

# Programowanie zespołowe

## Laboratorium 5 - scrum cz. 1

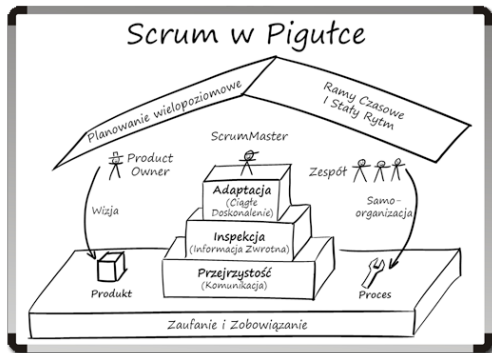
mgr inż. Krzysztof Szwarc

[krzysztof@szwarc.net.pl](mailto:krzysztof@szwarc.net.pl)

Sosnowiec, 21 marca 2017

# Filary scruma

- Przejrzystość (widoczny jest rzeczywisty stan procesu).
- Inspekcja (analiza procesu wytwarzania).
- Adaptacja (dostosowanie procesu do potrzeb).

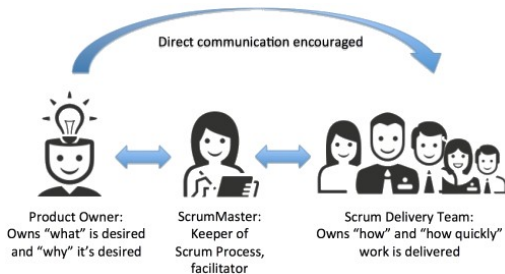


<http://procognita.pl>

- 1 Role.
- 2 Narzędzia (Rejestr Produktowy i Sprintu).
- 3 Zdarzenia (spotkania i sprinty; zapewniają ramy czasowe projektu i ułatwiają komunikację).
- 4 Reguły (łączą pozostałe elementy).

# Główne role w zespole scrumowym

- 1 Właściciel produktu (ang. Product Owner) - reprezentuje klienta.
- 2 Scrum Master - coach scrumowy. Współpracuje z zespołem, właścicielem produktu i całą firmą.
- 3 Zespół deweloperski - odpowiada za dostarczenie produktu.

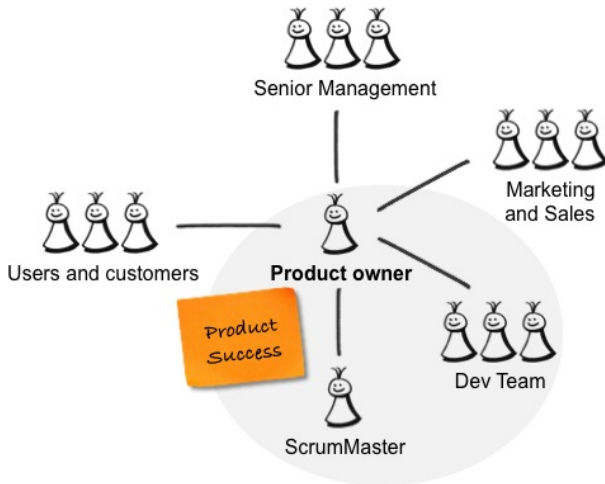


<https://www.safaribooksonline.com>

# Właściciel produktu

- Posiada wizję tworzonego produktu, którą zmienia na uporządkowaną listę oczekiwań, tj. **Rejestr Produktowy** (ang. Product Backlog).
- Koncentruje się na maksymalizacji zysku z inwestycji (ROI) - posiada wiedzę dziedzinową i stale analizuje rynek (zajmuje się stroną biznesową).
- Pozyskuje informacje zwrotne o produkcie i wykorzystuje je do ustalenia kolejności zadań w **Rejestrze Produktowym**.
- Współpracuje z zespołem i interesariuszami.
- Akceptuje lub odrzuca efekty pracy.

# Miejsce właściciela produktu

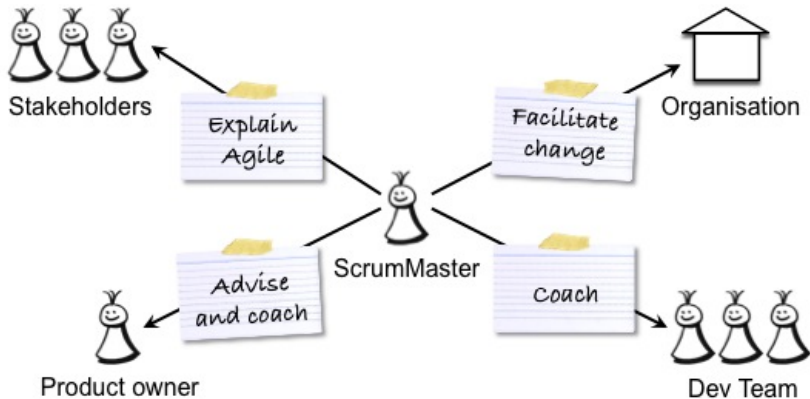


<http://www.romanpichler.com/>

# Scrum Master

- Odpowiada za poprawne wdrożenie oraz promowanie podejścia zwinnego.
- Rozumie i tłumaczy zasady scruma.
- Wspiera i szkoli zespół.
- Upewnia się, że następuje efektywna współpraca oraz wszyscy uczestnicy rozumieją cel i wiedzą jak go osiągnąć.
- Poprzez pytania pomaga zespołowi odnaleźć rozwiązanie (nie narzuca własnego).
- Chroni zespół i usuwa problemy (np. organizuje potrzebne narzędzia).
- Pomaga w ciągłym doskonaleniu identyfikując przyczyny problemów i możliwe rozwiązania.
- Prowadzi spotkania.
- Przywódca służebny.

# Miejsce Scrum Mastera



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



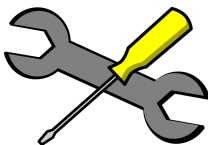
# Zespół deweloperski

- Jest wielofunkcyjny - członkowie posiadają wszystkie potrzebne umiejętności i wiedzę (w zakresie analizy, testów, kodowania i różnych elementów systemu, które należy zmienić, aby dostarczyć przyrost produktu).
- Najczęściej składa się z 3-9 osób (dla większych projektów zaleca się stworzenie wielu wielofunkcyjnych zespołów).
- Jest samoorganizujący (członkowie sami dobierają niezbędne metody, narzędzia i procesy).
- Członkowie nie mają formalnych ról i są przypisani do jednego zespołu.
- Jest przypisany do jednego **Rejestru Produktowego**.

- Członkowie stale współpracują i komunikują się ze sobą, właścicielem produktu, Scrum Masterem oraz interesariuszami.
- Rozwiązuje problemy. Jeżeli jego członkowie nie potrafią sami rozwiązać problemu, zwracają się do Scrum Mastera i właściciela produktu.
- Pomaga właścicielowi produktu przygotować wymagania do kolejnych sprintów oraz zbiera od niego i klientów informację zwrotną.

# Narzędzia (artefakty)

- 1 Rejestr Produktowy (ang. Product Backlog) – uszeregowana lista oczekiwań.
- 2 Rejestr Sprintu (ang. Sprint Backlog) - elementy wybrane z Rejestru Produktowego do sprintu.
- 3 Przyrost Produktu (ang. Product Increment) - efekt kolejnego sprintu.



**Rejestr Produktowy** to uszeregowana lista wymagań (Elementy o wyższym priorytecie znajdują się wyżej).

- Właściciel produktu jest odpowiedzialny za zarządzanie nim.
- Często Elementami Rejestru (ang. Product Backlog Items) są Historyjki Użytkownika (ang. User Stories) lub przypadki użycia (ang. Use Cases).
- Jest on ogólnodostępny (jako jedyne źródło informacji na temat zadań zespołu deweloperskiego).
- Zawiera wszystkie rzeczy wymagane do zrobienia.
- Każdy Element zawiera trzy atrybuty: opis, priorytet (z punktu widzenia biznesowego) oraz koszt realizacji. Na priorytet wpływa ryzyko, wartość oraz konieczność realizacji.

# Przykład Rejestru Produktowego

ToDo List		
Story	Estimation	Priority
As a user I want to be able to reset my password	1	1
As a user I want to edit items	3	2
As a user I want to export data	2	3
As an administrator I want to define KPI's for my sales team	4	4
As a user I want to view my data on mobile	5	5
As an administrator I want to send alerts when new leads come in	2	6
As a user I want to create a report of my data	5	7
As a user I want to update my reminder settings when a date is added	3	8
As a user I want filtering enhancements	4	9
As an administrator I want to configure views of data	5	10
<b>Total</b>	<b>34</b>	

<https://d2myx53yhj7u4b.cloudfront.net/>

# Przykładowy proces tworzenia Rejestru Produktowego

- Na początku procesu właściciel produktu znajduje tylko główne, najważniejsze historyjki użytkownika i opierając się na celach biznesowych stara się je prioryteżować (z biznesowego punktu widzenia). W dalszych etapach dokument aktualizowany jest przez niego oraz zespół deweloperski.
- Utworzone historyjki są konsultowane z zespołem, który stara się dokonać szacowania za pomocą tzw. Story Points (uniezależniona od czasu wykonania uniwersalna miara złożoności danego zadania). Zakłada się, że im niższa wartość tym łatwiejsze zadanie. Zazwyczaj Story Points przybierają wartości z następującego zbioru:  $< 0, \frac{1}{2}, 1, 2, 3, 5, 8, 13, 20, 40, 100 >$ .

# Prędkość zespołu

## Wydajność

Liczba ukończonych story pointów w jednostce czasu (np. w sprincie) to wydajność zespołu określana mianem prędkości zespołu (ang. velocity). Na jej podstawie możemy szacować czas zakończenia prac.



<https://www.countersoft.com>

## Fazy pokera

- Moderator (zwykle Właściciel Produktu) czyta opis Historyjki i odpowiada na ewentualne pytania.
- Wszyscy jednocześnie wyciągają kartę z wartością mówiącą, ile Story Points powinna ich zdaniem mieć dana Historyjka.
- Jeżeli estymaty się różnią to właściciele najniższej i najwyższej wartości tłumaczą, dlaczego zaproponowali taką estymatę i głosowanie jest powtarzane.



<http://img.freepik.com>



# Od czego zacząć estymację?

Problem z estymacją pierwszych Historyjek rozwiązuje się poprzez znalezienie prostej Historyjki i przypisanie jej 1 SP, albo Historyjki o średniej złożoności uznać za 5 SP. Resztę Historyjek szacuje się relatywnie do wyestymowanych.

## Historyjki użytkownika

**Historyjki użytkownika (ang. User Stories)** to jednozdaniowe oczekiwania względem projektu, określone z punktu widzenia użytkowników. Piszemy je w konwencji:

Jako (kto) chcę (czego oczekuję), **by (w jakim celu)**.

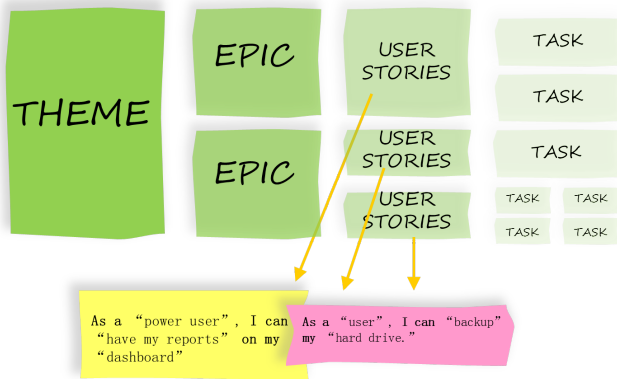
Przykład: Jako zwykły użytkownik chcę powiększyć zdjęcie, by lepiej mu się przyjrzeć.

# Akronim INVEST dla Historyjek użytkownika

- **I**NDEPENDENT (niezależna) - historyjka nie powinna zależeć od innych.
- **N**EGOTIABLE (negocjowalna) - powinna istnieć możliwość delikatnej modyfikacji.
- **V**ALUABLE (cenna) - musi dostarczyć wartość interesariuszom.
- **E**STIMABLE (dająca się oszacować) - musi istnieć możliwość oszacowania rozmiaru historyjki.
- **S**MALL (niewielka) - powinna być możliwa do zaplanowania, wykonania i priorytetyzacji.
- **T**ESTABLE (testowalna) - musi umożliwić przetestowanie dodanej funkcji.

# Dekompozycja Historyjek użytkownika

## USER STORIES



[www.agile-scrum.be](http://www.agile-scrum.be)

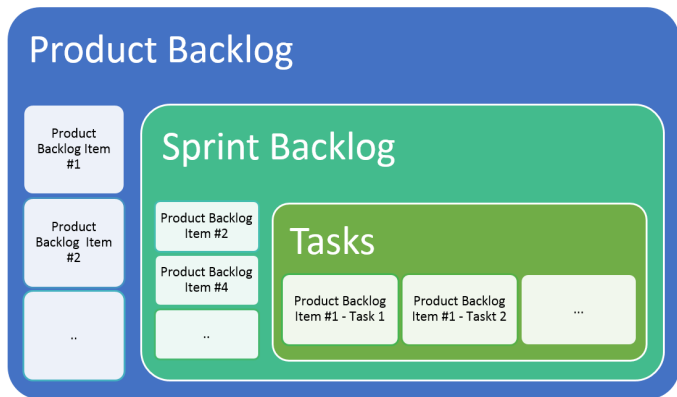
<http://www.agile-scrum.be>

# Technika priorytetyzacji MoSCoW

- **MUST** - wymaganie, które musi być spełnione w końcowym, finalnym rozwiązaniu.
- **SHOULD** - wymaganie o wysokim priorytecie, które powinno być zawarte w rozwiązaniu, jeżeli jest to możliwe.
- **COULD** - wymaganie, które jest pożądane, ale może go zabraknąć, tj. zostanie dodane, jeżeli pozwoli na to czas i zasoby.
- **WON'T** - wymaganie, które nie zostanie dodane w aktualnej wersji tworzonego oprogramowania (za zgodą interesariuszy), ale może zostać zaimplementowane w przyszłości.

# Rejestr Sprintu

**Rejestr Sprintu** jest listą Elementów Rejestru wybranych z Rejestru Produktowego do wykonania podczas danego sprintu.



<https://www.scrumalliance.org/>

# Przykładowy Rejestr Sprintu

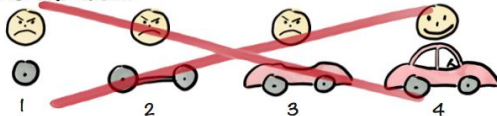
User Story	Tasks	Day 1	Day 2	Day 3	Day 4	Day 5	...
As a member, I can read profiles of other members so that I can find someone to date.	Code the ...	8	4	8	0		
	Design the ...	16	12	10	4		
	Meet with Mary about ...	8	16	16	11		
	Design the UI	12	6	0	0		
	Automate tests ...	4	4	1	0		
	Code the other ...	8	8	8	8		
As a member, I can update my billing information.	Update security tests	6	6	4	0		
	Design a solution to ...	12	6	0	0		
	Write test plan	8	8	4	0		
	Automate tests ...	12	12	10	6		
	Code the ...	8	8	8	4		

<https://www.mountaingoatsoftware.com>

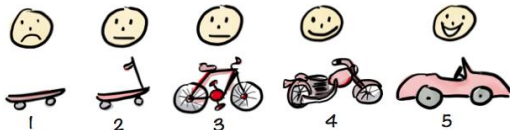
# Przyrost Produktu

**Przyrost Produktu** jest wynikiem prac dostarczanych na koniec sprintu, tj. **działający** produkt rozszerzony o nowe funkcje w stosunku do wersji z poprzedniej iteracji. Powinien być pozbawiony długu technologicznego.

Not like this....



Like this!



<http://www.enhancedscrumguidedotcom.files.wordpress.com/>



## Definition of Done

**Definition of Done** (DoD) to lista wymagań określająca stan wykonania zadania. Kiedy zespół deweloperski stwierdza, że aktywność została ukończona oznacza to, że spełnia ona wszystkie wymagania określone w DoD.

- Powinny być określone przed estymacją.
- Są ogólnodostępne.
- Muszą być zrozumiałe i zaakceptowane.
- Określają jakość produktu.

# Przykładowe Definition of Done

- Testy jednostkowe pokrywają 40% kodu.
- Pomyślnie przeprowadzono testy manualne.
- Przeprowadzono inspekcję kodu.
- Zaktualizowano dokumentację.

## Kryteria akceptacji

**Kryteria akceptacji** (ang. Acceptance Criteria) określają jakie założenia muszą zostać spełnione przez rozwiązanie, aby mogło ono zostać uznane, przez zamawiającego, jako ukończone. Dotyczy konkretnych zadań (historyjek użytkownika) i dzieli się na funkcjonalne (użyteczne z punktu widzenia użytkownika) oraz нефункционалне (dotyczy szczegółów technicznych, np. „System musi być dostępny poprzez przeglądarkę internetową zgodną z ACID 3 na poziomie 100/100”).

# Przykładowe kryteria akceptacji dla rejestracji

- Użytkownik nie może zakończyć procesu rejestracji bez wypełnienia wszystkich pól obowiązkowych w formularzu (oznaczonych gwiazdką).
- Wszystkie wprowadzone przez użytkownika informacje, w formularzu rejestracyjnym, będą przechowywane w SZBD MySQL.

- Przygotujcie w zespole DoD dla zadania projektowego.

- Wytypujcie w zespole Właściciela produktu.

- Jestem klientem - porozmawiajmy o ramach projektu.

- Stwórzmy historyjki użytkownika.



- Napiszcie kryteria akceptacji dla historyjek użytkownika.

- Dokonajcie priorytetyzacji historyjek z wykorzystaniem MoSCoW.

# Zadania

- Przypiszmy Story Points do historyjek z wykorzystaniem planistycznego pokera (play.planningpoker.com).



<https://pbs.twimg.com/>

Dziękuję za uwagę