

WPROWADZENIE Do SCRUM

„Ludzie i interakcje ponad procesy i narzędzia

Metodyki tradycyjne (PRINCE)



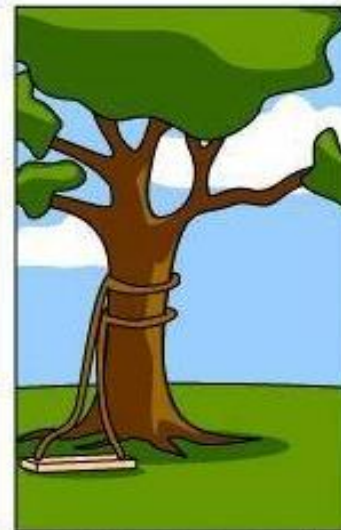
How the customer explained it



How the Project Leader understood it



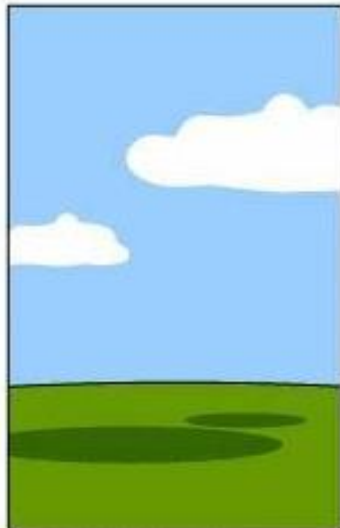
How the Analyst designed it



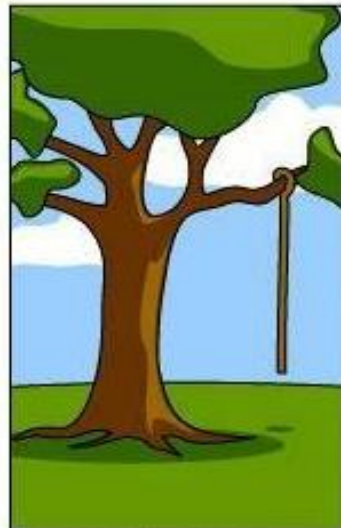
How the Programmer wrote it



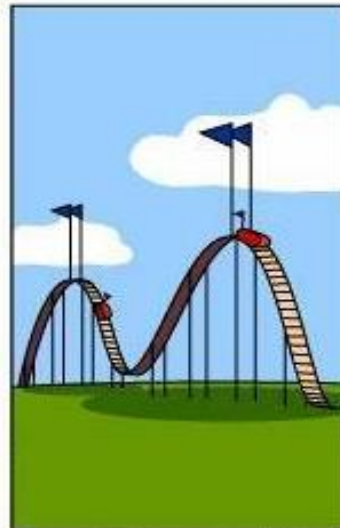
How the Business Consultant described it



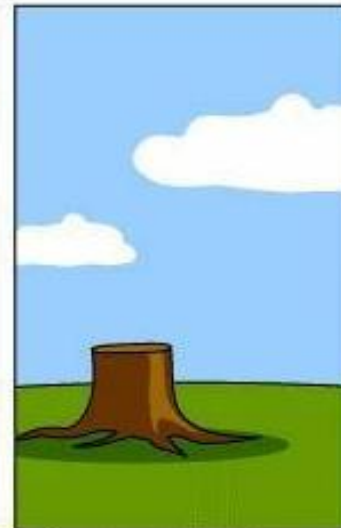
How the project was documented



What operations installed



How the customer was billed



How it was supported



What the customer really needed

Straty w biegu sztafetowym

Proces „sztafety” w wytwarzaniu oprogramowania często pozostaje w konflikcie z jej głównym celem tj. maksymalną szybkością i uzyskaniem odpowiedniej elastyczności produktu....

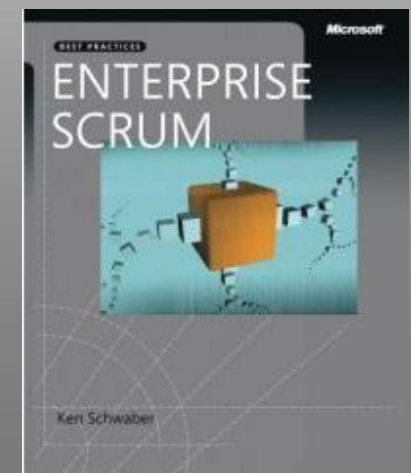
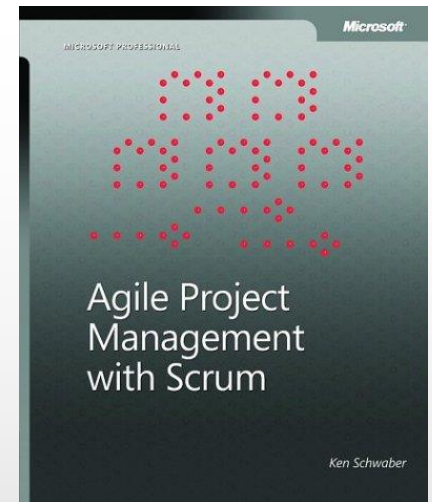
W zamian: holistyczne (proces „rugby”) podejście do projektu — gdzie zespół próbuje przebiec dystans jako całość, poddając piłkę do tyłu i do przodu — znacznie efektywniej pozwala na realizację wysublimowanych wymagań dzisiejszych systemów informatycznych

Założenia - Scrum

- **Scrum zwinny** (agilny) proces pozwalający i skupiający się na dostarczeniu najwyższej wartości biznesowej (produkt) w możliwie najkrótszym czasie.
- **Scrum pozwala** szybko i powtarzalnie prowadzić inspekcję bieżąco funkcjonującego oprogramowania (inspekcje – przeglądy co dwa tygodnie w miesiącu).
- **Wymagania biznesowe** klienta ustalają priorytety do realizacji. Zespół organizuje się sam dla wybrania najlepszej drogi do dostarczenia najlepszego prod.
- **Co dwa tygodnie** (do miesiąca) każdy z zespołu obserwuje realnie pracujące oprogramowanie, wydaje je lub kontynuuje prace dla jego zmiany w następnym sprincie.

Scrum historia

- 1986 Hirotaka Takeuchi Ikujiro Nonaka (eksperci biznesu) – założenia metodyki „Harvard Business Review”
- 1993 Jeff Sutherland -wprowadza SCRUM jako framework w Easel Corporation.
- Scrum przedstawione na OOPSLA 96
 - **OOPSLA** (Object-Oriented Programming, Systems, Languages & Applications) coroczna konferencja naukowa Association for Computing Machinery w USA
- 1993 – 2000 razem z Ken Schwaber
MANIFEST SCRUM
 - Założenie stowarzyszenia Scrum Alliance w 2002, początkowo w stowarzyszeniu Agile Programming
 - Od 2010 Scrum Alliance oddzielnie



Scrum jest wykorzystywane przez:

- Microsoft
- Yahoo
- Google
- Electronic Arts
- High Moon Studios
- Lockheed Martin
- Philips
- Siemens
- Nokia
- Capital One
- BBC
- Intuit
- Intuit
- Nielsen Media
- First American Real Estate
- BMC Software
- Ipswitch
- John Deere
- Lexis Nexis
- Sabre
- Salesforce.com
- Time Warner
- Turner Broadcasting
- Oce

Scrum jest używane do/dla/w:

- Budowanie oprogramowania komercyjnego
- Prowadzenie kontraktów
- Projekty o narzuconej cenie
- Aplikacje finansowe
- ISO 9001-projekty
- Systemy wbudowane
- 24x7 systemy 99.999% efektywności działania
- Gry strategiczne - Joint Strike Fighter
- Duże systemy informatyczne
- Video gry
- FDA- potwierdzone, krytyczne dla życia systemy (food @ drug administration)
- Oprogramowanie kontroli satelitów
- Oprogramowanie stron www
- Oprogramowanie podręczne
- Aplikacje dla systemów mobilnych
- Aplikacje obsługi i administracji sieci i funkcjonowania sieci
- Aplikacje dla specjalizowanych systemów operacyjnych

Charakterystyka ogólna SCRUM

- Samo–organizujący się zespół
- Rozwój produktu w serii miesięcznych sprintów
- Wymagania są ujmowane jako produkty w liście zwanej portfelem produktów “product backlog”
- Nie ma specyfikacji praktyk inżynierskich
- Użycie twórczych zasad kreujących środowisko agilne dla dostarczenie projektu
- Jeden z tzw. “agile processes”

Manifest – metodyki agilne zestawienie wartości

Indywidualność i
współdziałanie

ponad

Procesy i
narzędzia

Pracujące
oprogramowanie

ponad

Wszechstronna
dokumentacja

Współpraca z
Klientem

ponad

Negocjacje kontraktu
formalne ustalenia

Reagować na
zmiany

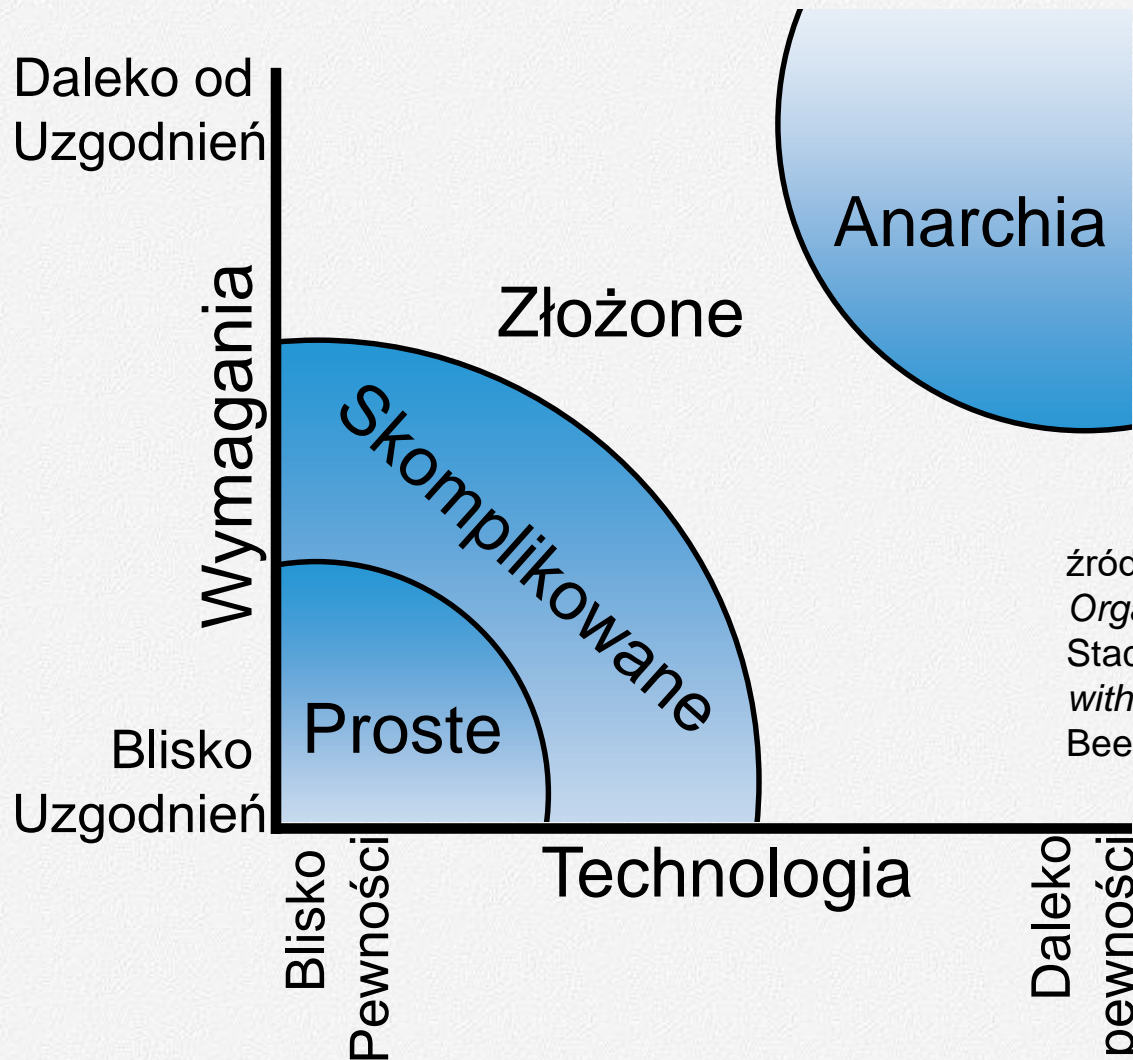
ponad

Podążanie za
planem

Bardziej cenimy

Doceniamy

Wymagania a Technologia

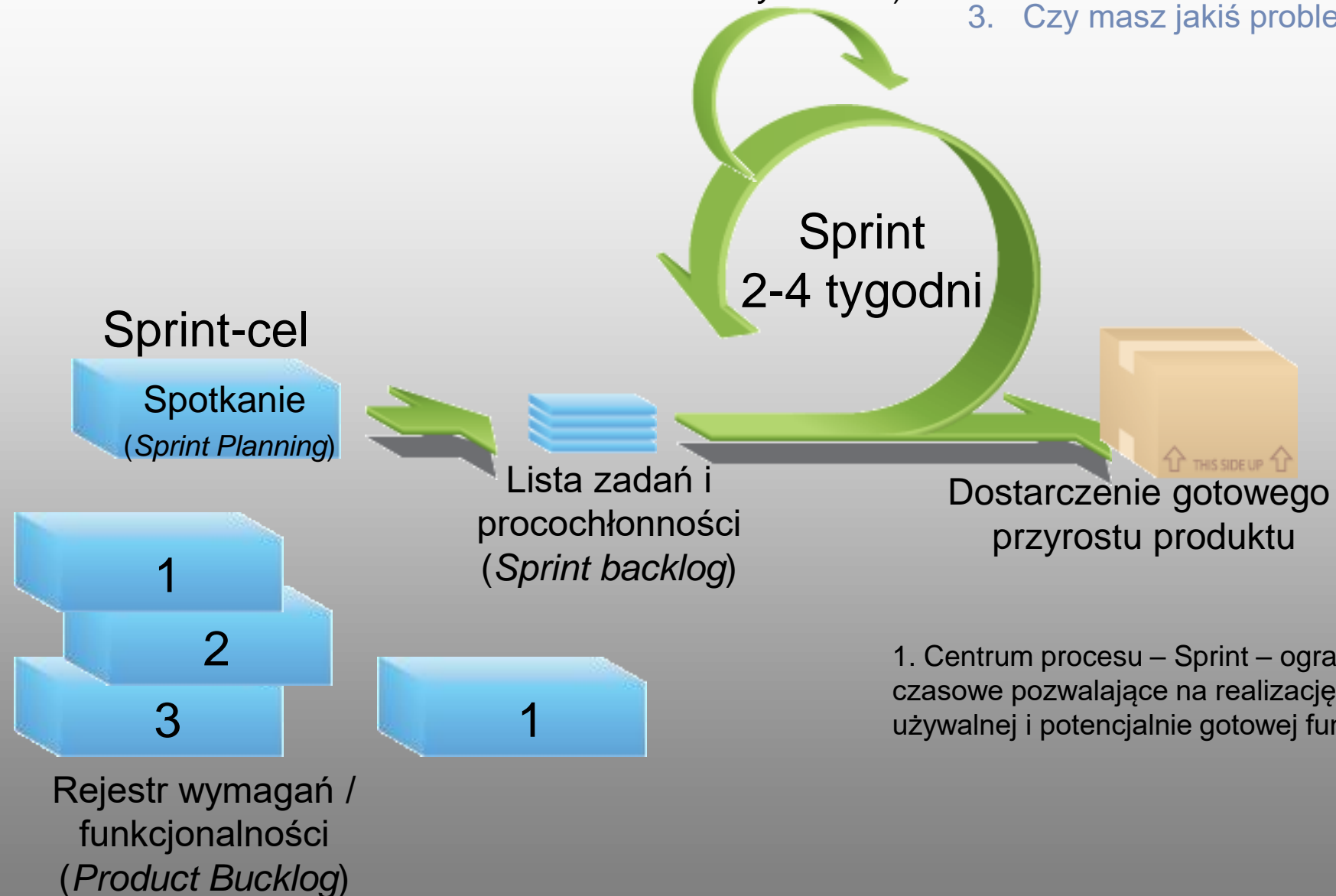


źródło: *Strategic Management and Organizational Dynamics* by Ralph Stacey in *Agile Software Development with Scrum* by Ken Schwaber and Mike Beedle.

Scrum

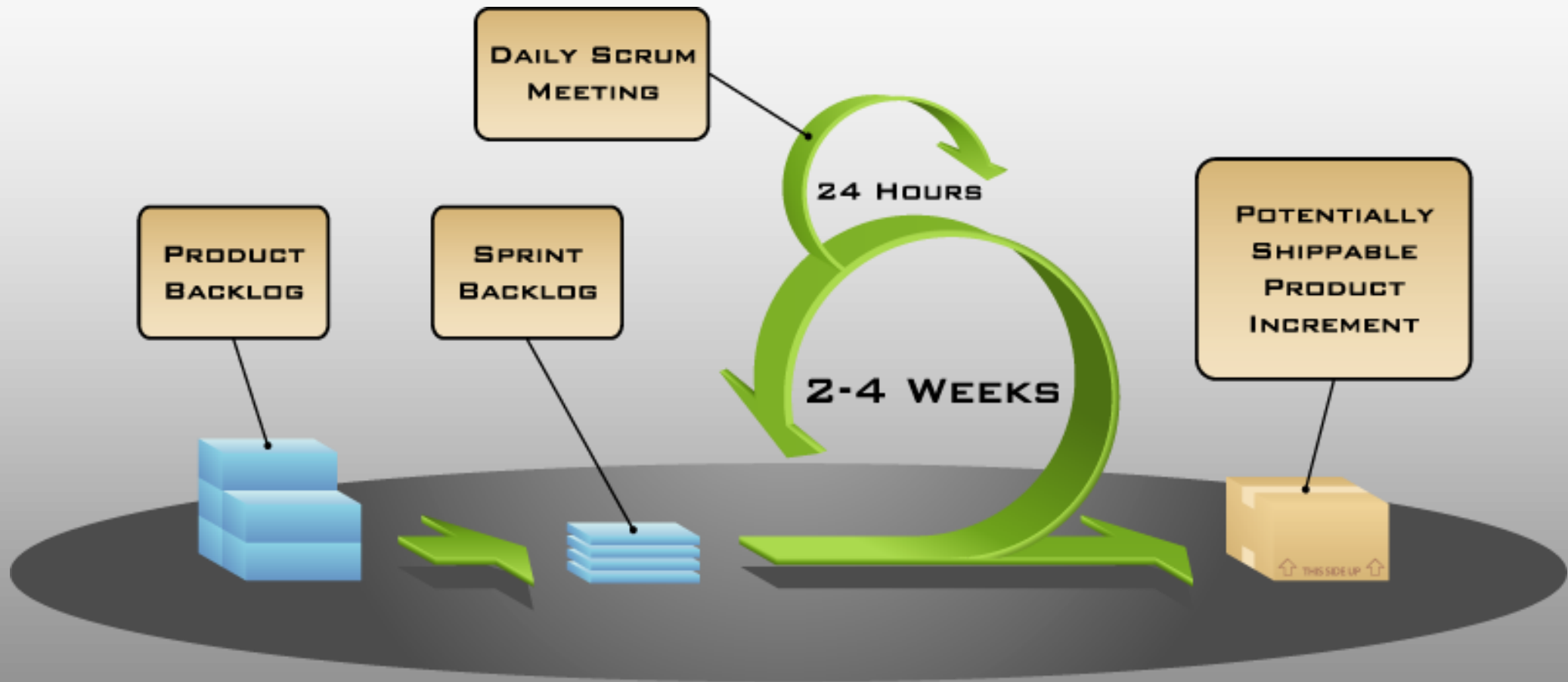
24 godziny
(spotkania 15min
Daily Scrums)

1. Nad czym pracowałeś wczoraj?
2. Nad czym będziesz pracować dzisiaj?
3. Czy masz jakiś problem?



1. Centrum procesu – Sprint – ograniczenie czasowe pozwalające na realizację PRZYROSTU - używalnej i potencjalnie gotowej funkcjonalności

SCRUM



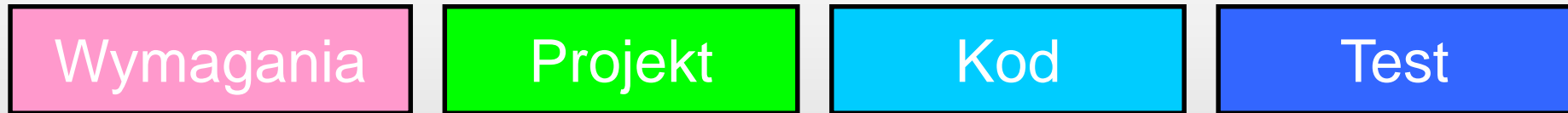
COPYRIGHT © 2005, MOUNTAIN GOAT SOFTWARE

Image available at
www.mountaingoatsoftware.com/scrum

Sprints

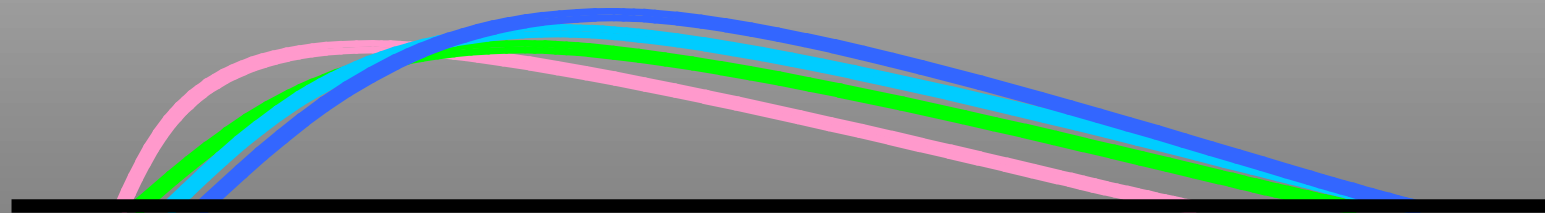
- Postęp w realizacji projektu następuje w serii tzw. Sprintów – analogicznie jak w kolejnych iteracjach programowania ekstremalnego
- Typowy czas dla sprintu – 2-4 tygodni, nie więcej jednak jak miesiąc
- Stała długość sprintu pozwala utrzymać odpowiedni rytm pracy
- W obrębie sprintu produkt jest projektowany, kodowany i testowany

Sekwencyjne vs. zazębione (współbieżne z przesunięciem –overlapping) wytwarzanie – rozwój oprogramowania

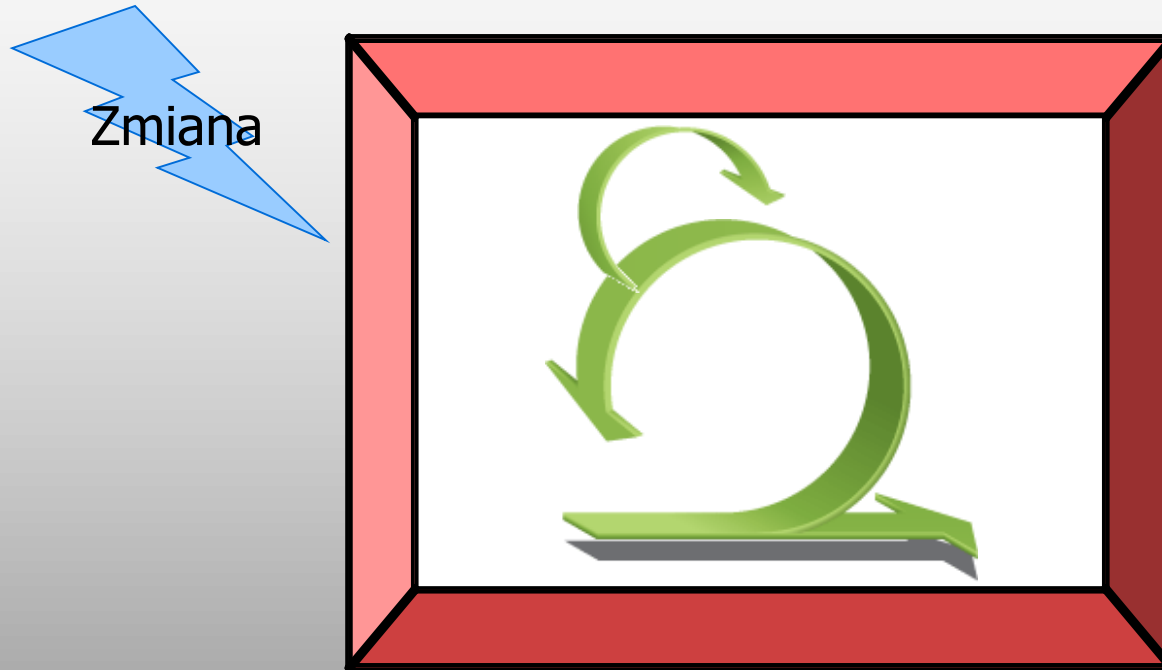


Zamiast realizować kolejne elementy w swoim czasie...

...zespół Scrum realizuje wszystkiego po trochu przez cały czas trwania projektu



W obrębie sprintu nie ma zmian



- Czas trwania sprintu powinien być tak zaplanowany aby gwarantował brak konieczności wprowadzania zmian do sprintu.

Scrum framework

Role

- Właściciel produktu (*Product owner*)
- Mistrz „Młyna” (*ScrumMaster*)
- Zespół (*Team*)

Zdarzenia

- Spotkanie (*Sprint planning*)
- Widok zadania (*Sprint review*)
- Podsumowanie sprintu (*Sprint retrospective*)
- Odprawa scrum (*Daily scrum meeting*)

Artefakty

- Rejestr funkcjonalności (*Product backlog*)
- Lista zadań i pracochońność (*Sprint backlog*)
- Czas do końca projektu (*Burndown charts*)

Scrum framework

Role:

- Właściciel Produktu
- Mistrz
- Zespół

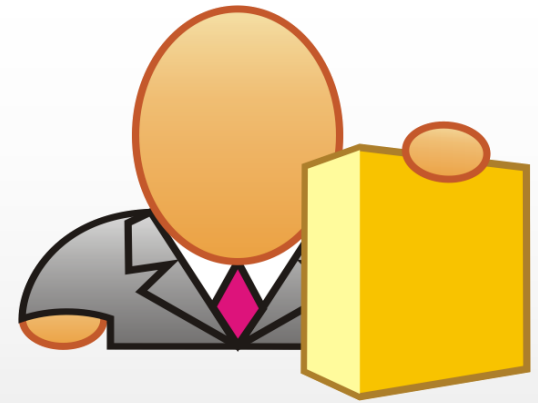
Wydarzenia:

- Planowanie sprintu
- Przegląd sprintu
- Podsumowanie sprintu
- Dzienna odprawa

Artefakty:

- Rejestr funkcjonalności
- Lista zadań
- Czas do końca (wypalenie)

Właściciel produktu



- Definiuje własności produktu
- Decyduje o terminie wydania produktu
- Jest odpowiedzialny za dodatnie profity z produktu
- Ustala priorytety funkcjonalności w dopasowaniu do rynku
- Dopasowuje własności i priorytety na każdej iteracji jeżeli potrzeba
- Akceptuje lub odrzuca rezultaty prac

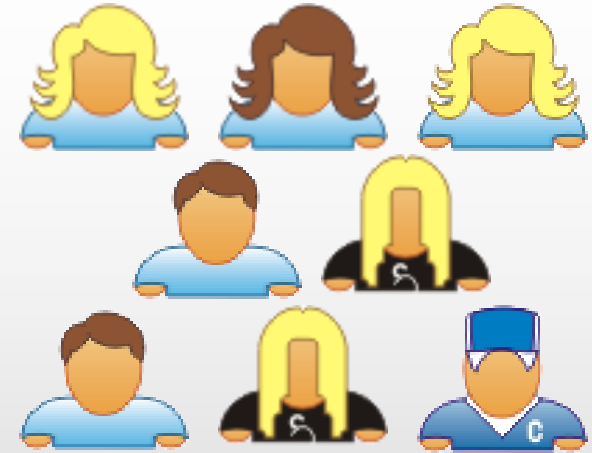
Mistrz Młyna ScrumMaster



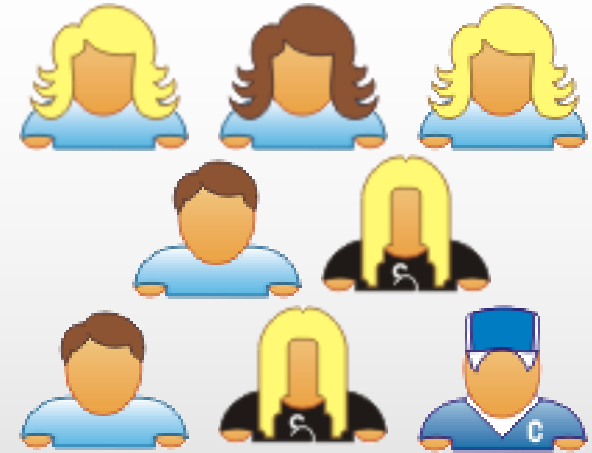
- Reprezentuje zarządzanie w projekcie
- Odpowiedzialny za realizację zasad i praktyk SCRUM
- Usuwa przeszkody
- Zapewnia pełną funkcjonalność i produktywność zespołu
- Umożliwia ścisłą współpracę pomiędzy wszystkimi aktorami i ich rolami
- Osłania zespół przed zewnętrznymi czynnikami

Zespół

- Zwykle 5 – 9 ludzi
- Funkcyjnie przekrojowi:
 - Programiści, testerzy, doświadczeni przez użytkowników projektanci, etc.
 - Członkowie zespołu powinni być zatrudnieni na stałe (Full-time)
 - Mogą być wyjątki (n.p., Administrator bazy danych)



Zespół



- Zespół organizuje się sam
- Idealnie, żadnych tytułów raczej możliwości
- Uczestnictwo w zespole powinno się zmieniać jedynie pomiędzy sprintami

Scrum framework

Role

- Właściciel Produktu
- Mistrz
- Zespół

Wydarzenia

- Planowanie sprintu
- Przegląd sprintu
- Podsumowanie sprintu
- Dzienna odprawa

Artefakty

- Rejestr funkcjonalności
- Lista zadań
- Czas do końca (wypalenie)



Planowanie Sprintu

- Zespół wybiera zadania z rejestru funkcjonalności które zobowiązują się wykonać
- Lista zadań zostaje wykreowana
 - Zadania są zidentyfikowane i każde jest oszacowane (1-16h)
 - Wspólnie, nie samodzielnie przez ScrumMastera
- Uwzględniamy projektowanie wyższego poziomu

Jako planujący wakacje, Chcę zobaczyć zdjęcie hoteli.

Kodowanie pośredniego poziomu (8h)
Kod interfejsu użytkownika (4)
Testowanie połączeń (4)
Kodowanie Klas (6)
Testowanie (4)

Odprawa (Daily scrum)

- Parametry
 - Codziennie
 - 15-minut
 - Na stojaco (w biegu)
- Nie ma służyć rozwiązaniu problemów
 - Wszyscy są zaproszeni (każdy może słuchać)
 - Tylko członkowie zespołu, ScrumMaster, właściciel produktu, mogą mówić
- Pomaga uniknąć innych niepotrzebnych spotkań



Każdy odpowiada na 3 pytania

1

Co robiłeś wczoraj?

2

Co będziesz robił dzisiaj?

3

Czy masz jakiś problem?

- One **nie są** zobowiązaniami dla ScrumMastera
- Są to zobowiązania przed rówieśnikami!!

Przeгляд sprintu

- Zespół przedstawia co zostało zrealizowane w sprincie
- Zwykle przyjmuje to formę prezentacji nowej funkcjonalności i architektury architecture
- Nieformalnie
 - 2-h prezentacja zwykle
 - Nie przeźrocza
- Cały zespół bierze udział
- Zapraszamy otoczenie



Podsumowanie Sprintu

- Okresowo jest sprawdzane co działa i nie działa w projekcie
- Zwykle 15–30 minut
- Realizowane po każdym sprincie
- Cały zespół bierze udział
 - ScrumMaster
 - Właściciel produktu (designer, twórca)
 - Zespół
 - Możliwie klient i zaproszeni

Start / Stop / Continue

- Cały zespół dyskutuje co dalej chcieliby robić:

Zaczynamy pracę

Kończymy pracę

To jedna z wielu
drog realizacji
podsumowania
sprintu.

Kontynuujemy pracę

Scrum framework

Role

- Właściciel Produktu
- Mistrz
- Zespół

Wydarzenia

- Planowanie sprintu
- Przegląd sprintu
- Podsumowanie sprintu
- Dzienna odprawa

Artefakty

- Rejestr funkcjonalności
- Lista zadań
- Czas do końca (wypalenie)

Scrum framework

Role

- Właściciel Produktu
- Mistrz
- Zespół

Wydarzenia

- Planowanie sprintu
- Przegląd sprintu
- Podsumowanie sprintu
- Dzienna odprawa

Artefakty

- Rejestr funkcjonalności
- Lista zadań
- Czas do końca (wypalenie)

Rejestr Funkcjonalności (product backlog)

- Wymagania
- Lista wszystkich koniecznych prac w projekcie
- Wyrażonych jasno tak że każdy element ma wartość dla użytkownika i klienta produktu
- Uszeregowany przez właściciela produktu
- Ustawione priorytety na starcie każdego sprintu



To jest
product backlog

Przykładowy rejestr funkcjonalności

Element rejestru	Szacowanie
Pozwala gościowi zrobić rezerwację	3
Jako gość chcę usunąć rezerwację	5
Jako gość chcę zmienić datę rezerwacji	3
Jako hotelarz, chcę uruchomić aplikację wskazującą na dostępność pokoi	8
Popraw obsługę wyjątków	8
...	30
...	50

Cel Sprintu

- Krótkie zdanie opisujące na jakim zdarzeniu będzie skupiona praca systemu w realizowanym sprincie

Nauki o życiu

Dostarczają funkcjonalności dla studiów nad genetyką populacji

Zastosowanie Bazy Danych

Pozwala uruchomić aplikację na SQL-owej bazie ORACLE.

Wsparcie finansowe

Wspiera stronę techniczną działania firmy ABC w czasie rzeczywistym.

Zarządzanie Rejestrem Funkcjonalności

- Pracownicy decydują o pracy którą wybierają do realizacji
- Praca nigdy nie jest przypisana
- Pozostała do zrealizowania praca jest co dziennie estymowana

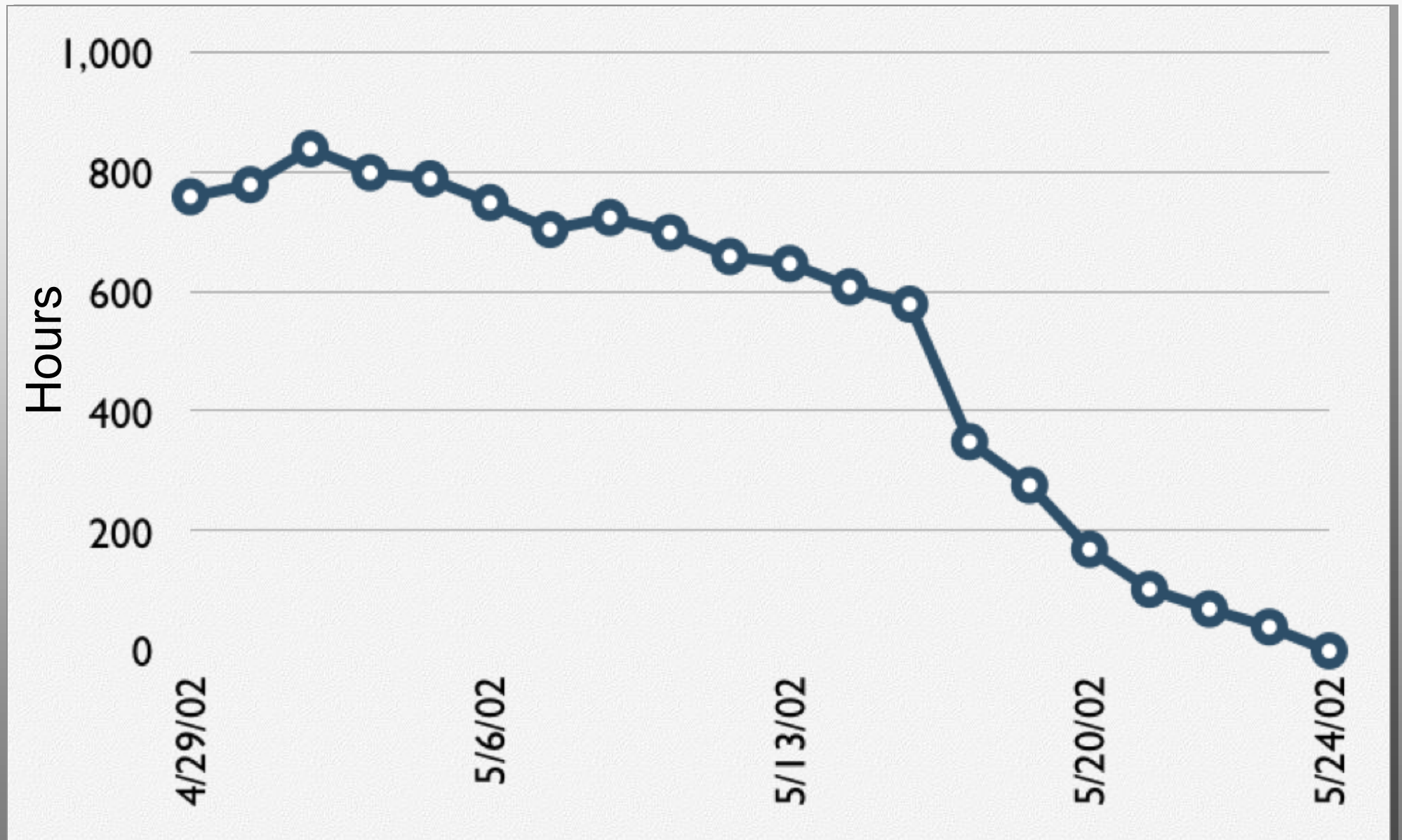
Zarządzanie rejestrem sprintu

- Każdy członek zespołu może dodać, usunąć lub zmienić rejestr sprintu
- Praca w sprincie wzrasta
- Jeżeli zadanie nie jest jasne zdefiniuj element rejestru sprintu z dłuższym czasem i zakończ go później
- Popraw (Skoryguj) pozostałe pracy w momencie gdy staną się „jaśniejsze”

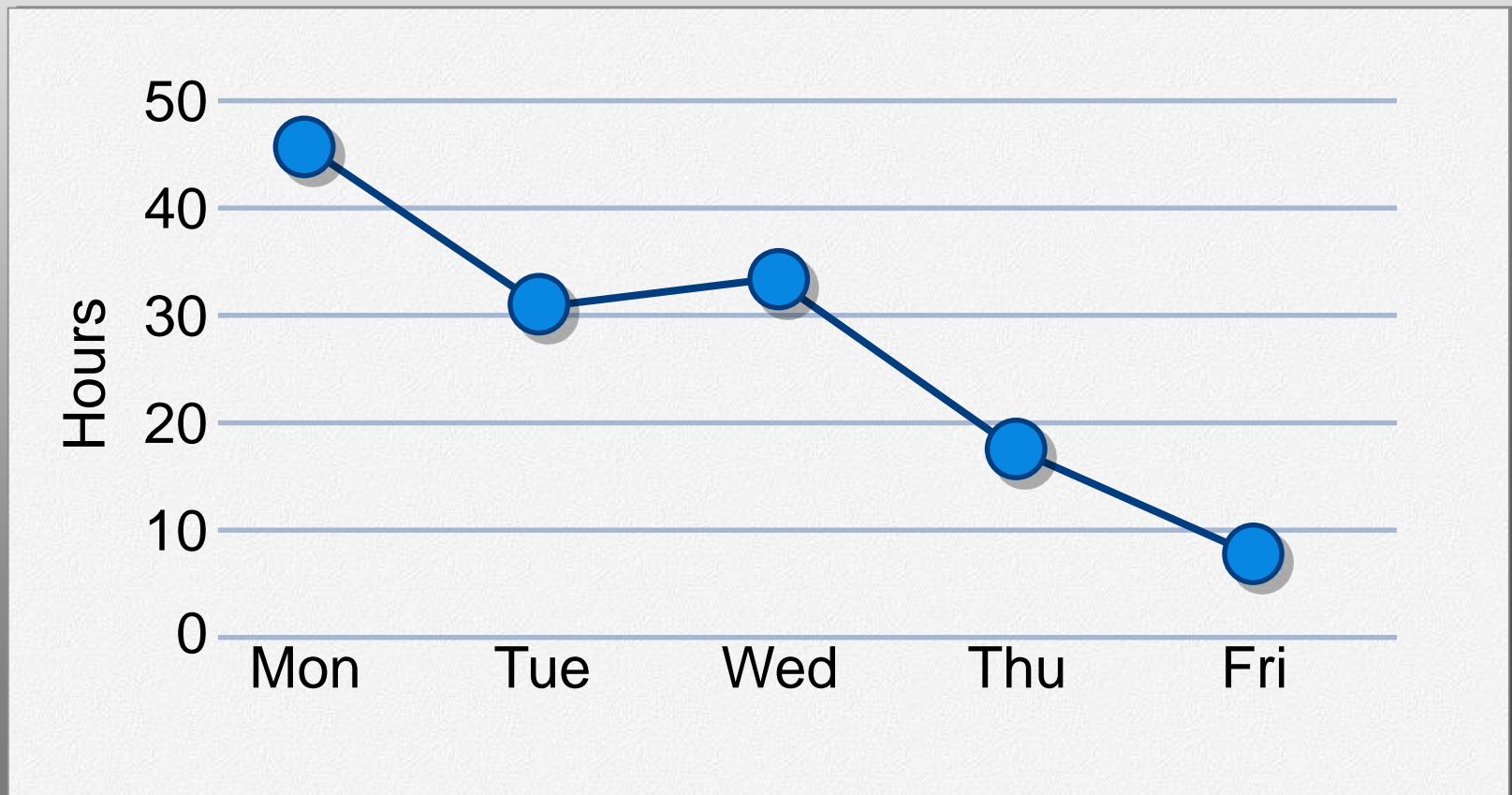
Rejestr sprintu

Zadania	Pn	Wt	Śr	Czw	Pt
Koduj interface użytkowni.	8	4	8		
Koduj poziom pośredni	16	12	10	4	
Testuj poziom pośredni	8	16	16	11	8
Napisz help	12				
Napisz bla, bla, klasy	8	8	8	8	8
Dodaj rejestrację błędów			8	4	

Diagram „wypalenia” sprintu



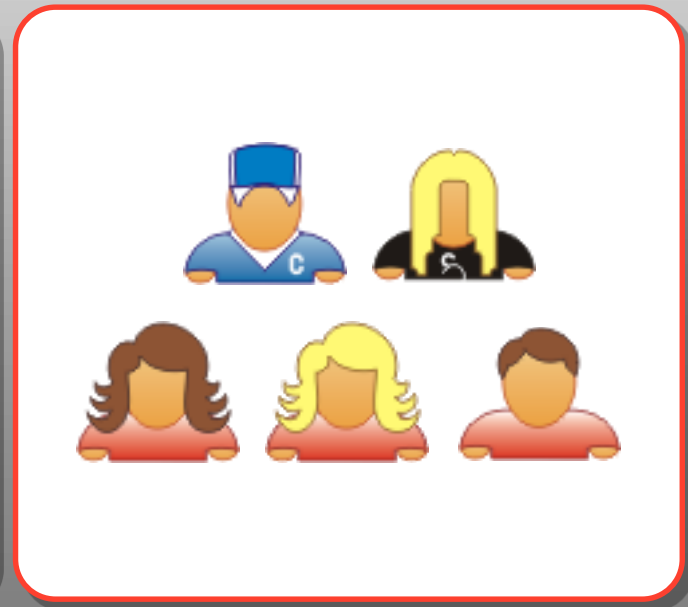
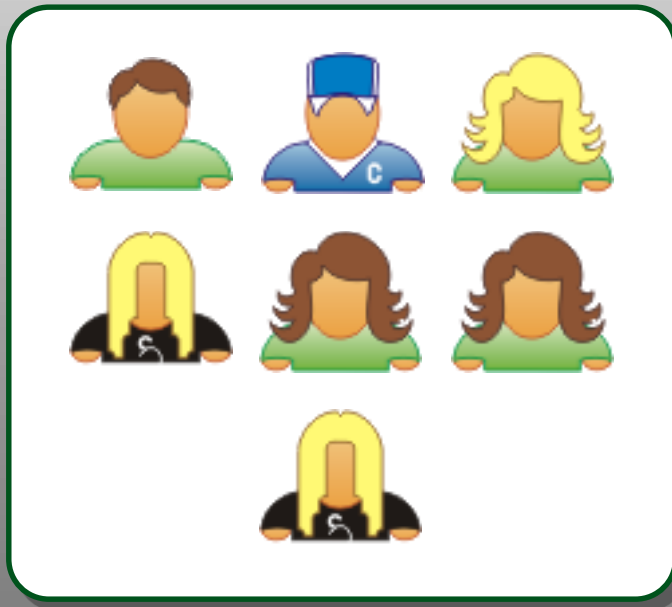
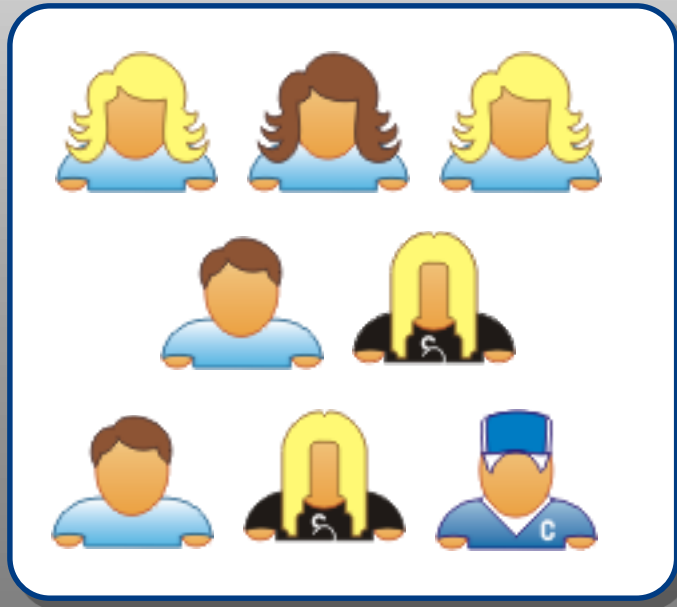
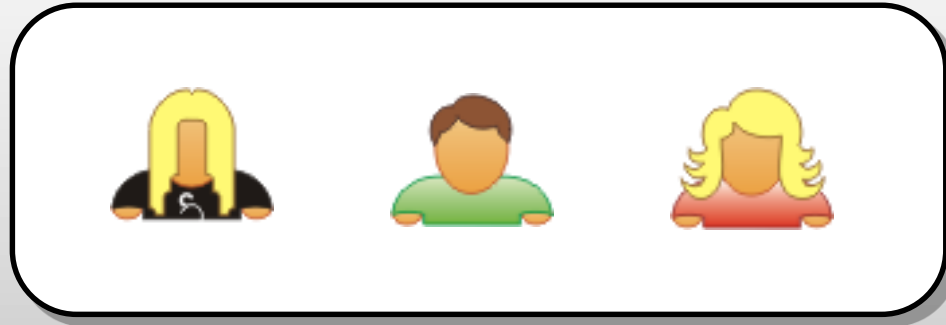
Zadania (tasks)	Pn	Wt	Śr	Czw	Pt
Koduj interface użytko.	8	4	8		
Koduj poziom pośredni	16	12	10	7	
Testuj poziom pośredni	8	16	16	11	8
Napisz help on-line	12				



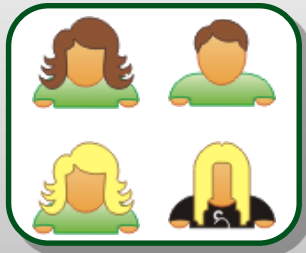
Skalowalność

- Typowy zespół to 7 ± 2 osób
 - Skalowalność wynika z zespołów zespołów
- Wskaźniki w skalowaniu
 - Typ aplikacji
 - Rozmiar zespołu
 - Rozproszenie zespołu
 - Czas trwania projektu
- Scrum jest zwykle używany w projektach wymagających więcej niż 500 osób.

Skalowanie poprzez Scrum Scrumów



Scrum scrumów, scrumów



Gdzie można znaleźć więcej

- www.mountangoatsoftware.com/scrum
- www.scrumalliance.org
- www.controlchaos.com
- scrumdevelopment@yahogroups.com

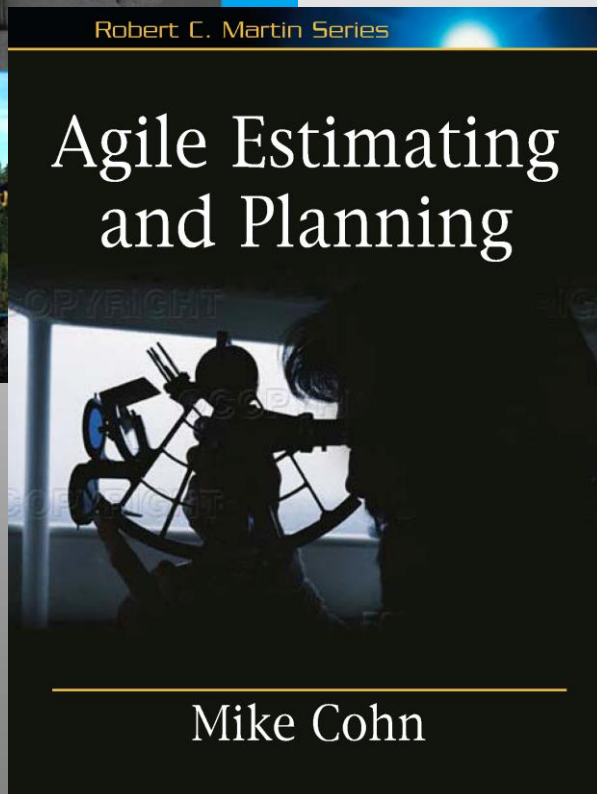
Literatura

- *Agile and Iterative Development: A Manager's Guide* by Craig Larman
- *Agile Estimating and Planning* by Mike Cohn
- *Agile Project Management with Scrum* by Ken Schwaber
- *Agile Retrospectives* by Esther Derby and Diana Larsen

Literatura

- *Agile Software Development Ecosystems* by Jim Highsmith
- *Agile Software Development with Scrum* by Ken Schwaber and Mike Beedle
- *Scrum and The Enterprise* by Ken Schwaber
- *Succeeding with Agile* by Mike Cohn
- *User Stories Applied for Agile Software Development* by Mike Cohn

Informacje kontaktowe



Mike Cohn

mike@mountaingoatsoftware.com

www.mountaingoatsoftware.com

(720) 890-6110 (office)