

# Metodyki zwinne wytwarzania oprogramowania

## Wykład 1

Marcin Młotkowski

5 października 2016

# Plan wykładu

- 1 Sprawy organizacyjne
  - Organizacja pracowni
- 2 Wstęp
  - Wprowadzenie
  - Zarządzanie projektem
- 3 Powstanie metodyk zwinnych

# O wykładzie i wykładowcy

## Prowadzący

Marcin Młotkowski

konsultacje: środa 08:15–10:00, pok. 303

## Strona wykładu

<http://www.ii.uni.wroc.pl/~marcinm/dyd/zwinne/>

## Cel wykładu

Wykład dostarczy wiadomości o zasadach programowania zwinnego, o metodykach oraz ich zaletach i wadach. Wykład obejmuje również zasady projektowania obiektowego a także teorię i praktykę testowania. Zadaniem pracowni jest nabycie pewnej praktyki w stosowaniu zasad zwinnych, w szczególności pracy zespołowej.

# Plan wykładu

- Ogólne zasady zwinnego wytwarzania oprogramowania
- Planowanie działania
- Współpraca z użytkownikiem: określanie wymagań funkcjonalnych i нефункциональных
- Rola testowania w programowaniu zwinnym
- Wzorce projektowe
- Udoskonalanie kodu

# Literatura I



Alistar Cockburn.

Agile Software Development. Gra zespołowa.

Wydawnictwo Helion, 2008.



Martin Fowler.

Refaktoryzacja.

Wydawnictwa Naukowo–Techniczne, 2006.



Erich Gamma, Richard Helm, Ralph Johnson, and John  
Vilissides.

Wzorce Projektowe.

Wydawnictwa Naukowo–Techniczne, 2005.



Ron Jeffries.

Programowanie ekstremalne w C#.

Microsoft Press, 2005.

## Literatura II



Robert C. Martin and Micah Martin.

Agile. Programowanie zwinne: zasady, wzorce i praktyki zwinnego wytwarzania oprogramowania w C#.

Wydawnictwo Helion, 2008.



Ken Schwaber.

Agile project management with SCRUM.

Microsoft Press, 2004.

# Do poczytania



Frederick Brooks.

The Mythical Man-Month.



Jason Fried and David Heinemeier Hansson.

ReWork: Change the Way You Work Forever.



# Podział na zespoły

- podział na grupy 3-osobowe;
- praca podczas pracowni;

# Plan wykładu

- 1 Sprawy organizacyjne
  - Organizacja pracowni
- 2 Wstęp
  - Wprowadzenie
  - Zarządzanie projektem
- 3 Powstanie metodyk zwinnych

# Pomieszanie pojęć

metodyka, metodologia, zarządzanie projektami

# Słownik

## Projekt

Zorganizowane działanie zmierzające do osiągnięcia zamierzonego celu.

# Słownik

## Projekt

Zorganizowane działanie zmierzające do osiągnięcia zamierzonego celu. Cechy:

- projekt ma założony czas trwania;
- projekt jest unikatowy;
- projekt ma cele;
- projekt potrzebuje zasobów;
- projekt jest obciążony ryzykiem.

# Słownik

## Projekt

Zorganizowane działanie zmierzające do osiągnięcia zamierzonego celu. Cechy:

- projekt ma założony czas trwania;
- projekt jest unikatowy;
- projekt ma cele;
- projekt potrzebuje zasobów;
- projekt jest obciążony ryzykiem.

## Proces

Powtarzalny ciąg czynności, bez zaplanowanego końca. Przykładem może być wyprodukowanie partii towaru na linii produkcyjnej.

## Przykłady projektów

- zaprojektowanie, implementacja i wdrożenie projektu informatycznego;
- kampania marketingowa;
- prace remontowe;
- opracowanie nowego leku;
- ...

## Słownik, cd.

Metodyka zarządzania projektem

Standard zarządzania projektem.



## Słownik, cd.

### Metodyka zarządzania projektem

Standard zarządzania projektem.

### Metodologia

Standard zarządzania projektem opracowany dla specyficznych dziedzin.

Na przykład badanie opinii publicznej w jakiejś sprawie za pomocą ankiet telefonicznych.

## Wniosek

Metodyki zarządzania projektem to nie tylko metodyki w projektach informatycznych!

## Przykłady uniwersalnych metodyk

PMI (Project Management Institute), PRINCE (Projects In a Controlled Environment)

# Najprostsza metodyka: horda Mongolska

## Opis

Terminy gonią, więc zatrudniamy jak najwięcej osób.

# Najprostsza metodyka: horda Mongolska

## Opis

Terminy gonią, więc zatrudniamy jak najwięcej osób.

## Kiedy się sprawdza

Kopanie rowów.

# Najprostsza metodyka: horda Mongolska

## Opis

Terminy gonią, więc zatrudniamy jak najwięcej osób.

## Kiedy się sprawdza

Kopanie rowów.

## Kiedy się nie sprawdza

Kopanie studni.

# Metodyka zarządzania vs metodyka wytwarzania

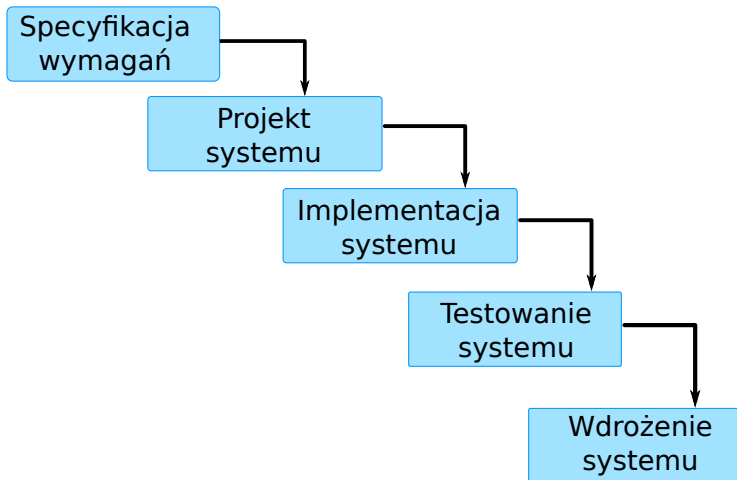
## Metodyka zarządzania

Zarządzanie całym projektem.

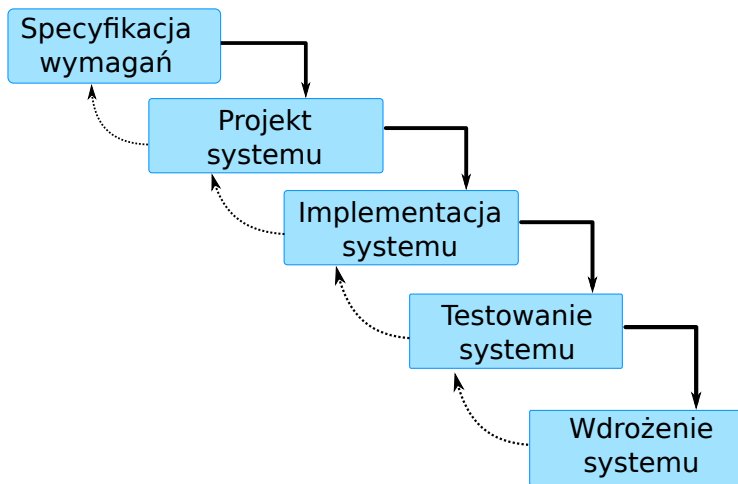
## Metodyka wytwarzania

Zarządzanie samym procesem wytwarzania produktu.

## Przykład metodyki wytwarzania: model kaskadowy



## Kolejne iteracje w modelu kaskadowym





## Niektóre cechy modelu kaskadowego

- nie można przejść do kolejnej fazy przed zakończeniem bieżącej
- iteracje są kosztowne
- nieelastyczny podział
- wymuszenie sporządzenia dokładnej specyfikacji przed rozpoczęciem projektowania i kodowania

# Współpraca z klientem

- klient za bardzo nie wie, czego chce;

# Współpraca z klientem

- klient za bardzo nie wie, czego chce;
- klient nawet wie, czego chce, ale nie zna się na programowaniu;

# Współpraca z klientem

- klient za bardzo nie wie, czego chce;
- klient nawet wie, czego chce, ale nie zna się na programowaniu;
- klient zmienia specyfikację

# Zmiany

Jedyną rzeczą stałą w projekcie są zmiany.

# Zmiany

Jedyną rzeczą stałą w projekcie są zmiany.

## Przyczyny zmian

- klient zmienił wymagania
- klient powiedział dokładnie o co mu chodzi
- zmiana prawa

## Czym się zarządza w projekcie?

# Klasyfikacja Project Management Institute

- zarządzanie integracją projektu;
- zarządzanie zakresem projektu;
- zarządzanie czasem;
- zarządzanie kosztami;
- zarządzanie jakością;
- zarządzanie zasobami ludzkimi;
- zarządzanie komunikacją;
- zarządzanie ryzykiem;
- zarządzanie zaopatrzeniem.



# Zarządzanie integracją projektu

- różne dziedziny (informatyka + ...)
- różni specjaliści
- planowanie projektu (sposób współpracy)
- analiza ekonomiczna projektu.

# Zarządzanie zakresem

- ustalenie celów strategicznych;
- ustalenie kryteriów (czas, budżet, jakość);
- podział na elementy i ich analiza.

# Zarządzanie czasem

- ustalenie listy zadań i ich zależności;
- oszacowanie czasu;
- kontrola terminowości.

# Zarządzanie kosztami

- koszty pracy;
- koszty powierzchni biurowej;
- koszty szkoleń, licencji etc;
- kontrola kosztów.

# Zarządzanie jakością

- kryteria jakości (produktu i procesu);
- wdrożenie kontroli jakości;
- analiza błędów

# Zarządzanie zasobami ludzkimi

- wiedza o umiejętnościach osób, brakach;
- rozwijanie wiedzy zespołu;
- stosunki międzyludzkie.

# Zarządzanie komunikacją

- standardy wymiany informacji (wymagania, kody źródłowe etc);
- kontrola wykonywanych zadań;
- archiwizacja.

# Zarządzanie ryzykiem

- identyfikacja zagrożeń (włamanie, awaria dysku);
- usuwanie skutków zagrożeń;



# Zarządzanie zaopatrzeniem

- zaopatrzenie w materiały (np. specjalistyczny sprzęt);
- zaopatrzenie w oprogramowanie i licencje, instalacja

# Plan wykładu

- 1 Sprawy organizacyjne
  - Organizacja pracowni
- 2 Wstęp
  - Wprowadzenie
  - Zarządzanie projektem
- 3 Powstanie metodyk zwinnych

# Inspiracje

- duża liczba projektów zakończona niepowodzeniem

# Inspiracje

- duża liczba projektów zakończona niepowodzeniem
- wady stosowanych metodyk

## Sposób prowadzenia projektu: kaskada vs zwinne

Szczegółowe procedury i zaawansowane narzędzia do zarządzania projektem.

## Sposób prowadzenia projektu: kaskada vs zwinne

Szczegółowe procedury i zaawansowane narzędzia do zarządzania projektem.

Postawienie na współpracę i zaufanie

# Specyfikacja projektu: kaskada vs zwinne

Wyczerpująca dokumentacja

## Specyfikacja projektu: kaskada vs zwinne

Wyczerpująca dokumentacja

Działające oprogramowanie



# Współpraca z klientem: kaskada vs zwinne

Negocjacje kontraktu

# Współpraca z klientem: kaskada vs zwinne

Negocjacje kontraktu

Stała współpraca z klientem

# Organizacja projektu: kaskada vs zwinne

Trzymanie się planu

# Organizacja projektu: kaskada vs zwinne

Trzymanie się planu

Reagowanie na zmiany

## Rezultat tych zmian

W lutym 2001, w Snowbird (Utah)  
spotkało się 17 deweloperów.

## Rezultat tych zmian

W lutym 2001, w Snowbird (Utah)

spotkało się 17 deweloperów.

I napisało Agile Manifesto

# Agile Manifesto

Poprzez wytwarzanie oprogramowania oraz pomaganie innym w tym zakresie odkrywamy lepsze sposoby realizowania tej pracy. W wyniku tych doświadczeń zaczęliśmy przedkładać:

**Ludzi i ich wzajemne interakcje (współdziałanie)** ponad procedury i narzędzia.

**Działające oprogramowanie** nad wyczerpującą dokumentacją.

**Współpracę z klientem** nad negocjacją umów.

**Reagowanie na zmiany** nad realizowanie planu.

Oznacza to, że wprawdzie doceniamy to co wymieniono po prawej stronie, to jednak bardziej cenimy to co wymieniono po lewej.

# Zasady programowania zwinnego

- 1 osiągnięcie satysfakcji klienta poprzez szybkość wytwarzania oprogramowania,



## Zasady programowania zwinnego

- 1 osiągnięcie satysfakcji klienta poprzez szybkość wytwarzania oprogramowania,
- 2 działające oprogramowanie jest dostarczane okresowo (raczej tygodniowo niż miesięcznie),

## Zasady programowania zwinnego

- 1 osiągnięcie satysfakcji klienta poprzez szybkość wytwarzania oprogramowania,
- 2 działające oprogramowanie jest dostarczane okresowo (raczej tygodniowo niż miesięcznie),
- 3 podstawową miarą postępu jest działające oprogramowanie,

## Zasady programowania zwinnego

- 1 osiągnięcie satysfakcji klienta poprzez szybkość wytwarzania oprogramowania,
- 2 działające oprogramowanie jest dostarczane okresowo (raczej tygodniowo niż miesięcznie),
- 3 podstawową miarą postępu jest działające oprogramowanie,
- 4 późne zmiany w specyfikacji nie mają destrukcyjnego wpływu na proces wytwarzania oprogramowania,

## Zasady programowania zwinnego

- 1 osiągnięcie satysfakcji klienta poprzez szybkość wytwarzania oprogramowania,
- 2 działające oprogramowanie jest dostarczane okresowo (raczej tygodniowo niż miesięcznie),
- 3 podstawową miarą postępu jest działające oprogramowanie,
- 4 późne zmiany w specyfikacji nie mają destrukcyjnego wpływu na proces wytwarzania oprogramowania,
- 5 bliska, dzienna współpraca pomiędzy biznesem a developerem,

## Zasady programowania zwinnego

- 1 osiągnięcie satysfakcji klienta poprzez szybkość wytwarzania oprogramowania,
- 2 działające oprogramowanie jest dostarczane okresowo (raczej tygodniowo niż miesięcznie),
- 3 podstawową miarą postępu jest działające oprogramowanie,
- 4 późne zmiany w specyfikacji nie mają destrukcyjnego wpływu na proces wytwarzania oprogramowania,
- 5 bliska, dzienna współpraca pomiędzy biznesem a developerem,
- 6 bezpośredni kontakt, jako najlepsza forma komunikacji w zespole i poza nim,

## Zasady programowania zwinnego

- 1 osiągnięcie satysfakcji klienta poprzez szybkość wytwarzania oprogramowania,
- 2 działające oprogramowanie jest dostarczane okresowo (raczej tygodniowo niż miesięcznie),
- 3 podstawową miarą postępu jest działające oprogramowanie,
- 4 późne zmiany w specyfikacji nie mają destrukcyjnego wpływu na proces wytwarzania oprogramowania,
- 5 bliska, dzienna współpraca pomiędzy biznesem a developerem,
- 6 bezpośredni kontakt, jako najlepsza forma komunikacji w zespole i poza nim,
- 7 ciągła uwaga nastawiona na aspekty techniczne oraz dobry projekt (design),

## Zasady programowania zwinnego

- 1 osiągnięcie satysfakcji klienta poprzez szybkość wytwarzania oprogramowania,
- 2 działające oprogramowanie jest dostarczane okresowo (raczej tygodniowo niż miesięcznie),
- 3 podstawową miarą postępu jest działające oprogramowanie,
- 4 późne zmiany w specyfikacji nie mają destrukcyjnego wpływu na proces wytwarzania oprogramowania,
- 5 bliska, dzienna współpraca pomiędzy biznesem a developerem,
- 6 bezpośredni kontakt, jako najlepsza forma komunikacji w zespole i poza nim,
- 7 ciągła uwaga nastawiona na aspekty techniczne oraz dobry projekt (design),
- 8 prostota,

## Zasady programowania zwinnego

- 1 osiągnięcie satysfakcji klienta poprzez szybkość wytwarzania oprogramowania,
- 2 działające oprogramowanie jest dostarczane okresowo (raczej tygodniowo niż miesięcznie),
- 3 podstawową miarą postępu jest działające oprogramowanie,
- 4 późne zmiany w specyfikacji nie mają destrukcyjnego wpływu na proces wytwarzania oprogramowania,
- 5 bliska, dzienna współpraca pomiędzy biznesem a developerem,
- 6 bezpośredni kontakt, jako najlepsza forma komunikacji w zespole i poza nim,
- 7 ciągła uwaga nastawiona na aspekty techniczne oraz dobry projekt (design),
- 8 prostota,
- 9 samzarządzalność zespołów,



## Zasady programowania zwinnego

- 1 osiągnięcie satysfakcji klienta poprzez szybkość wytwarzania oprogramowania,
- 2 działające oprogramowanie jest dostarczane okresowo (raczej tygodniowo niż miesięcznie),
- 3 podstawową miarą postępu jest działające oprogramowanie,
- 4 późne zmiany w specyfikacji nie mają destrukcyjnego wpływu na proces wytwarzania oprogramowania,
- 5 bliska, dzienna współpraca pomiędzy biznesem a developerem,
- 6 bezpośredni kontakt, jako najlepsza forma komunikacji w zespole i poza nim,
- 7 ciągła uwaga nastawiona na aspekty techniczne oraz dobry projekt (design),
- 8 prostota,
- 9 samozarządzalność zespołów,
- 10 regularna adaptacja do zmieniających się wymagań.