

Projektowanie Systemów Informatycznych - wstęp

mgr inż. Wiktor Kuśmirek

Organizacja zajęć

- ▶ Wprowadzenie do projektowania systemów informatycznych (2h)
- ▶ Specyfikacja wymagań systemu informatycznego (2h)
- ▶ Język UML - diagram przypadków użycia (2h)
- ▶ Język UML - diagram sekwencji (2h)
- ▶ Język UML - diagram klas (2h)
- ▶ Język BPMN (2h)
- ▶ Projektowanie bazy danych systemu informatycznego (2h)
- ▶ Projekt i implementacja graficznego interfejsu użytkownika. ASP .NET MVC (2h)

- ▶ Wprowadzenie do projektowania systemów informatycznych (1pkt)
- ▶ Specyfikacja wymagań systemu informatycznego (7pkt)
- ▶ Język UML - diagram przypadków użycia (7pkt)
- ▶ Język UML - diagram sekwencji (7pkt)
- ▶ Język UML - diagram klas (7pkt)
- ▶ Język BPMN (7pkt)
- ▶ Projektowanie bazy danych systemu informatycznego (7pkt)
- ▶ Projekt i implementacja graficznego interfejsu użytkownika. ASP .NET MVC (7pkt)

- ▶ **Krzysztof Sacha. Inżynieria oprogramowania, 2010**
- ▶ Jacek Płodzień. Analiza i projektowanie systemów informatycznych, 2005
- ▶ Bernd Bruegge, Allen H. Dutoit. Inżynieria oprogramowania w ujęciu obiektowym. UML, wzorce projektowe i Java, 2011
- ▶ Stanisław Wrycza, Bartosz Marcinkowski, Krzysztof Wyrzykowski. Język UML 2.0 w modelowaniu systemów informatycznych, 2015
- ▶ Szymon Drejewicz. Zrozumieć BPMN. Modelowanie procesów biznesowych, 2012
- ▶ Campbell Laine. INŻYNIERIA NIEZAWODNYCH BAZ DANYCH. PROJEKTOWANIE SYSTEMÓW ODPORNYCH NA BŁĘDY, 2018

Proces rozwoju systemu informatycznego

- ▶ określenie wymagań
- ▶ wytworzenie systemu
- ▶ wdrożenie
- ▶ eksploatacja i konserwacja

Proces rozwoju systemu informatycznego

- ▶ określenie wymagań
 - ▶ specyfikacja wymagań
 - ▶ studium wykonalności
 - ▶ negocjacje biznesowe
- ▶ wytworzenie systemu
- ▶ wdrożenie
- ▶ eksploatacja i konserwacja

Proces rozwoju systemu informatycznego

- ▶ określenie wymagań
 - ▶ specyfikacja wymagań
 - ▶ studium wykonalności
 - ▶ negocjacje biznesowe
- ▶ wytworzenie systemu
 - ▶ analiza wymagań
 - ▶ projektowanie systemu
 - ▶ opracowanie oprogramowania
 - ▶ przygotowanie infrastruktury
 - ▶ integracja i testowanie
- ▶ wdrożenie
- ▶ eksploatacja i konserwacja

Proces rozwoju systemu informatycznego

- ▶ określenie wymagań
 - ▶ specyfikacja wymagań
 - ▶ studium wykonalności
 - ▶ negocjacje biznesowe
- ▶ wytworzenie systemu
 - ▶ analiza wymagań
 - ▶ projektowanie systemu
 - ▶ opracowanie oprogramowania
 - ▶ przygotowanie infrastruktury
 - ▶ integracja i testowanie
- ▶ wdrożenie
 - ▶ przeniesienie danych
 - ▶ szkolenie użytkowników
 - ▶ zmiana procesu biznesowego
- ▶ eksploatacja i konserwacja

Proces rozwoju systemu informatycznego

- ▶ określenie wymagań
 - ▶ specyfikacja wymagań
 - ▶ studium wykonalności
 - ▶ negocjacje biznesowe
- ▶ wytworzenie systemu
 - ▶ analiza wymagań
 - ▶ projektowanie systemu
 - ▶ opracowanie oprogramowania
 - ▶ przygotowanie infrastruktury
 - ▶ integracja i testowanie
- ▶ wdrożenie
 - ▶ przeniesienie danych
 - ▶ szkolenie użytkowników
 - ▶ zmiana procesu biznesowego
- ▶ eksploatacja i konserwacja
 - ▶ używanie systemu
 - ▶ usuwanie błędów
 - ▶ modyfikacja i ewolucja

Wytwarzanie oprogramowania - rodzaje działań

Rodzaje działań podczas wytwarzania oprogramowania:

- ▶ analiza
- ▶ projektowanie
- ▶ implementacja
- ▶ weryfikacja i zatwierdzenie
- ▶ konserwacja

Wytwarzanie oprogramowania - koszty

Koszty wytwarzania oprogramowania:

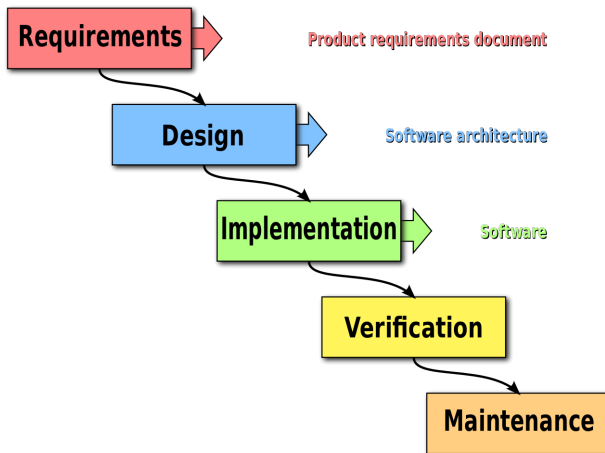
- ▶ analiza - 15%
- ▶ projektowanie - 25%
- ▶ implementacja - 20%
- ▶ weryfikacja i zatwierdzenie - 40%
- ▶ konserwacja - 100% - 300%

Procesy wytwarzania oprogramowania

Istnieje kilka procesów wytwarzania systemów informatycznych. Do wiodących metodologii wytwarzania oprogramowania zaliczamy:

- ▶ kaskadowy
- ▶ iteracyjny

Proces kaskadowy



https://en.wikipedia.org/wiki/Waterfall_model#/media/File:Waterfall_model.svg

Proces kaskadowy

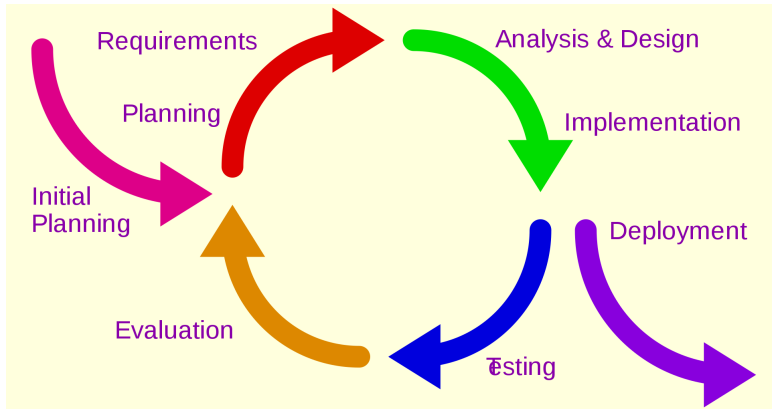
► Zalety:

- brak powtarzania działań
- uporządkowanie pracy
- naturalna weryfikacja wyników każdej fazy

► Wady:

- późna ocena stopnia spełnienia wymagań
- brak miejsca na wprowadzenie zmian
- kosztowne i trudne wprowadzanie zmian i poprawienie niespełnionych wymagań

Proces iteracyjny



https://en.wikipedia.org/wiki/Iterative_and_incremental_development#/media/File:Iterative_development_model.svg

Proces iteracyjny

► Zalety:

- możliwość wczesnego wykrycia i poprawienia błędów
- łatwa zmiana wymagań
- szybkie wykrywanie i poprawianie błędów
- zmniejszenie kosztów projektu
- szybka wersja podstawowego oprogramowania
- łatwość modyfikacji wymagań

► Wady:

- trudne zarządzanie
- skomplikowana implementacja
- krótki horyzont planowania
- praca z niepełną dokumentacją
- proces możliwy do zrealizowania w niewielkim zespole

Weryfikacja i zatwierdzenie

Dwa rodzaje działań testujących przygotowane oprogramowanie:

- ▶ weryfikacja - zgodność oprogramowania ze specyfikacją
- ▶ zatwierdzenie - zgodność oprogramowania z potrzebami użytkownika

Wiodące metody weryfikacji i zatwierdzenia:

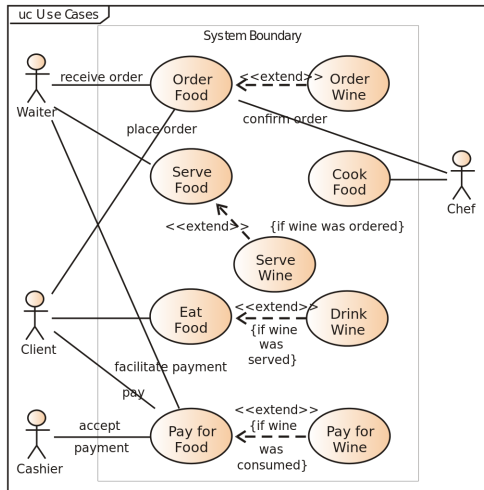
- ▶ testowanie
- ▶ przeglądy i inspekcje

Tematyka zajęć

Specyfikacja wymagań

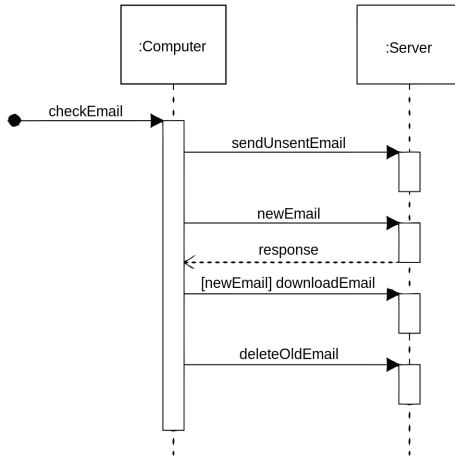
- ▶ opis i cel projektu
- ▶ charakterystykę aktorów, np. manager banku, klient, pracownik itp.
- ▶ wymagania funkcjonalne
- ▶ wymagania niefunkcjonalne
- ▶ wymagania zgodności
- ▶ słownik dziedziny problemu, np. opis pojęcia lokata, dywidenda

Diagram przypadków użycia (UML)



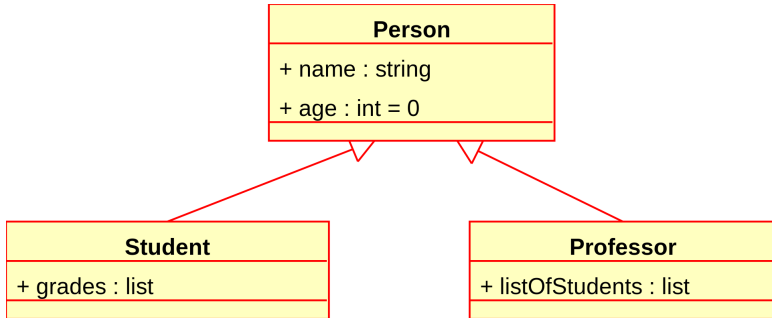
https://en.wikipedia.org/wiki/Use_case_diagram#/media/File:Use_case_restaurant_model.svg

Diagram sekwencji (UML)



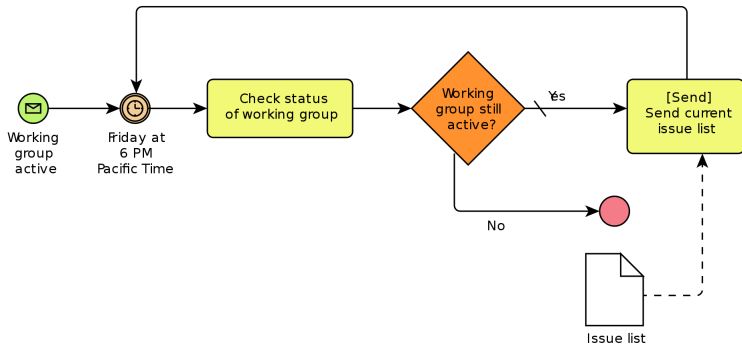
https://en.wikipedia.org/wiki/Sequence_diagram#/media/File:CheckEmail.svg

Diagram klas (UML)



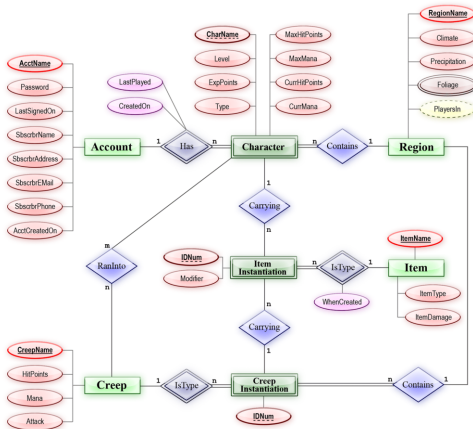
https://en.wikipedia.org/wiki/Class_diagram#/media/File:KP-UML-Generalization-20060325.svg

BPMN - Business Process Model and Notation



https://en.wikipedia.org/wiki/Business_Process_Model_and_Notation#/media/File:BPMN-AProcesswithNormalFlow.svg

Projekt bazy danych systemu informatycznego



https://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model#/media/File:ER_Diagram_MMORPG.png

Projekt i implementacja graficznego interfejsu użytkownika

- ▶ technologia ASP .NET MVC
- ▶ możliwość tylko zaprojektowania interfejsu, bez implementacji
- ▶ interfejs powinien być:
 - ▶ przejrzysty
 - ▶ intuicyjny

- ▶ Etapy wytwarzania oprogramowania:
 - ▶ określenie wymagań
 - ▶ wytworzenie systemu
 - ▶ wdrożenie
 - ▶ eksploatacja i konserwacja
- ▶ Procesy wytwarzania oprogramowania:
 - ▶ kaskadowy
 - ▶ iteracyjny
- ▶ Dwupłaszczyznowe testowanie oprogramowania:
 - ▶ weryfikacja
 - ▶ zatwierdzenie

Dziękuję