**Nazwa przedmiotu:**

Inżynieria wymagań oprogramowania

**Koordynator przedmiotu:**

dr inż. Michał Śmiałek, michal.smialek@ee.pw.edu.pl, +48222345387

**Status przedmiotu:**

Obowiązkowy

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Wspólne

**Kod przedmiotu:**

**Semestr nominalny:**

6 / rok ak. 2009/2010

**Liczba punktów ECTS:**

4

**Liczba godzin pracy studenta związanych z osiągnięciem efektów uczenia się:**

**Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:**

**Język prowadzenia zajęć:**

polski

**Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:**

**Formy zajęć i ich wymiar w semestrze:**

|  |  |
| --- | --- |
| Wykład: | 30h |
| Ćwiczenia: | 0h |
| Laboratorium: | 0h |
| Projekt: | 30h |
| Lekcje komputerowe: | 0h |

**Wymagania wstępne:**

Inżynieria oprogramowania, Modelowanie oprogramowania w języku UML

**Limit liczby studentów:**

**Cel przedmiotu:**

Umiejętność zespołowego tworzenie specyfikacji wymagań użytkownika oraz wymagań oprogramowania

**Treści kształcenia:**

Wykład
1. Proces inżynierii wymagań a wymagania. Problemy inżynierii oprogramowania. Proces kaskadowy oraz iteracyjny. Metodyka dla inżynierii oprogramowania – metodyki formalne i agilne. Sześć najlepszych praktyk inżynierii oprogramowania. Podstawowe role w procesie wytwórczym dla oprogramowania. Rola wymagań w procesie wytwórczym. Wymagania sterujące procesem wytwórczym.
2. Specyfikowanie wymagań użytkownika. Modele wymagań a dokumentacja wymagań. Struktura modeli i dokumentów wymagań. Wizja systemu. Opis procesów biznesowych. Określanie zakresu systemu oprogramowania. Typy wymagań. Atrybuty wymagań. Techniki zbierania wymagań.
3. Modelowanie przypadków użytkownika i słownika. Definicja przypadku użycia i aktora. Omówienie modelu przypadków użycia. Relacje między przypadkami użycia. Punkty rozszerzenia. Zawartość przypadków użycia. Konstrukcja słownika. Pojęcia słownikowe w opisach przypadków użycia.
4. Modelowanie biznesowe dla systemów oprogramowania. Wpływ biznesu na system oprogramowania. Wizja
systemu oparta na biznesie. Biznesowe przypadki użycia – porównanie z systemowymi przypadkami użycia. Czynności – opisy przypadków użycia. Diagramy czynności. Transformacja z modelu biznesu do modelu
wymagań.
5. Struktura i proces dla wymagań oprogramowania. Specyfikacja wymagań oprogramowania a proces
iteracyjny. Kryteria nadawania priorytetów dla przypadków użycia. Wymagania oprogramowania a testy
akceptacyjne systemu. Wymagania oprogramowania a dokumentacja dla użytkownika.
6. Modelowanie statyczne wymagań oprogramowania. Model klas. Modelowanie klas na poziomie wymagań
oprogramowania. Mapowanie klas z pojęć słownikowych.
7. Modelowanie dynamiczne wymagań oprogramowania. Model scenariuszy. Zdania scenariuszy zapisywane
w sposób formalny. Scenariusze a diagramy czynności. Scenariusze pochodzące od przypadków użycia.
8. Organizacja i jakość wymagań oprogramowania. Typy wymagań pozafunkcjonalnych. Cechy dobrej
specyfikacji wymagań oprogramowania. Realizacja wymagań. Techniki zbierania wymagań. Projekt
Stworzenie pełnej specyfikacji wymagań użytkownika oraz wymagań oprogramowania. Na projekt składają się: specyfikacja procesów biznesowych, specyfikacja przypadków użycia, specyfikacja słownika (modelu klas), specyfikacja scenariuszy przypadków użycia. Projekt wykonywany jest w zespołach po ok. 3 osoby. Każdy zespół tworzy specyfikację wymagań dla niedużego systemu oprogramowania (ok. 15-20 przypadków użycia).

**Metody oceny:**

o

**Egzamin:**

**Literatura:**

1. Michał Śmiałek – Zrozumieć UML 2.0. Metody modelowania obiektowego, Helion, 2005

**Witryna www przedmiotu:**

**Uwagi:**

## Efekty przedmiotowe