**Nazwa przedmiotu:**

Wytwarzanie oprogramowania sterowane modelami

**Koordynator przedmiotu:**

dr hab. inż. Michał Śmiałek, smialek@iem.pw.edu.pl, tel. +48222347370

**Status przedmiotu:**

Obowiązkowy

**Poziom kształcenia:**

Studia II stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Wspólne

**Kod przedmiotu:**

**Semestr nominalny:**

1 / rok ak. 2009/2010

**Liczba punktów ECTS:**

3

**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: | 30h |
| Projekt: | 0h |
| Lekcje komputerowe: | 0h |

**Wymagania wstępne:**

Modelowanie oprogramowania w języku UML, Podstawy inżynierii oprogramowania, Modelowanie procesów biznesowych

**Limit liczby studentów:**

**Cel przedmiotu:**

umiejętność definiowania języków modelowania, umiejętność transformacji modeli, umiejętność organizacji budowy oprogramowania w oparciu o modele

**Treści kształcenia:**

1. Modelowanie oprogramowania. Podstawy modelowania struktury i dynamiki oprogramowania.
(2h)
2. Narzędzia CASE. Najważniejsze cechy narzędzi wspomagających tworzenie oprogramowania. Tworzenie modeli, wspomaganie pracy grupowej, generowanie kodu, generowanie raportów. (4h)
3. Proces wytwarzania oprogramowania sterowany modelami. Etapy modelowania oprogramowania. Rodzaje modeli tworzonych w ramach procesu. (4h)
4. Zasady budowy języków modelowania. Definiowanie składni abstrakcyjnej. Podstawy metamodelowania. Składania abstrakcyjna a składnia konkretna. Semantyka języków modelowania. (6h)
5. Podstawy transformacji modeli. Cele transformacji modeli. Definiowanie reguł transformacji. Zapisywanie reguł przy pomocy programów w językach 3GL. Przegląd języków transformacji modeli. (6h)
6. Programowanie transformacji. Konstrukcje wybranego języka transformacji modeli. (4h)
7. Narzędzie ReDSeeDS i tworzenie przypadków oprogramowania. Prezentacja systemu transformacji modeli. Modelowanie wymagań, modelowanie architektury, modelowanie projektu szczegółowego, przejście do kodu. (4h)

**Metody oceny:**

Za wykład: 55 pkt., za laboratorium: 45 pkt. Zaliczenie na podstawie sumy punktów (od 51 pkt. ocena 3,0; od 61 pkt. ocena 3,5 itd.). Należy zaliczyć obydwie części (uzyskać ponad połowę punktów).

**Egzamin:**

**Literatura:**

1. Michał Śmiałek – Zrozumieć UML 2.0. Metody modelowania obiektowego, Helion, 2005
2. Stephen Mellor, Kendall Scott – MDA Distilled, Addison Wesley, 2004
2. Ian Sommerville – Inżynieria oprogramowania, WNT, 2003
3. Roger S. Pressman – Praktyczne podejście do inżynierii oprogramowania, WNT, 2004

**Witryna www przedmiotu:**

**Uwagi:**

## Efekty przedmiotowe