guide – zaledwie 17 stronicowym dokumencie

Scrum - definicja

Oficjalna definicja metody wg Scrum Guide

“Scrum (rzecz.): ramy postępowania (ang. framework), dzięki którym ludzie mogą z powodzeniem rozwiązywać złożone problemy adaptacyjne, by w sposób produktywny i kreatywny wytwarzać produkty o najwyższej możliwej wartości.

”.

Scrum w Pigułce

1. Ustal datę prezentacji
2. Stwórz produkt do zaprezentowania
3. Przedstaw go odbiorcy
4. Uzyskaj informację zwrotną i wykorzystaj ją w toku dalszego rozwoju
5. Powtarzaj ten cykl regularnie i często, jak najczęściej

W Scrumie właśnie chodzi o to:

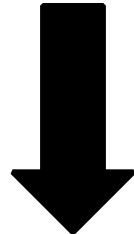
- ✓ Wszyscy docenią Scruma, bowiem opisuje on dokładnie to, co robimy, gdy zostaniemy przyparci do muru.

Jim Coplien The Scrum Guide, Ken Schwaber, Jeff Sutherland)

- ✓ Paradoksalnie, dopiero kiedy jest naprawdę tragicznie postępujemy właściwie:
 - ✓ zbieramy zespół
 - ✓ odwołując się do nadziednego celu prosimy o inicjatywę,
 - ✓ deklarując pełne wsparcie.

Scrum nie jest:

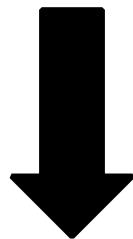
✓ Cudotwórcą, nie uzdrawia cudownie, nie leczy arogancji, ignorancji, lenistwa i wiary w cuda



✓ Scrum powoduje natomiast, że wszystkie te problemy stają się bardzo widoczne

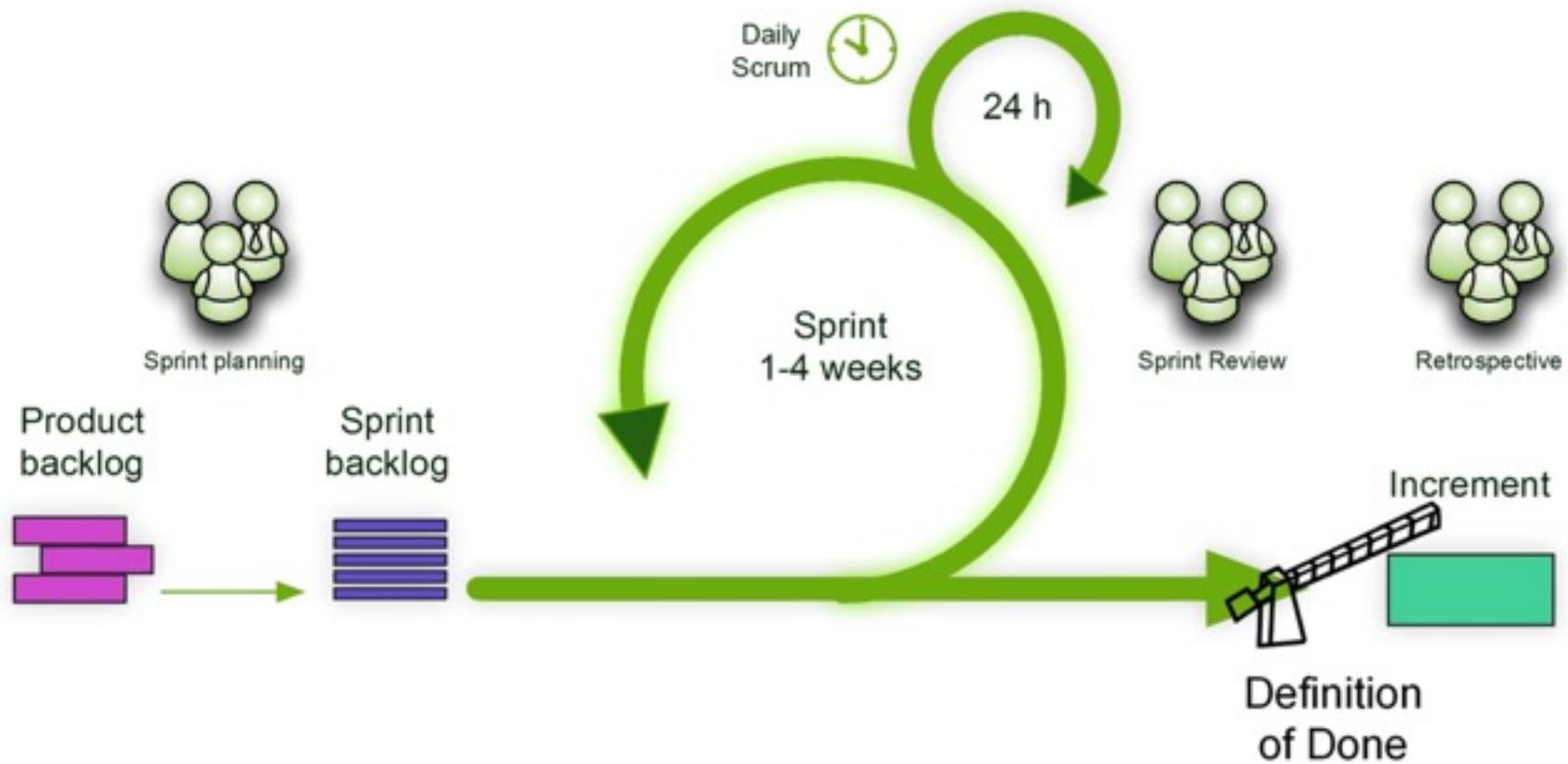
Scrum nie jest:

- ✓ Scrum bazuje na postulacie, że nie istnieje super-ekstra-hiper-rozwiązań dla produkcji oprogramowania



- ✓ Możliwe jest jedynie zdefiniowanie ram w obrębie których stosowane procesy i narzędzia będą empirycznie doskonalone

Tak to wygląda w Scrumie



3 Role, 5 Spotkań, 2 Listy

✓ 3 Role

- ✓ Właściciel Produktu (Product Owner)
- ✓ Zespół deweloperski (Development Team)
- ✓ Scrum Master

✓ 5 Spotkań

- ✓ Planowanie Sprintu (Sprint Planning)
- ✓ Codzienny Scrum (Daily Scrum)
- ✓ Przegląd Sprintu (Sprint Review)
- ✓ Retrospekcja Sprintu (Sprint Retrospective)
- ✓ Pielęgnacja Backlogu (Backlog Refinement)

✓ 2 Listy

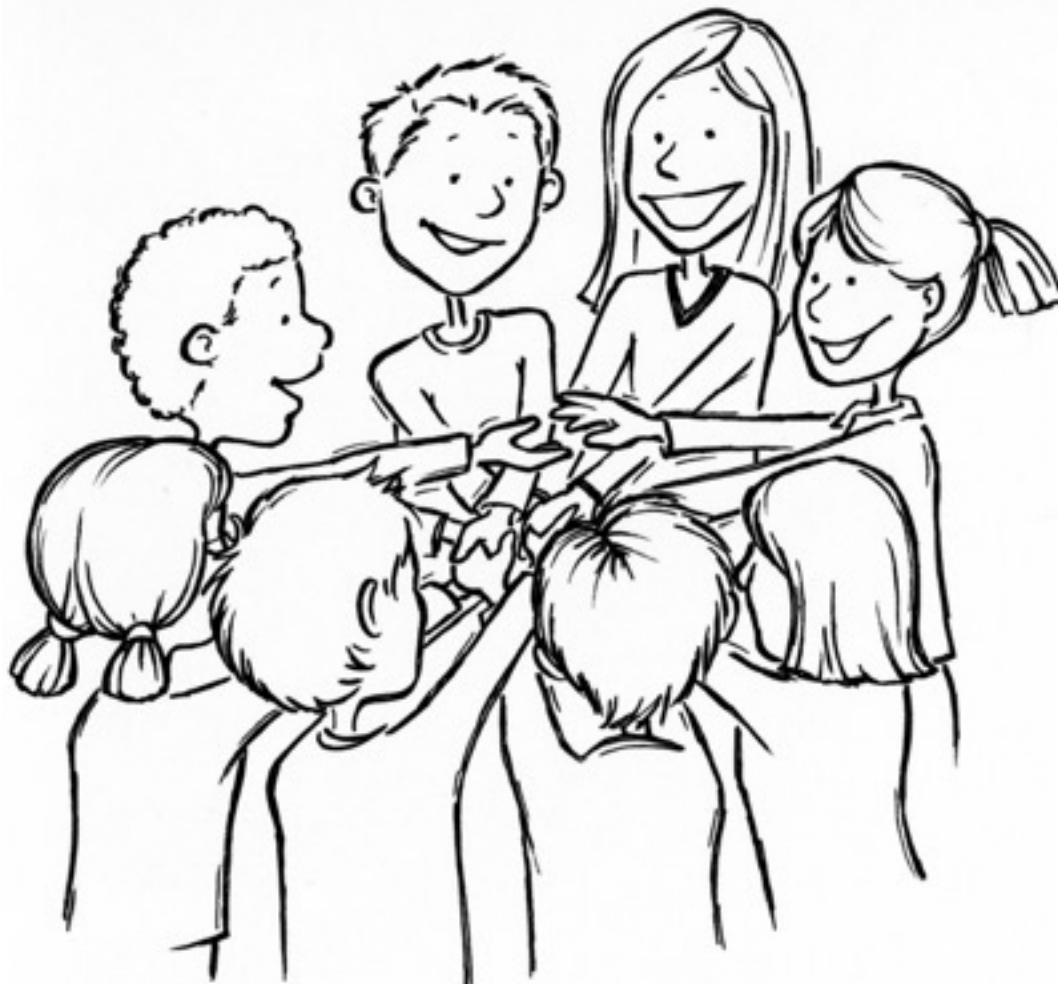
- ✓ Backlog Produktu (Produkt Backlog)
- ✓ Backlog Sprintu

Co to są Artefakty w Scrum

- ✓ Mamy 3 Artefakty w Scrum:
 - ✓ Backlog Produktu
 - ✓ Backlog Sprintu
 - ✓ Przyrost
- ✓ Artefakty Scruma reprezentują pracę lub wartość, aby uzyskać przejrzystość i okazję do dokonania inspekcji i adaptacji.
- ✓ Są one zaprojektowane w taki sposób, by zwiększać dostępność i czytelność kluczowych informacji tak, by wszyscy zainteresowani rozumieli dany artefakt w taki sam sposób

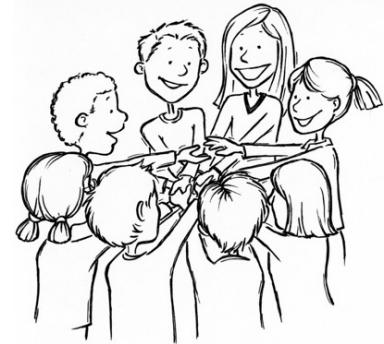
Role w
Scrum

Ludzki węzeł



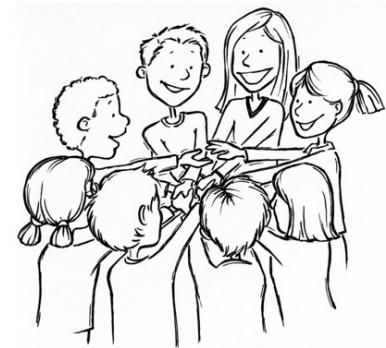
Ludzki węzeł

Scenariusz 1



- ✓ Dzielimy się na maksymalnie 10 osobowe zespoły
- ✓ Wybieramy jedną osobę, która będzie PM na zespół
- ✓ Każdy bierze wstążkę w swoją prawą rękę
- ✓ Lewą ręką łapiemy wstążkę, lecz innej osoby niż złapała Twoją wstążkę
- ✓ PM stara się rozwiązać zadanie (popiątaną grupę)
- ✓ Zespół może wykonywać tylko wskazówki PM, słuchamy jego rozkazów
- ✓ 5 minut na ukończenie zadania

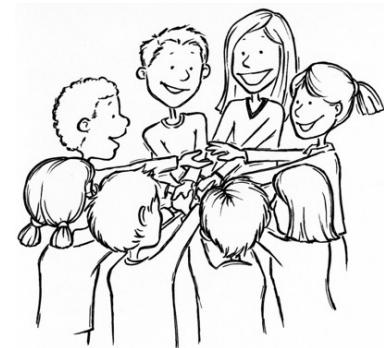
Ludzki węzeł



Scenariusz 2

- ✓ Dzielimy się na maksymalnie 10 osobowe zespoły
- ✓ Zespół nie posiada PM, jest „samoregulujący się”
- ✓ Każdy bierze wstążkę w swoją prawą rękę
- ✓ Lewą ręką łapiemy wstążkę, lecz innej osoby niż złapała Twoją wstążkę
- ✓ Zespół samemu stara się rozwiązać zadanie (poplątaną grupę)
- ✓ 5 minut na ukończenie zadania

Ludzki węzeł



Wnioski

- ✓ Jak czuliśmy się w scenariuszu 1, otrzymując polecenia ?
- ✓ Jak czuliśmy się w scenariuszu 2 mając pełną kontrolę?
- ✓ Jaką strategię przyjął zespół w scenariuszu 2?
- ✓ Jak wyglądała komunikacja w zespole w scenariuszu 1 oraz scenariuszu 2?

Role w Scrumie

Product Owner



Scrum Master



Development Team



Product Owner



Obowiązki Product Ownera

- ✓ Odpowiada za rejestr produktowy (ang. *product backlog*), czyli to, co jest do zrobienia.
- ✓ Dba o to, żeby praca zespołu miała wartość więc odpowiada za to, czy produkt będzie sensowny i dobrze pasujący do wymagań użytkowników/klientów, czy też nie.

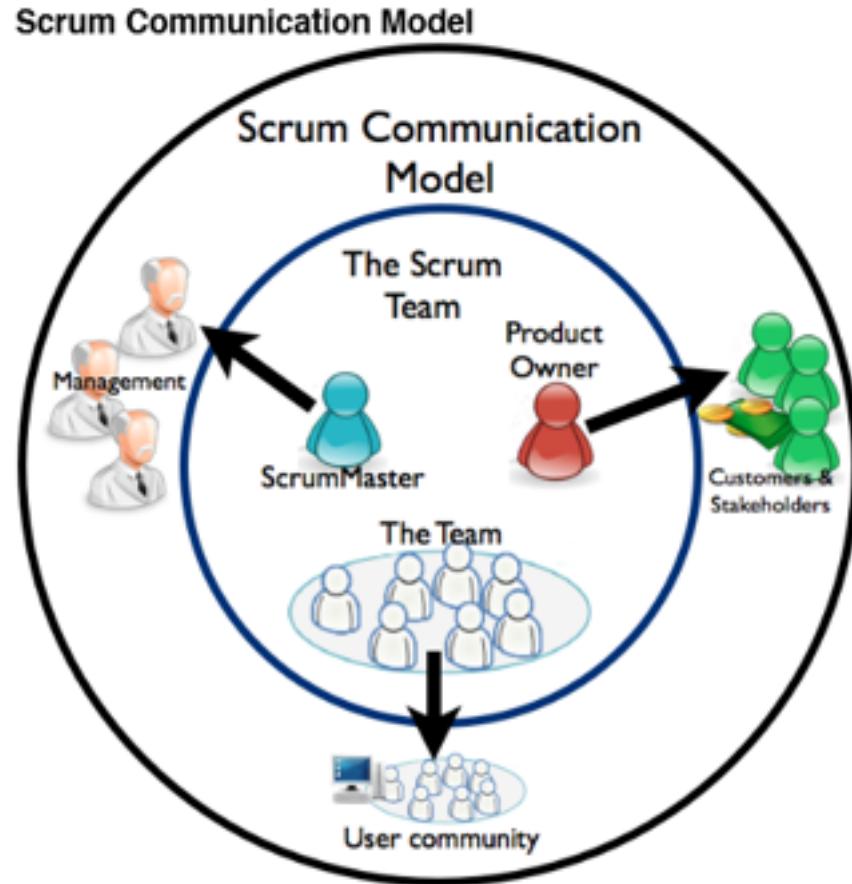
Dodatkowe obowiązki PO

Mogą być przez niego wykonywane następujące rzeczy

- ✓ Pisanie wymagań
- ✓ Negocjowanie z interesariuszami
- ✓ Projektowanie rozwiązań

Gdzie podziewa się PO?

- ✓ PO pracuje z zespołem,
- ✓ A nawet PO jest częścią zespołu



Relacje PO z otoczeniem



Idealny Product Owner

- ✓ Idealny PO to osoba, która wymyśliła dany produkt i z własnych pieniędzy opłaca jej rozwój
- ✓ Właściwie historia tej osoby nie jest tak ważna.
- ✓ Ważne, że posiada ona wiedzę i władzę nad rozwojem produktu.

Realny Product Owner

W rzeczywistości korporacyjnej działalność PO ma 2 wymiary:

- ✓ Produktowy (ustalenie jakie wymagania przyniosą w danej chwili wartość)
- ✓ Polityczny (negocjowanie z interesariuszami)

Proxy Product Owner

- ✓ Ludzie posiadający wiedzę i władzę zazwyczaj nie mają czasu
- ✓ Jest to osoba posiadająca odpowiednią wiedzę, ale jedynie część władzy

Scrum Master jako Product Owner

- ✓ Pomiędzy PO a Scrum Masterem występuje naturalny konfliktem interesów, który jednak jest twórczy, nie szkodzi żadnej ze stron.
- ✓ Właściciel Produktu chce ciągle więcej funkcjonalności
- ✓ Scrum Master czuwa nad procesem i broni zespołu, który nie może przyjąć na siebie więcej, bo nie pozwala mu na to tempo jego pracy.
- ✓ Doświadczenia z projektów IT wskazują, że nie jest możliwe sprawne pogodzenie tych 2 ról

Scrum Master



Keep the wolves of scope creep at bay.

Call in a certified ScrumMaster
to guard your team's success.

www.scrumalliance.org

 **Scrum**
Alliance

info Share
<academy/>

**Scrum Master odpowiada za to,
by Scrum był stosowany i rozumiany**

- ✓ Uczy, pokazuje sens, wspomaga w rozwoju - role edukacyjna
- ✓ Zależy mu na tym, by Scrum był sensownie realizowany i przynosił korzyści - rola procesowa

Scrum Master

- ✓ Scrum Master - osoba **odpowiedzialna za proces.**
- ✓ **Scrum** Master ma dbać o to, by Scrum był prawidłowo zrozumiany i stosowany zarówno w samym zespole jak i w organizacji.
- ✓ Scrum Master odpowiada także za zespół, jego produktywność i zaangażowanie.
- ✓ Dba o zespół pomagając mu w rozwiązywaniu problemów i usuwaniu przeszkód (w metodyce nazywanych impediments) oraz wspomagając jego samoorganizację.

Scrum Master

- ✓ Nie jest jednak kierownikiem zespołu ani liderem technicznym.
- ✓ Nie może bezpośrednio polecać członkom zespołu wykonania takiej czy innej pracy, karać ich czy nagradzać ani określać jaką rolę będą pełnić w zespole.

Scrum Master – problem władzy

- ✓ Scrum Master ma zupełnie innego rodzaju obowiązki w porównaniu z tradycyjną rolą kierownika projektu (odpowiadającego za zakres, budżet i harmonogram)
- ✓ Inaczej też „używa” władzy: nie ma władzy nad zespołem (w sytuacji, gdy ktoś zawala robotę, nie może tego kogoś zwolnić lub powiedzieć mu, co ma robić, żeby uniknąć podobnych wpadek w kolejnych sprintach); ma jednak władzę nad procesem.

Scrum Master - ma władzę na procesem



Moim zadaniem jest czuwanie nad tym, żeby członkowie zespołu konsekwentnie realizowali reguły i zasady Scruma w projekcie.

Scrum Master wspiera Product Ownera

- ✓ Pomaga (ale nie wyręcza) w zarządzaniu Product Backlogiem
- ✓ Pomaga zrozumieć podejście empiryczne w procesie tworzenia produktów
- ✓ Pomaga zrozumieć proces Scruma Product Ownerowi

Scrum Master wspiera Zespół Deweloperski

- ✓ Usuwa wszelkie przeszkody Zespołu Deweloperskiego
 - ✓ np. chroniczna niedostępność środowiska testowego
 - ✓ lub brak dostępnych urządzeń mobilnych w zespole
- ✓ Wspiera samoorganizację (ang. self-organization),
- ✓ Pomaga zrozumieć między-funkcjonalność zespołu (ang. cross-functionality),
- ✓ Moderuje spotkania, jeśli zachodzi taka potrzeba,
- ✓ Pomaga zrozumieć Scruma oraz zasady i wartości zwanego wytwarzania oprogramowania

Scrum Master wspiera organizację

- ✓ Pomaga zrozumieć Scruma pracownikom organizacji
- ✓ Przeprowadza procesy wdrażania Scruma
- ✓ Inicjuje działania zwiększające produktywność Zespołu Scrumowego

Spojrzenie z zewnątrz. W czym tkwi jego siła?

- ✓ Scrum Master często dba, by zespół dotrzymywał tych zobowiązań, które sam przed sobą podjął
 - ✓ Zobowiązań ze Sprintu
 - ✓ Zobowiązań z Retrospektywy
- ✓ Jak robimy dowolną rzecz, to szybko się angażujemy w tą rzecz i potrzebne jest spojrzenie z boku, by zastanowić się nad sensem

Czy łączenie roli Scrum Mastera i dewelopera jest korzystne?

- ✓ Jest to dopuszczalne i czasami działa, lecz nie zaleca się by to wykonywać
- ✓ Często firmy sięgają po Lead Dewelopera, by był Scrum Masterem, bo jest taki dobry prosiąc, by 80% czasu poświęcił na programowanie, 20% na bycie Scrum Masterem
 - ✓ Mamy wtedy przykład mechanicznego Scruma. SM wtedy tylko dba o tablice, organizuje spotkania, idzie według listy i nie ma na nic więcej czasu.
 - ✓ Ma mniej czasu na bycie Leadem, czyli uczenie innych i dowożenie w terminie
- ✓ Praca Scrum Mastera to praca z ludźmi, grupą, jej rozwojem, środowiskiem nie praca z kodem. Zdarzają się ludzie, którzy potrafią to łączyć, lecz nie ma ich wielu

Dyrektywny Scrum Master. Jak to wpływa na zespół?

- ✓ Scrum Master działający jak kierownik, który mówi ludziom co mają robić, stosujący mikrozarządzania, heroicznie dowożący Sprinty.
- ✓ To nie jest samoorganizujący się zespół, więc to NIE JEST SCRUM!
- ✓ To jest tradycyjny kierownik, tylko z inną naklejką!

Co oznacza, że Scrum Master jest przywódcą służebnym (ang. servant leadership)?

- ✓ To nie jest przyjmowanie postawy służalczej. Chodzi o wczesne identyfikowanie potrzeb otoczenia (zespołu, organizacji, klienta) i udzielanie im niezbędnego wsparcia
- ✓ Przywódca służebny słucha, wykazuje się empatią, jest otwarty, chętnie pomaga i wspiera ludzi w rozwoju.
- ✓ Osoby wychodzące z pozycji siły nie będą dobrymi Scrum Masterami.

Czy jak Scrum Master usunie przeszkody, to jest potrzebny?

- ✓ Misją Scrum Mastera jest tak nauczyć zespół, by SM był jak najmniej potrzebny
- ✓ Często doświadczeni Scrum Masterzy zajmują się 2-3 projektami, choć mówi się, że dobry SM wspomaga 2-3 projekty, a bardzo dobry 1 projekt
- ✓ Jedyne co pewne co się wydarzy to “Zmiana”. Nie ma idealnie naoliwionej maszyny i zawsze jest potrzeba posiadania Scrum Mastera
- ✓ Scrum Master nie działa w próżni, tylko w organizacji, współpracując z klientem. SM pracuje wtedy nad organizacją, klientem
- ✓ Porównuje się Scrum Mastera do administratora systemowego.

Proza życia Scrum Mastera

Zachęca

- ✓ Do komunikacji “twarzą w twarz”, a nie “zakopywaniu problemów”
- ✓ Do samoorganizacji się zespołu i podejmowaniu zobowiązań
- ✓ Do przejrzystości i szczerości na spotkaniach Daily Standup, Review, Retrospektyna
- ✓ Do otwartości na zmianę
- ✓ Do naprawiania problemów bez szukania kozłów ofiarnych

Proza życia Scrum Mastera

Zachęca do refleksji

- ✓ Pomaga zespołowi w przygotowaniu “Information Radiators” np. Burndown Charts
- ✓ Pomaga zespołowi w Retrospektywie i w ciągłym ulepszaniu procesu wytwarzania
- ✓ Pomaga zespołowi w utrzymaniu Narzędzi Scrumowych np. Scrum, Kanban Board
- ✓ Zbiera i przynosi obserwacje o zespole z otoczenia zespołu

Proza życia Scrum Mastera

Nagradza

- ✓ Potrafi docenić dobrze działający zespół
- ✓ Świętuje razem z zespołem, by podkreślić ważne momenty

Proza życia Scrum Mastera

Uczy i poszerza wiedzę

- ✓ Ciągle poszerza swoją wiedzę we wszystkim związanym z Agile
- ✓ Uczy i konsultuje z zespołem we wszystkim związanym z Agile
 - ✓ Jest Trenerem,
 - ✓ Mentorem,
 - ✓ Coachem w sprawach związanych z Agilem
- ✓ Konsultuje się i pomaga innym Scrum Masterom
- ✓ Daje regularny “feedback” zespołowi
- ✓ Pomaga zespołowi mierzyć, to co ważne, by móc na tej podstawie wyciągać wnioski

Proza życia Scrum Mastera

Facylituje

- ✓ Może prowadzić spotkania scrumowe
- ✓ Może prowadzić spotkania planistyczne zespołu np. Story Mapping, Release Planning
- ✓ Tworzenie zrozumiałego i osiągalnego “Definition of Done”
- ✓ Osiąganiu prędkości pracy, dzięki której dobrze i wydajnie się pracuje (ang. sustainable pace)
- ✓ Inne aktywności w zespole (pair programming, wspólne lunche, wyjścia zespołowe)

Proza życia Scrum Mastera

Chroni

- ✓ Zespół przed popadaniem w konflikty
- ✓ Przed różnymi przeszkodami, pomagając zespołowi je usunąć
- ✓ Przed naciskami z zewnątrz

Czy Project Manager jest potrzebny w Scrumie?

- ✓ Tak, lecz nie stosujący mikro-zarządzania
- ✓ Tak, lecz nie stosujący podejścia rozkaz i kontrola
- ✓ Tak, lecz wspierający samoorganizację
- ✓ PM może pełnić rolę Scrum Mastera lub Product Ownera (pod warunkiem, że potrafi dostosować się do nowej metodyki)

Co robi PM w Scrumie?

- ✓ Odpowiada za negocjacje kontraktowe
- ✓ Odpowiada za budżet i rentowność projektu po stronie Wykonawcy (określenie kosztów realizacji)
- ✓ Określa ilości zasobów potrzebnych do realizacji projekt
- ✓ Odpowiada za zbudowanie odpowiednich zespołów projektowych

Co robi PM w Scrumie?

- ✓ Zajmuje się raportowaniem projektu (raporty w Scrumie nie są przewidywane, lecz może być takim połączeniem np. gdy Wykonawca wykonuje Scrum a Zlecający ma wdrożonego Prince lub metodykę opartą na PMBOK)
- ✓ Współpracuje z innymi działami (Księgowość, HR)
- ✓ Zarządza polityką wynagrodzenie członków zespołu projektowego
- ✓ Raportuje przebieg projektu wyższemu kierownictwu
- ✓ Pomaga Scrum Masterowi w usuwaniu przeszkód projektowych

Zespół



Zespół

- ✓ Zespół jest w absolutnym centrum metody Scrum



Zespół

- ✓ Zwinne zespoły czują się współwłaścicielami produktu.



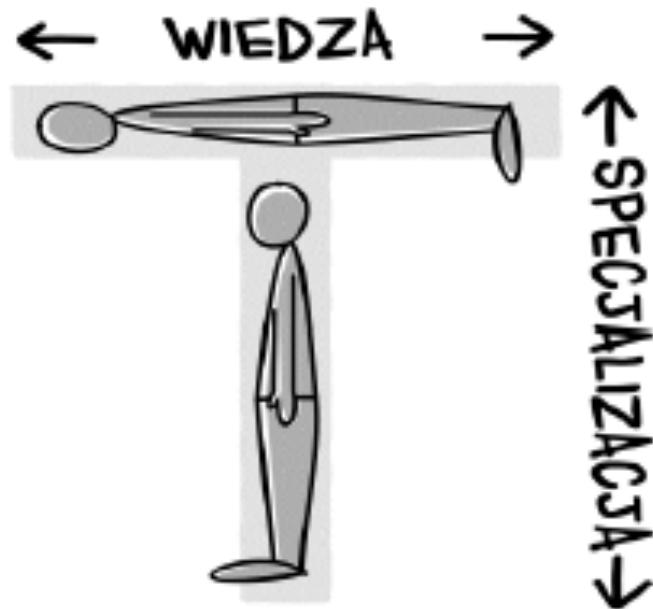
Zespół

- ✓ Nie wszyscy pasują do projektów Scrumowych
- ✓ Osoby, które są samotnikami, nie stawiają na współpracę nie znajdują się w tej metodzie
- ✓ Podobnie jak ścisli specjaliści, którzy nie mają zamiaru wychodzić poza „świat” swojej specjalizacji



Zespół

- ✓ Zespoły Scrumowe tworzą ludzie w kształcie litery „T”
- ✓ Z jednej strony są dobrymi specjalistami w swojej dziedzinie, z drugiej zaś mają na tyle wysoki poziom empatii, że potrafi zrozumieć, co robią inni

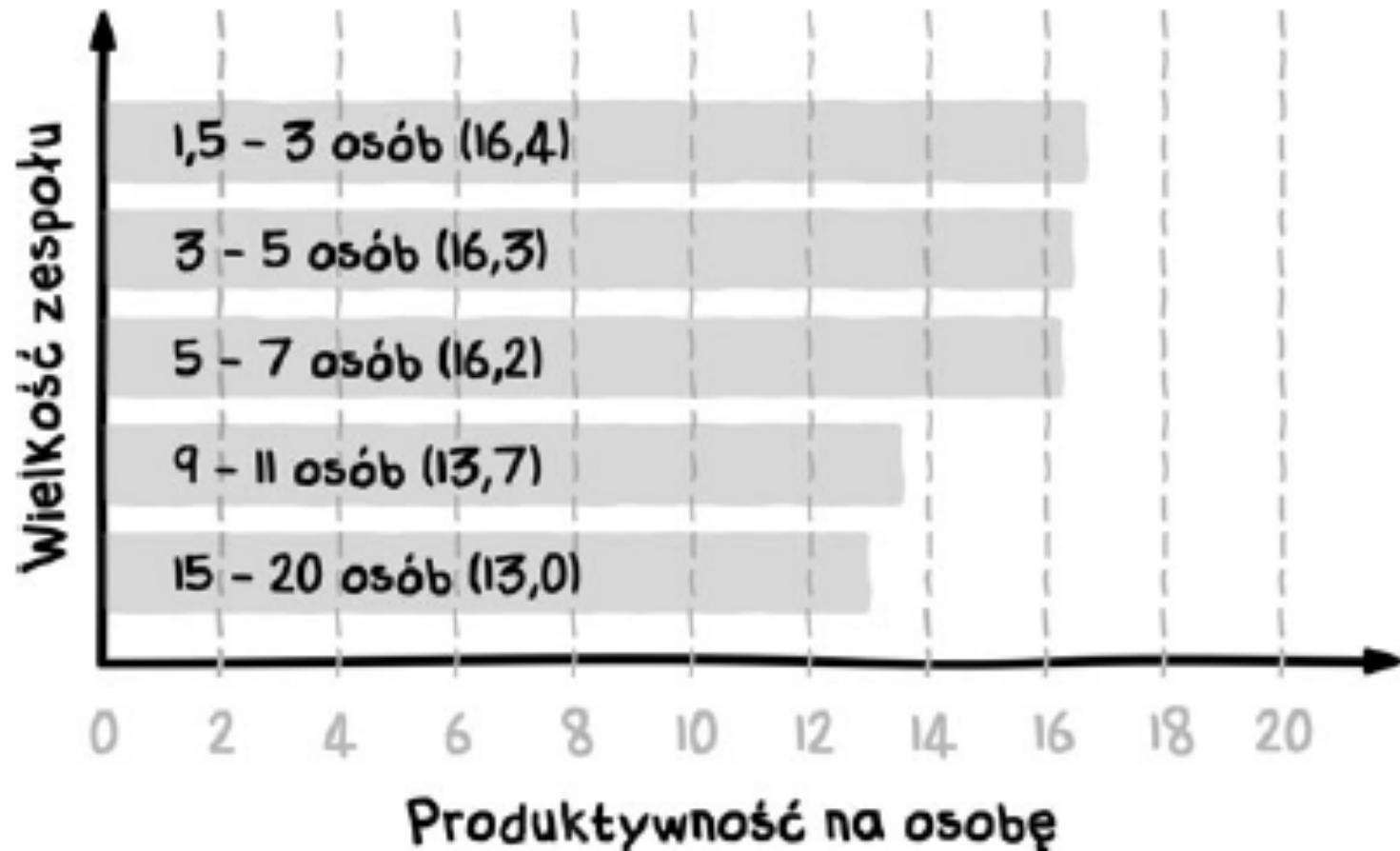


Wielkość zespołu

- ✓ Scrum mówi, że optymalna wielkość zespołu to od trzech do dziewięciu osób

3 - 9

Wielkość zespołu



Product Backlog

Backlog Produktu - definicja

- ✓ Backlog produktu - zbiór wszystkich rzeczy do zrobienia
- ✓ Zawiera PBI (Produkt Backlog Items), które w metodyce SCRUM zapisane są m.in. za pomocą historyjek
- ✓ Uporządkowany na podstawie wartości, zależności i ryzyka
- ✓ Zarządzany przez Product Ownera

Historyjka użytkownika (user story)

Zespoły agile preferują takie techniki jak historyjki użytkownika (ang. user stories)

Historyjki użytkownika to wyrażenie oczekiwanej zachowania systemu wedle następującego formatu:

- ✓ “**jako <rola>** mogę wykonać <czynność>
aby osiągnąć <rezultat>”

- ✓ przykład: jako użytkownik mogę utworzyć profil z loginem i hasłem
aby móc korzystać z usług systemu).

Historyjki użytkownika (user story)

Są **istniejące samodzielnie** (ang. Independent),

- ✓ Dobrze napisane historyjki nie zależą od innych.
- ✓ Historyjki, które zależą od siebie, również niepotrzebnie komplikują proces szacowania.

Historyjki użytkownika (user story)

Są **negocjalne** (ang. negotiable),

- ✓ Nie należy ich traktować jak spisanej umowy zawartej pomiędzy Właścicielem Produktu (zleceniodawcą) a zespołem (zleceniobiorcą) — wykonawcą,
- ✓ Zawierają tylko krótki opis funkcjonalności zapisany na małej kartce papieru. Na więcej nie ma miejsca. Reszta — wszystkie szczegóły i drobnostki — dopracowuje się w rozmowie

Historyjki użytkownika (user story)

Są **wartościowe** (ang. valuable)

- ✓ Najlepsze rozwiązanie to historyjki użytkownika pisane przez Właściciela Produktu, z jego perspektywy, gdyż myśli on o wartości historyjki dla projektu.

Historyjki użytkownika (user story)

Są **małe** (ang. small),

- ✓ Każdy zespół wypracowuje definicje we własnym zakresie.
- ✓ Przyjmuje się, że mała historyjka to taka, jeżeli jesteśmy w stanie zaimplementować ją w jednym sprincie.

Historyjki użytkownika (user story)

Są **estymowalne** (ang. estimable),

- ✓ Historyjka jest dobrze napisana, jeżeli jesteśmy w stanie oszacować jej wielkość.
- ✓ historyjka musi być do tego stopnia czytelna, żeby na jej podstawie, Właściciel Produktu mógł ją jakoś umiejscowić względem pozostałych, a także z grubsza zaplanować prace.

Historyjki użytkownika (user story)

Są **testowalne** (ang. testable).

- ✓ Testy akceptacyjne są bardzo ważnym elementem każdej historyjki użytkownika.
- ✓ nie można mówić o tym, że napisaliśmy dobrą historyjkę, gdy nie da się jej przetestować.
- ✓ Testy akceptacyjne zapisuje się jako DoD (Definition of Done)

Backlog Produktu

✓ jest uporządkowany (ang. ordered) -
wymagania są ułożone w kolejności realizacji,

Backlog Produktu

✓ jest przejrzysty (ang. transparent) - backlog jest dla zainteresowanych widoczny i czytelny - rozumieją wymagania w nim zapisane,

Backlog Produktu

✓ jest **adekwatny** (ang. relevant) - zawiera
wymagania odpowiadające aktualnym
potrzebom biznesowym,

Backlog Produktu

✓ jest wypielęgnowany (ang. refined) - stopień szczegółowości wymagań jest dostosowany do ich pozycji na backlogu.

Backlog Produktu

- ✓ Backlog może być również **oszacowany** (ang. estimated) jeśli jest to wymagane i potrzebne do podejmowania dobrych decyzji co do kolejności wymagań na nim oraz planowania długofalowego.
- ✓ User Stories mogą być oszacowane za pomocą roboczogodzin lub „punktów” – (ang. story points)

Epik

- ✓ Historyjki, które są duże, nazywa epikami (ang. epics.)
- ✓ Stosuje się zasadę, że rozmiar historyjki nie powinien przekraczać połowy wartości prędkości zespołu, który będzie ją rozwijał. Wszystko, co będzie większe, może być potraktowane jako epik

Spike - dedykowany czas na eksplorację.

- ✓ Projekt naukowo-badawczy mający na celu oszacowanie wymagania
- ✓ Stosuje się kiedy zespół co prawda dobrze rozumie samo wymaganie (rozumie jaką wartość biznesową ma dostarczyć lub jakie jest oczekiwane zachowanie systemu), ale nie jest pewien jak osiągnąć zadowalającą implementację.

Pielegnacja Backlogu

Pielęgnacja Backlogu

- ✓ Wypielęgnowany backlog - backlog produktu, w którym wymagania są odpowiedniej wielkości do swojej na nim pozycji (małe na początku, duże na dole),
- ✓ wymagania na około 2-3 sprinty do przodu są dobrze zrozumiane i gotowe do zaplanowania pracy nad nimi.

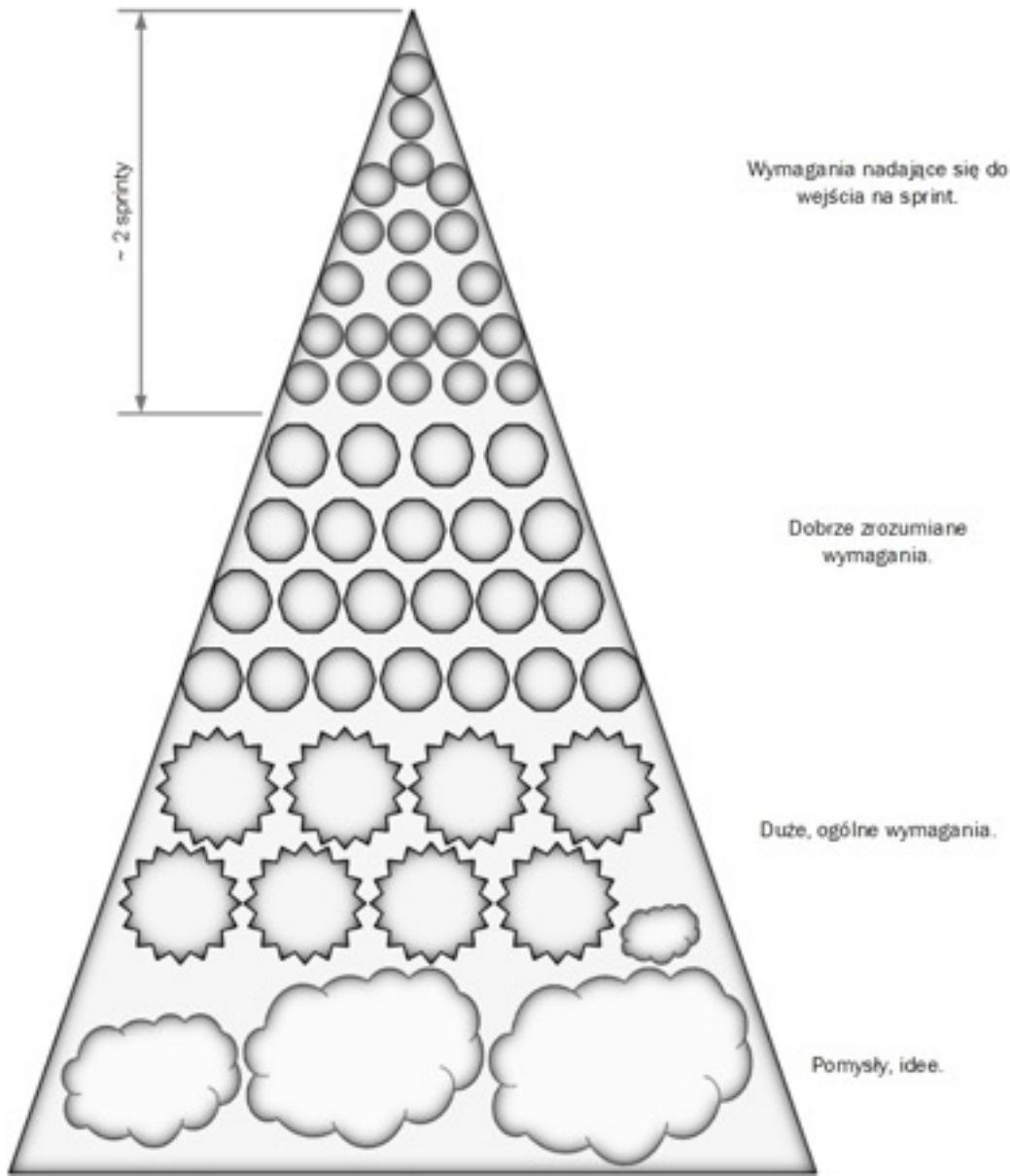
Backlog Grooming

Kiedy zajmujemy się
Pielęgnacją Backlogu w
Scrumie?

Backlog Refinement

- ✓ Kiedy – przed Sprintem (podczas poprzedniego Sprintu)
- ✓ Ile - do 10% czasu zespołu
- ✓ Kto – Product Owner, Scrum Master, Zespół, Każdy kto posiada wiedzę w danym temacie

Backlog Grooming



Podział Backlogu na DoR i Icebox

Pivotal Tracker

My Sample Project

CURRENT BACKLOG ICEBOX EPICS MORE PROJECT STORIES

CURRENT

- 8 | 3 Mar - Current Pts: 0 of 10
 - search, shopping Shopper should be able to search for product (rosz) Finish
 - shopping Some product photos not scaled properly when browsing products Finish
 - search configure solr for full text searching (rosz) Finish
 - shopping nie dziala logowanie Start
 - shopping Product browsing pagination not working in IE6 Start
 - checkout, shopping Shopper should be able to enter credit card information and shipping address Start
 - Initial demo to investors Finish
 - checkout, needs discussion, shopping When shopper submits order, authorize total product amount from payment gateway Start

BACKLOG

- 9 | 10 Mar Pts: 19 | 200%
 - checkout, ie6, shopping If authorization is successful, show order number and confirmation message to shopper Start
 - admin, checkout, shopping Send notification email of order placement to admin Start
 - orders Shopper should be able to check status of order by entering name and order number Start
 - orders Shopper should be able to ask question about order Start
 - admin, orders Admin can review all order questions and send responses to shoppers Start
 - deployment Set up Engine Yard production environment Start
 - Beta launch Finish
 - signup / signin Shopper Start

ICEBOX

- Integrate with automated order fulfillment system
- epic native iPhone app to allow product browsing and checkout
- Facebook app, allowing users to share favorite products

Backlog w Jira

Backlog

Board ▾



QUICK FILTERS: Only My Issues Recently Updated

VERSIONS

EPICS

All issues

Frontend setup

▼ Sprint 9 21 issues

5 8 0

26/Jan/16 10:35 AM • 02/Feb/16 10:35 AM



...

TP-316 Incorrect position of calendar (when opened directly on the list) Agenda

TP-317 Delete confirmation pop-up doesn't close automatically when new event occurs Agenda

TP-285 Refactor all set assignee dropdowns

TP-310 API: grant admin permissions to organization member Organizations

TP-311 API: revoke admin permissions to organization member Organizations

TP-294 Make breadcrumbs clickable Agenda

TP-51 Manage to-do comments on details view Agenda 3

TP-94 Display details of a to-do in separate view ("full screen" mode) Agenda 2

TP-181 Prompt warning when leaving the form without saving the changes Agenda

TP-314 Completed items don't have label with time of completion Agenda

TP-44 Display completed to-dos Agenda 3

TP-245 Create automated test cases for to-do drag&drop mechanism Testing tasks

Backlog Sprintu w Jira

Sprint 9

⌚ 1 day remaining Complete Sprint

Board ▾



QUICK FILTERS: Only My Issues Recently Updated

To Do

In Progress

Ready For Testing

Done

> TP-51 DONE = AC AND DESL... 8 sub-tasks Manage to-do comments on details view

▼ TP-94 DONE = AC AND DESL... 1 sub-task Display details of a to-do in separate view ("full screen" mode)

Move to Done

TP-286

↑ Design for to-do reference in "full screen" mode

▼ TP-44 IN PROGRESS 5 sub-tasks Display completed to-dos

TP-340

↑ Make close button following screen while scrolling

TP-341

↑ Lazy load completed todos

TP-345

↑ Allow to pass "offset" as request parameter to fetch items from exact position

TP-45

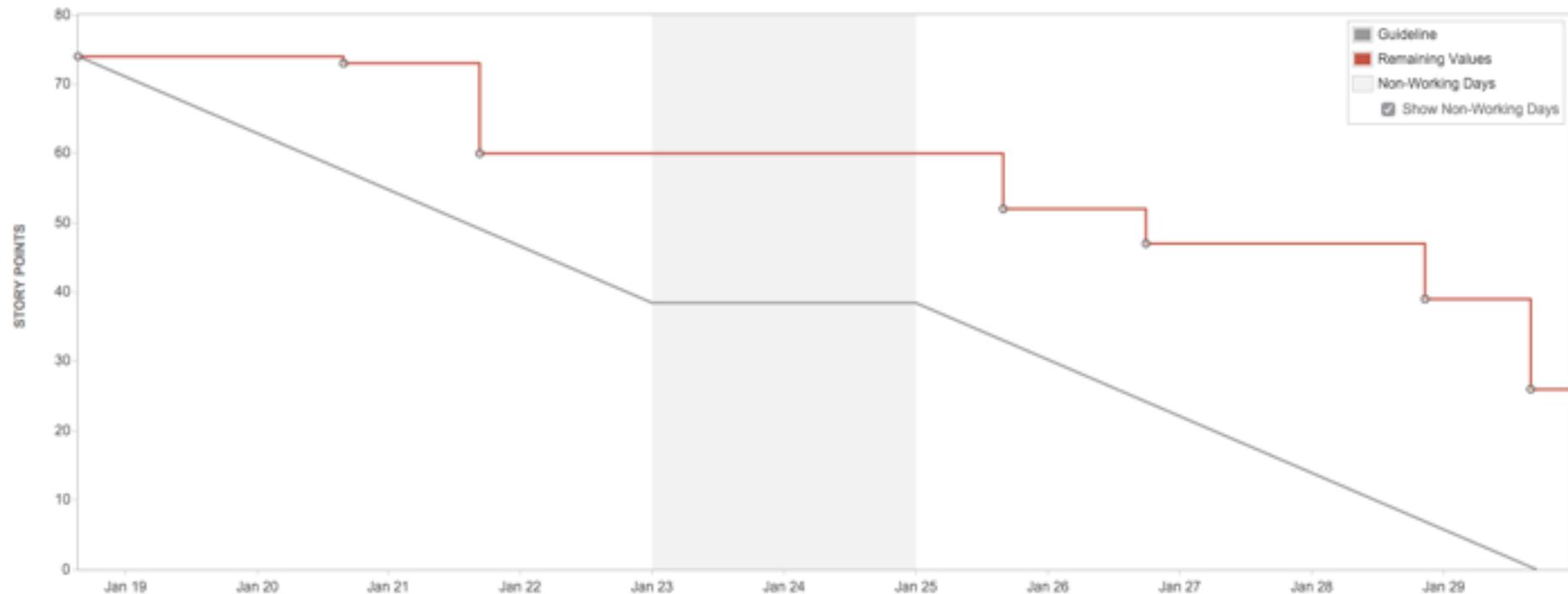
↑ Displaying completed to-dos?

TP-89

↑ Time intervals

Burndown chart w Jira

Burndown Chart [Switch report](#) ▾



Estymacja

Estymacja

Po co szacujemy?

Tradycyjne jednostki miary

Typowe, często spotykane miary wielkości wymagań to jednostki związane z czasem takie jak jak:

- ✓ roboczogodzina
- ✓ dzień roboczy
- ✓ miesiąc roboczy

Wady tradycyjnych jednostek miary

- ✓ Godzina pracy jednej osoby nie jest równa godzinie pracy innej osoby – każdy ma inne doświadczenie, umiejętności itp.
- ✓ Różne osoby będą szacować to samo wymaganie inaczej, adekwatnie do swoich kompetencji

Wady tradycyjnych jednostek miary

- ✓ Klienci zupełnie nie rozumieją, że oszacowanie podane w „dniach roboczych” nie przekłada się prosto na konkretne dni kalendarzowe.
- ✓ Jednostki czasu powodują zobowiązania wobec klientów, co zaburza zarówno oszacowanie jak i relację z klientami.

Story Points = SP

- ✓ Story Point to **bezwymiarowa jednostka relatywnej wielkości.**
- ✓ Szacowanie z użyciem tej jednostki polega zatem na rozważaniu jak mają się wymagania wzajemnie do siebie.
- ✓ Przykład: jedno wymaganie zostało oszacowane na **2 SP** a drugie na **4 SP** to jedyna informacja, jaką te liczby przekazują jest taka: **wymaganie drugie jest około dwa razy większe (bardziej złożone) niż pierwsze.**

Szacowanie w punktach dla nowego backlogu

- ✓ wybrać na backlogu wymaganie o średniej wielkości i przydzielić mu oszacowanie np. 5, następnie pozostałe wymagania szacować w odniesieniu do tego pierwszego,
- ✓ wybrać na backlogu dwa wymagania – małe, ale nie najmniejsze oraz duże, ale nie największe, następnie nadać im odpowiednio oszacowania 3 i 13 , następnie pozostałe wymagania szacować w odniesieniu do tych dwóch.

Zalety Story Points

- ✓ Pomimo różnic w umiejętnościach nowicjusza” vs. „wyjadacza”) obaj są w stanie się zgodzić co do tego, że jedno wymaganie jest np. trzy razy bardziej złożone niż drugie.
- ✓ Dzięki temu łatwiejsze staje się grupowe szacowanie wymagań.

Zalety Story Points

- ✓ Podczas szacowania z użyciem SP koncentrujemy się na samych wymaganiach i ich złożoności.
- ✓ Szacowanie nie jest zatem obciążone rozważaniem czasu (oraz przystawaniem tego czasu do jakichś terminów)

Ograniczenia Story Points

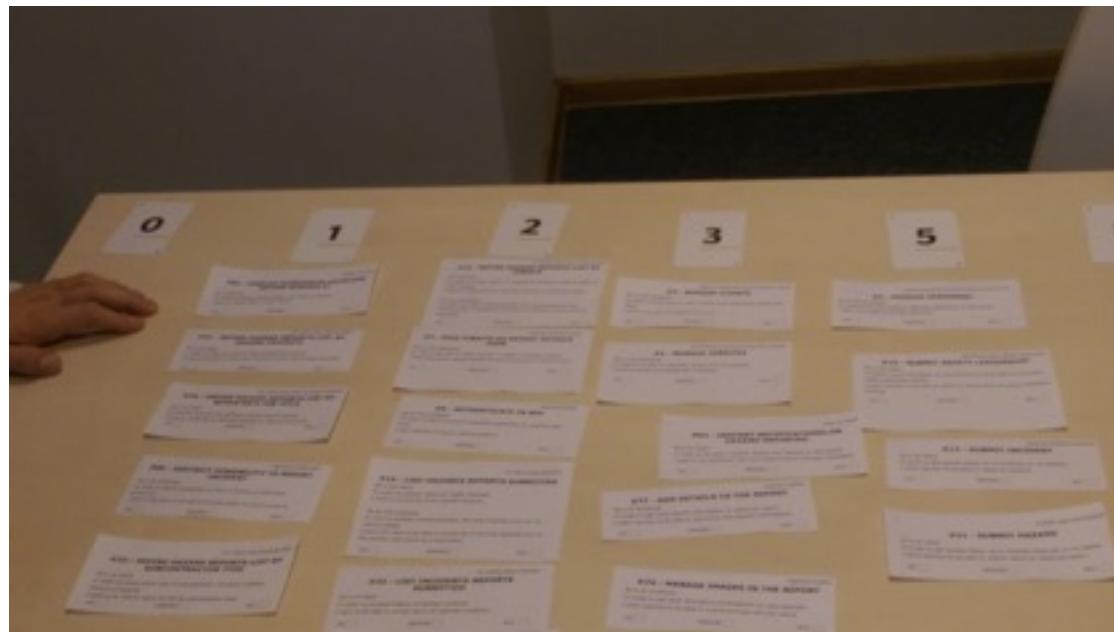
- ✓ Są one nieporównywalne pomiędzy zespołami,
- ✓ Nie da się ich odnieść wprost na sztywno do czasu lub kosztu.

Planning Poker



Affinity Mapping

1. Wymagania przygotowuje się w postaci umożliwiającej ich łatwe sortowanie - jedno wymaganie na kartce
2. Wymagania umieszcza się na stole lub na podłodze. Jednocześnie umieszcza się na nim kartki oznaczające możliwe oszacowania – najczęściej stosuje się ciąg Fibonacciego



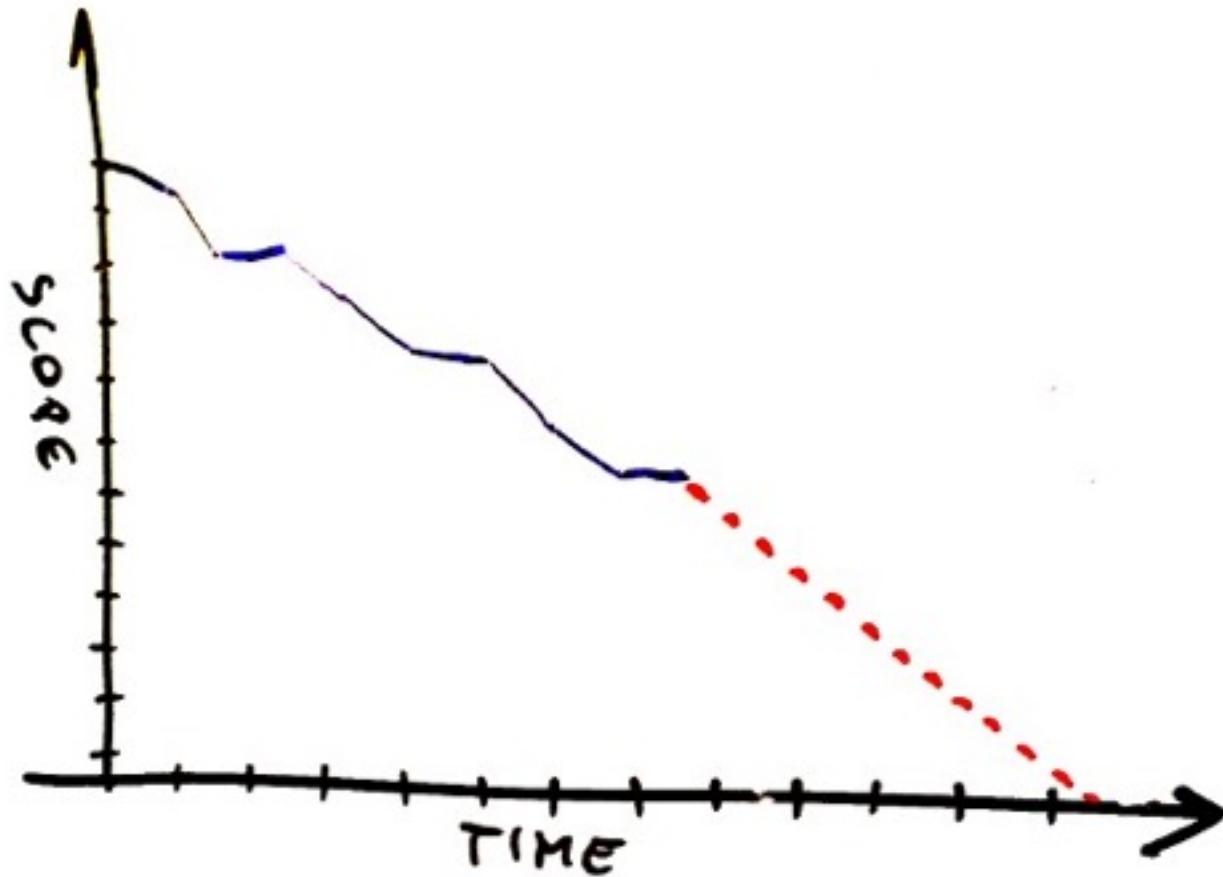
Affinity Mapping

3. Zadaniem grupy jest pogrupowanie karteczek z wymaganiami koło odpowiednich dla nich oszacowań. Jeśli występuje różnica zdań grupa rozstrzyga ją w dyskusji, a w razie konieczności poprzez głosowanie.

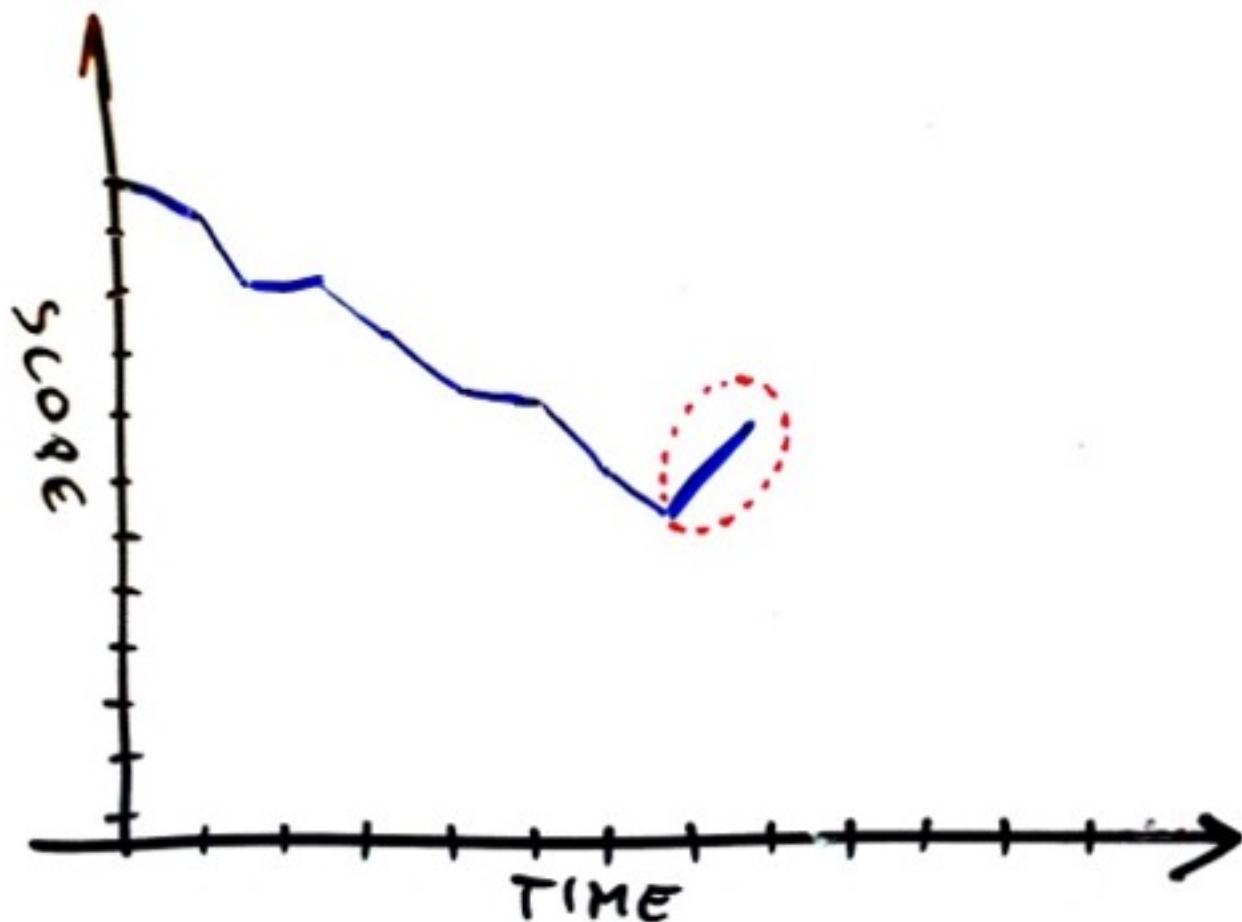


Planowanie wydania

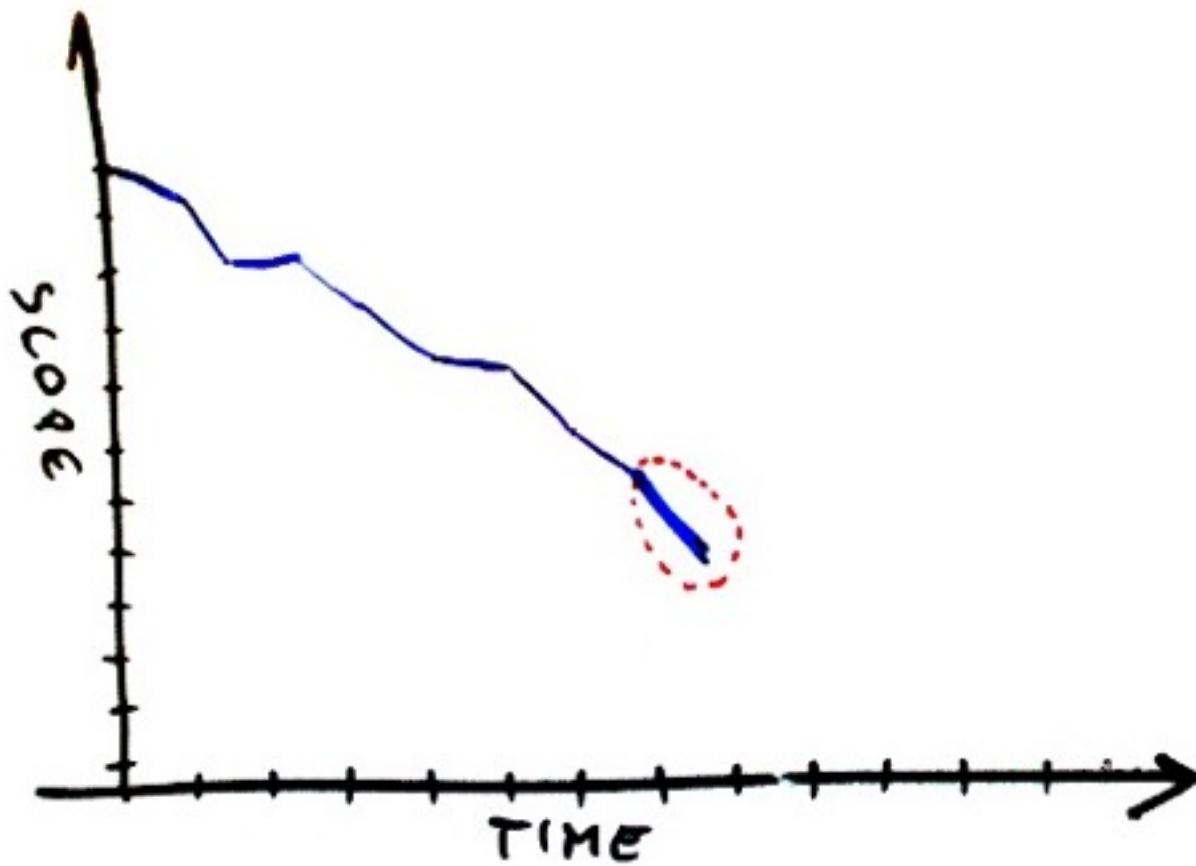
BurnDown Chart



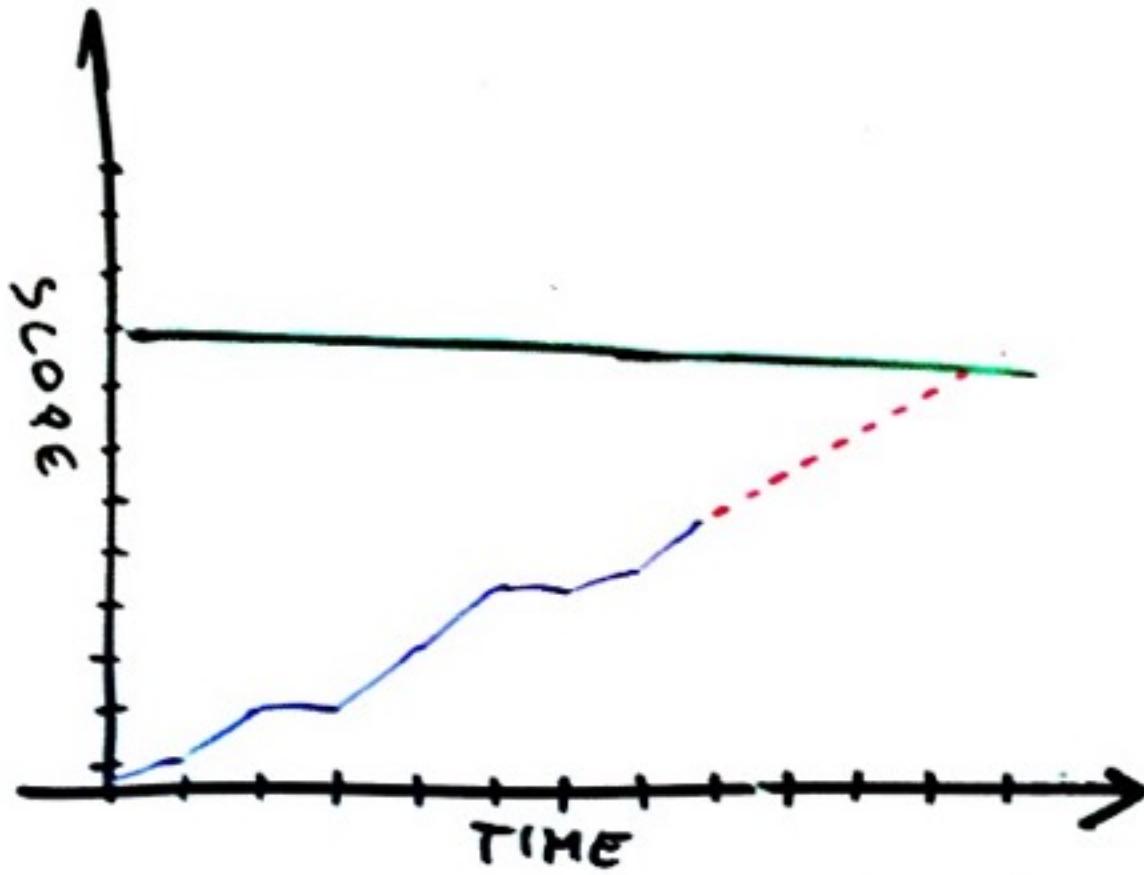
BurnDown Chart



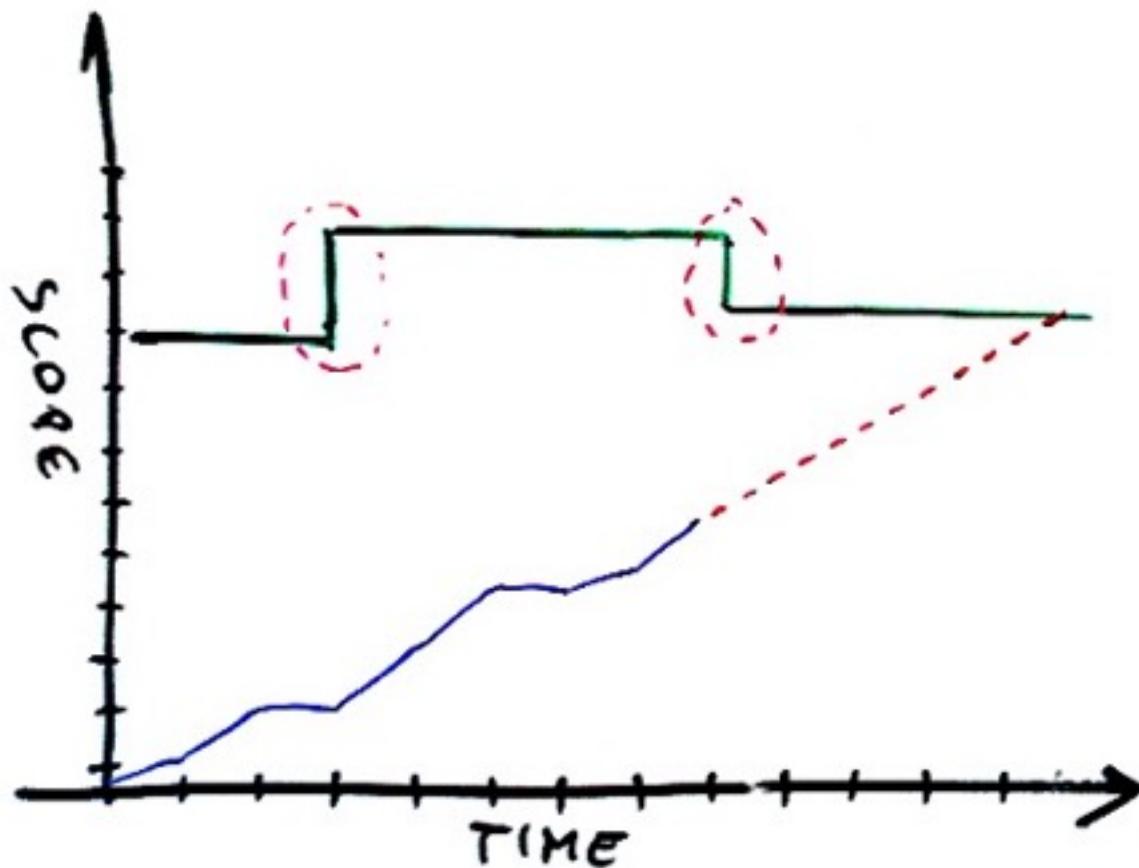
BurnDown Chart



Burnup



Burnup



Product Canvas

 Name	The name of the product.	 Goal	The reason for creating the product.	 Metrics	The measures to determine if the goal has been met.
 Target Group	 Big Picture			 Product Details	
<p>The users and the customers with their needs.</p> <p>Personas are a great way to describe the target group.</p>	<p>The desired user experience (UX): the user journeys, the product functionality, the visual design, and the nonfunctional properties.</p> <p>Epics, scenarios, storyboards, workflows, design sketches, mock-ups, and constraint stories are helpful techniques.</p>			<p>The goal of the next iteration with specific actionable items to reach the goal.</p> <p>The items are ordered from one to n, and may be captured as detailed user stories.</p>	

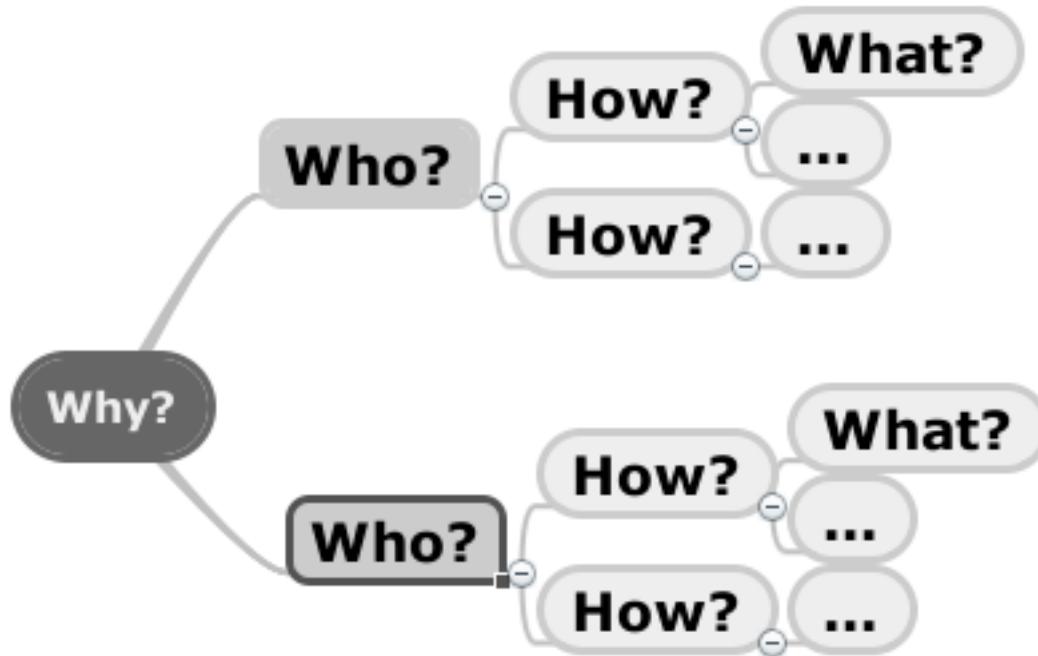
<http://www.romanpichler.com/>



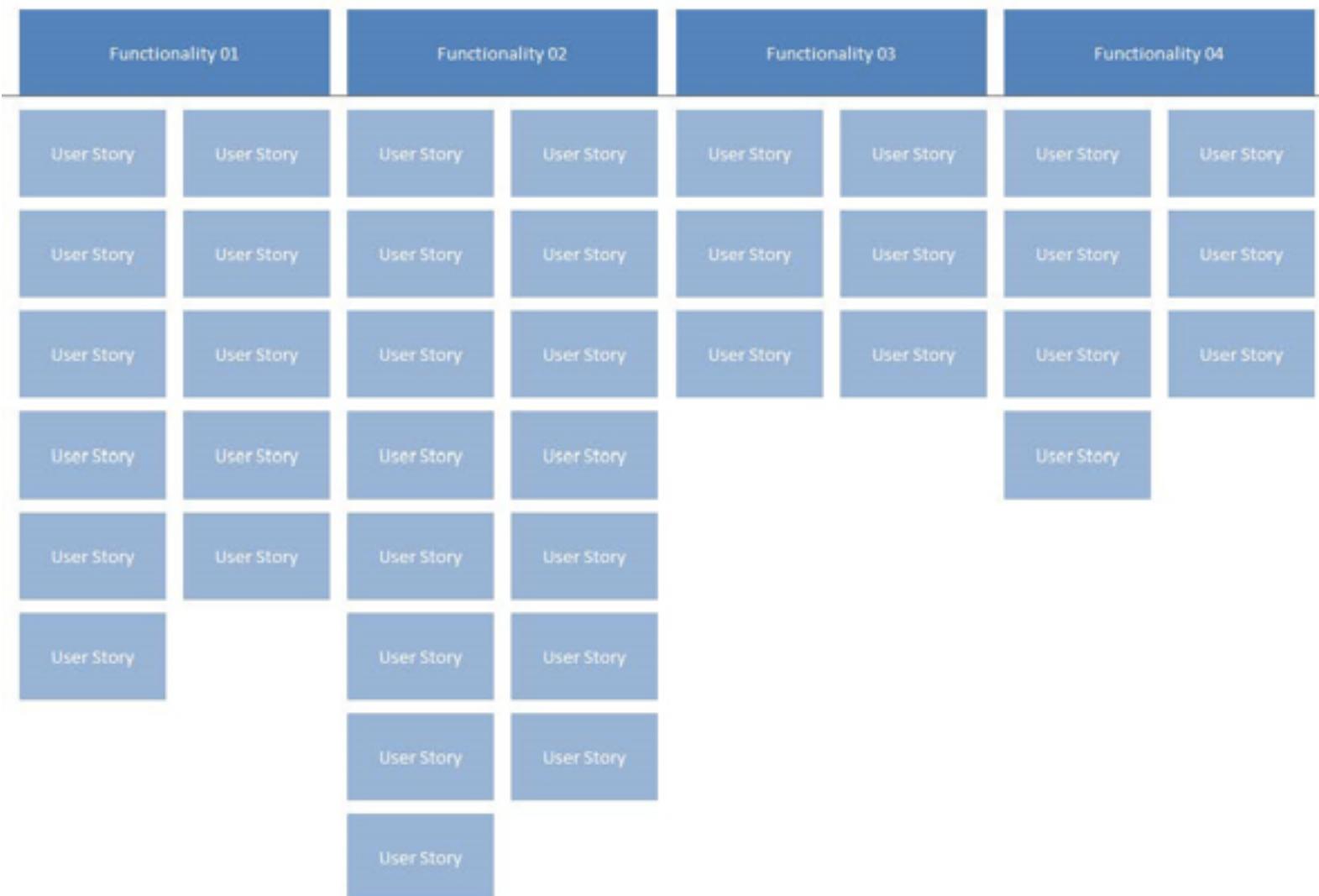
www.romanpichler.com

info Share
<academy>

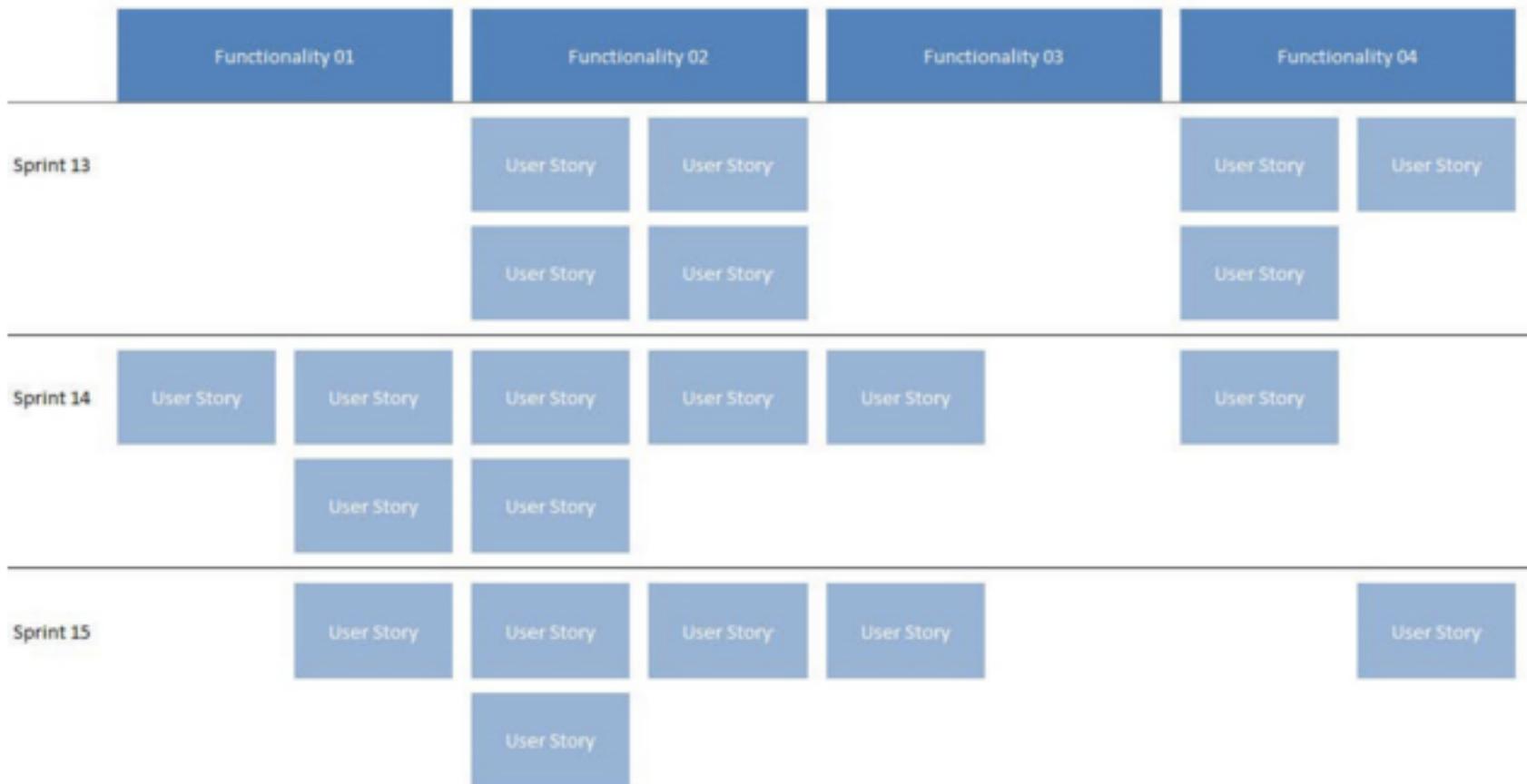
Impact Mapping



Story Mapping - krok 1



Story Mapping - krok 2



Planowanie Sprintu

SP1

WHAT

- commitment
- check acceptance criteria
- SM reminds
 - leave public holidays
 - previous velocity

when?

1st thing of the sprint.

time?

1hr per week of sprint
(2week sprint = 2hours)

who?

team, PO, SM, anyone with knowledge

SP2

HOW

- design session
- solutioning
- how team will deliver this
- might create tasks ☺

when?

after SP1

time?

1hr per week of sprint

who?

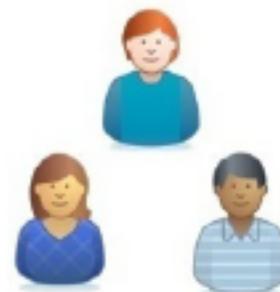
team, SM, PO on call

Sprint

Daily Scrum (Codzienny Scrum)

Co zrobiłem
wczoraj co pomogło
osiągnąć Cel Sprintu
?

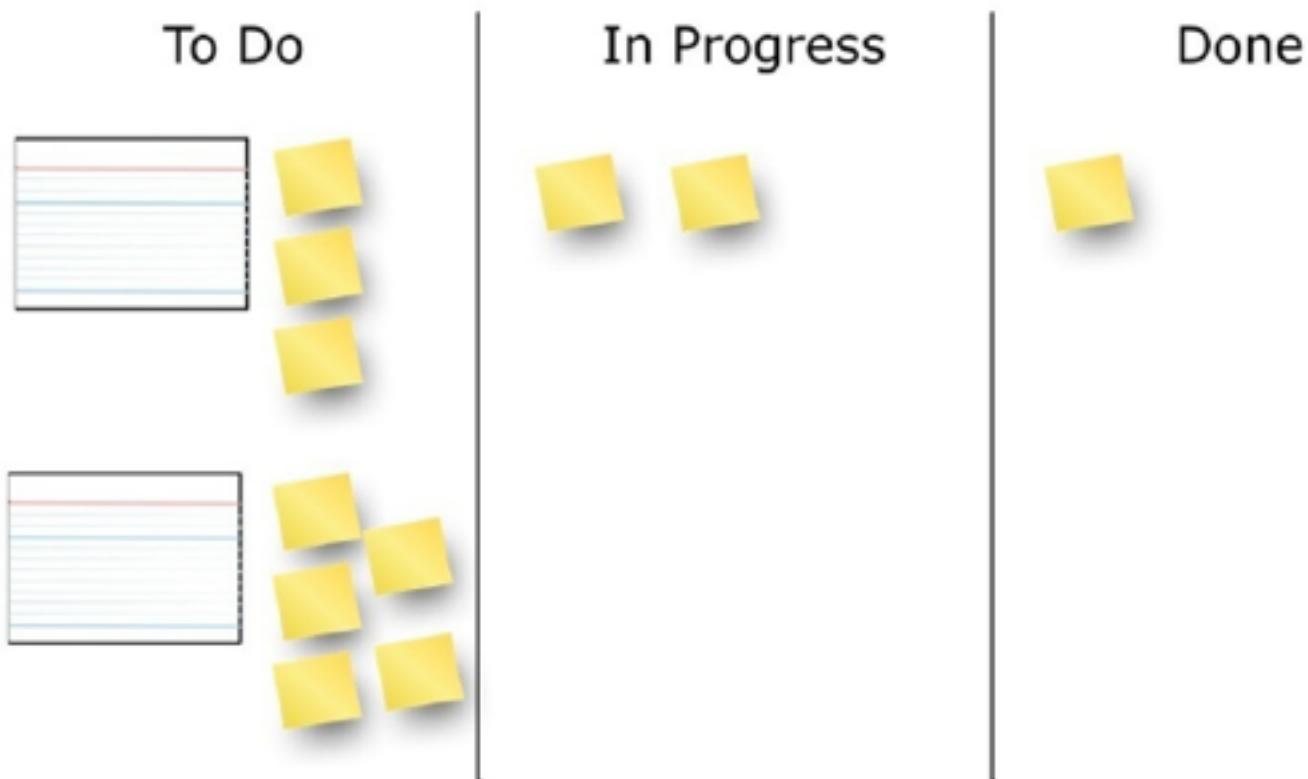
Co zrobię dziś, co
pomoże osiągnąć
Cel Sprintu ?

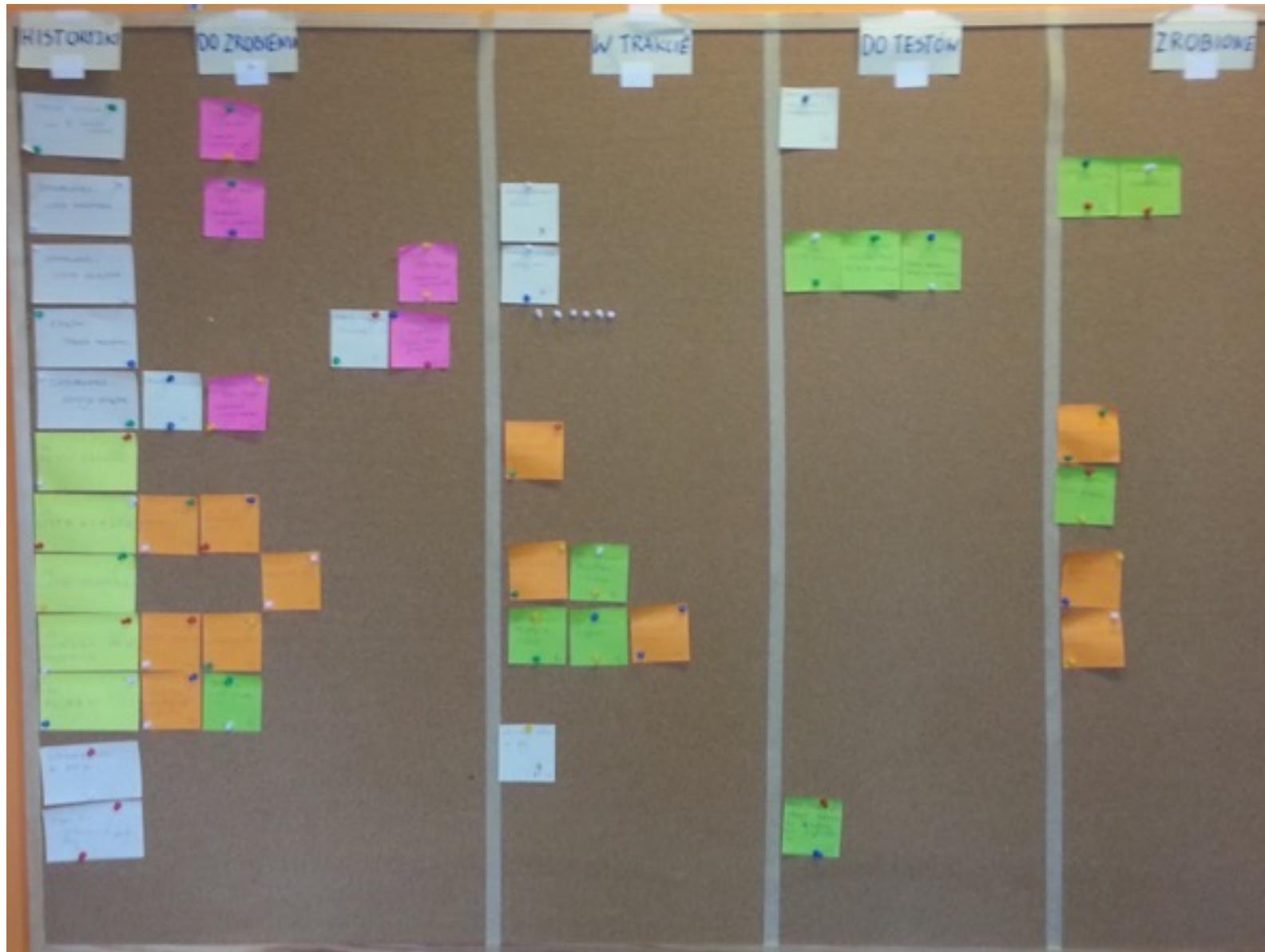


Czy widzę jakieś
przeszkody w
osiągnięciu Celu
Sprintu?

Sprint Backlog

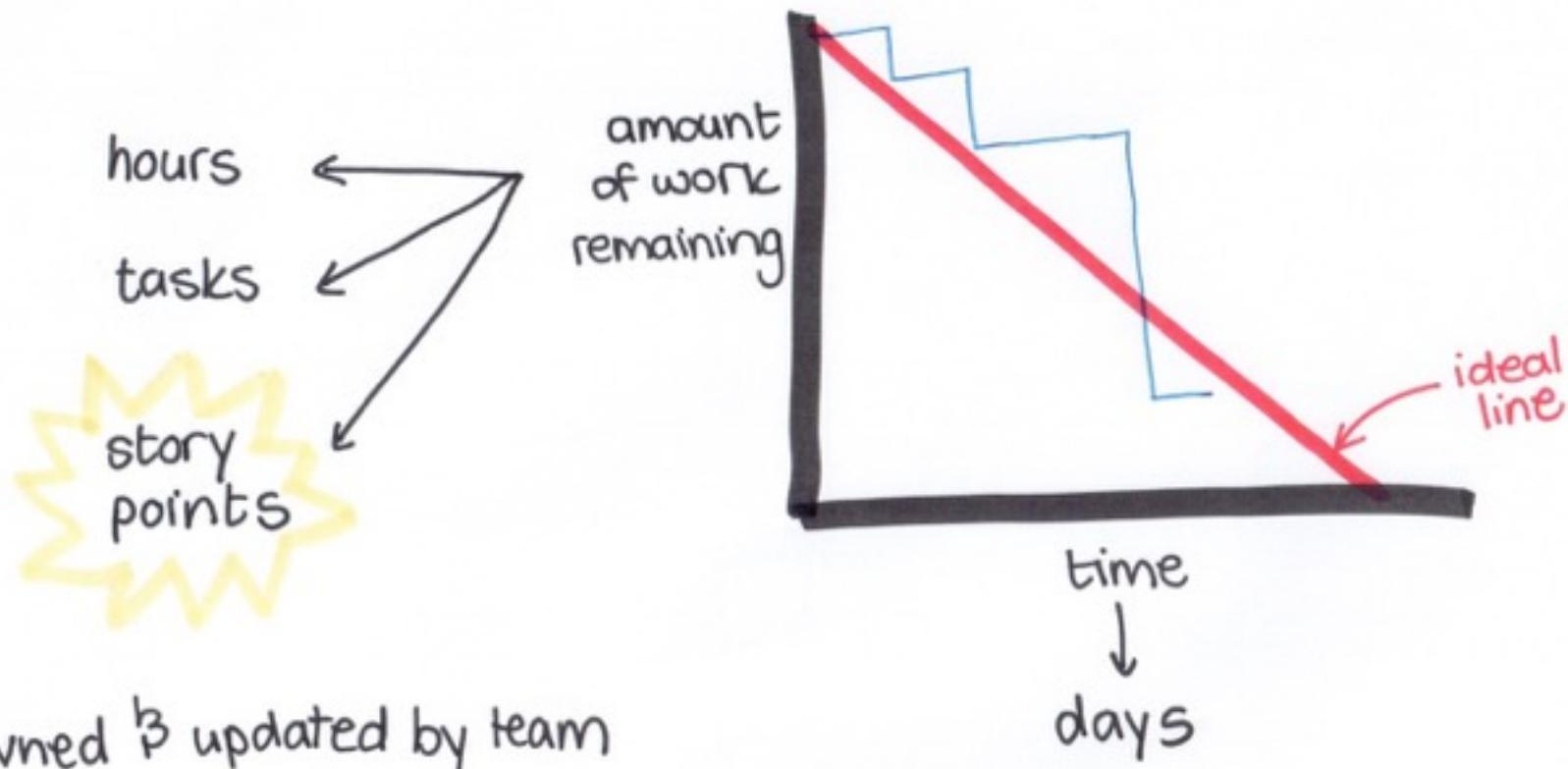
Sprint Backlog



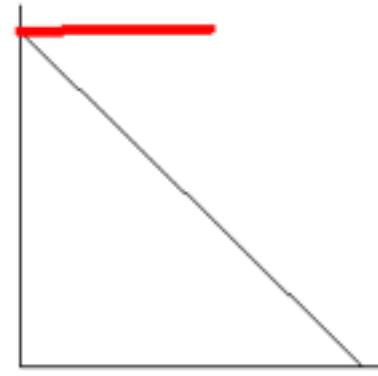
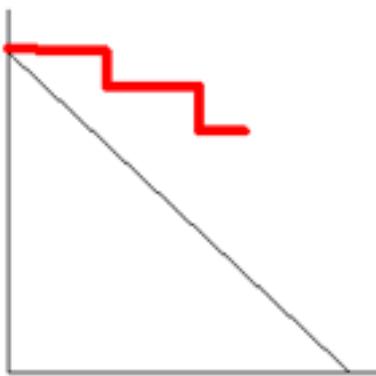
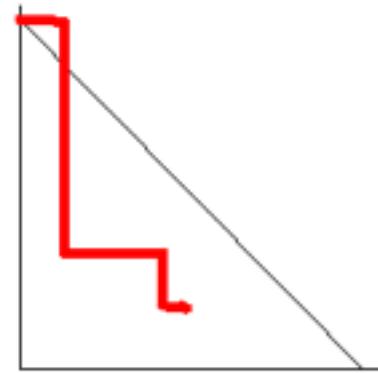
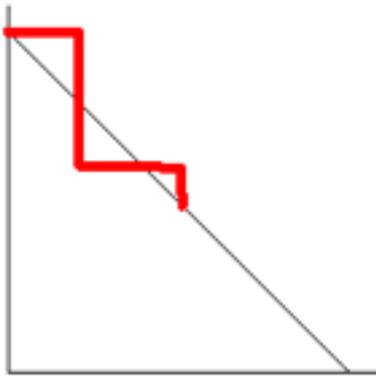




SPRINT BURNDOWN



*Co pokazują poniższe
wykresy spalania ?*



Przegląd Sprintu

Sprint Review

- ✓ Demonstracja pracy zespołu
- ✓ Tylko ukończone Produkty są prezentowane
- ✓ PO, SM, Zespół Scrumowy, wszyscy co mogą dać informację zwrotną

Sprint Review

- ✓ Trwa do 4 godzin dla miesięcznego Sprint(krócej zazwyczaj)
- ✓ Należy się przygotować do przeglądu Sprintu (około 1 godziny)
- ✓ Przegląd Sprintu to konstruktywna dyskusja, a nie tylko demonstracja

Retrospekywa

Sprint Retrospective

- ✓ Retrospekcja to spojrzenie wstecz, po to by wyciągnąć wnioski na przyszłość.
- ✓ Scrum wymaga retro, z czasem trwania do 3 h dla miesięcznego Sprinta
- ✓ Wynikiem retro ma być lista zadań do wdrożenia
- ✓ W retro biorą udział SM, Zespół, PO jak jest zaufanie między nim a zespołem

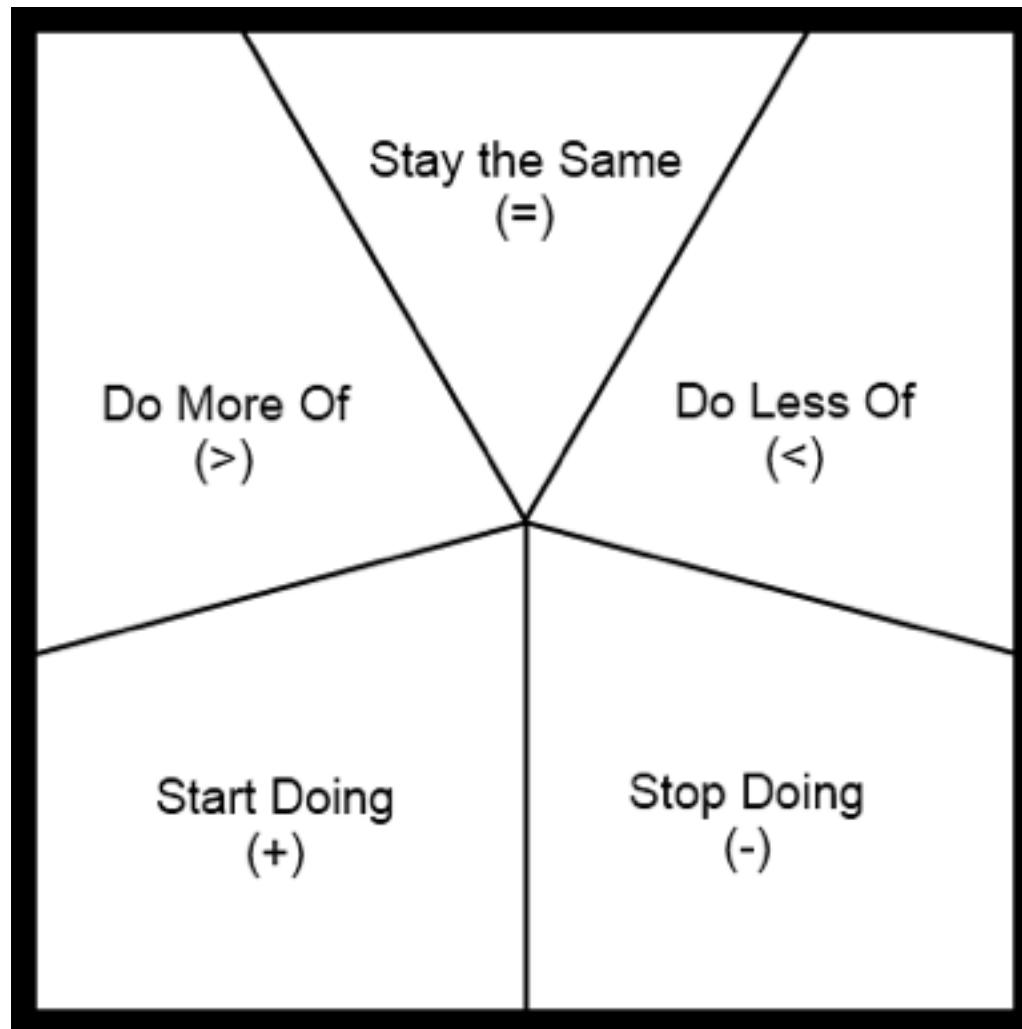
Sprint Retrospective

- ✓ Zespół Scrumowy sprawdza jak poszedł ostatni Sprint
- ✓ Zespół Scrumowy analizuje co należy poprawić i postanawia pewne zmiany
- ✓ Zespół Scrumowy wdraża zaproponowane zmiany w kolejnych Sprintach
- ✓ Scrum Master bierze udział w Retrospektywie Sprintu, jako członek zespołu odpowiedzialny za Sprint

Różne podejścia do retro

PRODUKT ZESPÓŁ BIZNES

StarFish



Wiele sposobów na retrospektywę



Esther Derby
Diana Larsen

Dziękuję i życzę udanych
projektów z zastosowaniem
Scrum

Następne kroki

Przeczytaj Scrum Guide

Do pobrania na stronie:

<http://www.scrumguides.org/>



Scrum Guide

Przewodnik po Scrumie:
Reguły Gry





Lipiec 2013

Przygotowany i utrzymywany przez Kena Schwabera
i Jeffa Sutherlanda

 Scrum.org

Improving the Profession of Software Development

Warto przeczytać

Scrum and XP from the Trenches - 2nd Edition,
Henrik Kniberg

Agile Software Development with Scrum, Ken Schwaber

Agile Product Management with Scrum, Roman Pichler

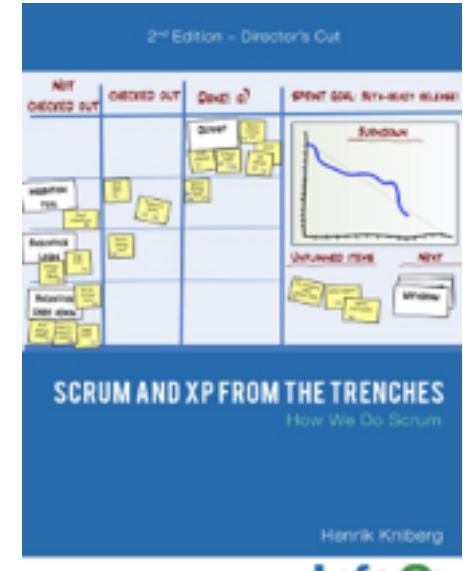
Agile Estimating and Planning, Mike Cohn

User Stories Applied, Mike Cohn

Coaching Agile Teams, Lyssa Adkins

Agile Retrospectives: Making Good Teams Great, Esther Derby

Scrum Mastery: From Good To Great Servant-Leadership, Geoff Watts



Sprawdź swoją wiedzę podczas
Open Assessment - 30 pytań

Aż otrzymasz 100% poprawnych odpowiedzi

<https://www.scrum.org/Assessments/Open-Assessments>

Podejdź do testu PSM I

- ✓ Dostępny na stronie: www.scrum.org/Assessments/Professional-Scrum-Master-Assessments/PSM-I-Assessment
- ✓ Koszt 150\$
- ✓ Wymagany procent poprawnych odpowiedzi - 85%
- ✓ Czas 60 minut
- ✓ Ilość pytań - 80
- ✓ Język testu: angielski

PREPARE

GET
PASSWORD

START
ASSESSMENT

Powodzenia :)



Źródła

- ✓ "Growing Agile: A Coach's Guide to Training Scrum" Karen Greaves, Samantha Laing
- ✓ "Agile w praktyce" Andy Brandt
- ✓ "Agile Answers" Andy Brandt, Kristian Kaczor
- ✓ "Scrum. O zwinnym zarządzaniu projektami" Mariusz Chrapko
- ✓ "Are we agile yet" - Tomasz Włodarek
- ✓ The Management Plaza PSM I Simulated Exams.
- ✓ <http://scrumtrainingseries.com/>
- ✓ Scrum Narrative and PSM Exam Guide Mohammed Musthafa Soukath Ali
- ✓ <http://www.agile247.pl/>
- ✓ www.scrumguide.org
- ✓ <http://blog.wlochowicz.com/>
- ✓ www.scrum.org
- ✓ www.agilealliance.org