

Analiza i projektowanie obiektowe w UML - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Analiza i projektowanie obiektowe w UML
Kod przedmiotu	11.3-WK-II-EP-UML-P-S14_pNadGenYQ349
Wydział	Wydział Matematyki, Informatyki i Ekonometrii
Kierunek	Informatyka i ekonometria
Profil	ogólnoakademicki
Rodzaj studiów	pierwszego stopnia z tyt. licencjata
Semestr rozpoczęcia	semestr zimowy 2016/2017

Informacje o przedmiocie	
Semestr	6
Liczba punktów ECTS do zdobycia	5
Typ przedmiotu	obieralny
Język nauczania	polski
Sylabus opracował	dr hab. inż. Silva Robak, prof. UZ

Formy zajęć					
Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Projekt	30	2	-	-	Zaliczenie na ocenę
Wykład	30	2	-	-	Egzamin

Cel przedmiotu

Celem przedmiotu jest nauczanie studenta podstaw obiektowej analizy i obiektowego projektowania oprogramowania z wykorzystaniem podstawowych elementów standardowego języka modelowania UML; definiowania wymagań oraz podstaw projektowania architektury oprogramowania systemu.

Przedmiot koncentruje się na fazie analizy i projektowania oprogramowania, natomiast nie obejmuje kwestii programistycznych i implementacyjnych (i nie jest powiązany z konkretnym językiem programowania).

Wymagania wstępne

Znajomość technologii informacyjnych.

Zakres tematyczny

Wykład/projekt

1. Modele systemu informatycznego. Kroki w procesie rozwiązywania problemu (analizy). Problem złożoności systemów oprogramowania.
2. Analiza i projektowanie obiektowe. Cykl tworzenia oprogramowania wg obiektowej metodyki (Rational) Unified Process (RUP) – etapy i iteracje.
3. Metodyka RUP – przepływy czynności; artefakty - zbiory wymagań, projektowy, implementacyjny i wdrożeniowy; modele.
4. Analiza obiektowa - analiza zdarzeń.
5. Zunifikowany język modelowania UML – wprowadzenie: model, elementy modelu, podstawowe diagramy.
6. Zunifikowany język modelowania UML - przypadki użycia i diagramy sekwencji.
7. Modele dziedziny i kontrakty operacji systemowych.
8. Wprowadzenie do obiektowego projektowania oprogramowania.
9. Projektowanie systemów – architektura trójwarstwowa; wzorzec projektowy singleton i obserwator (MVC).
10. Projektowanie obiektowe – diagramy interakcji UML.
11. Projektowanie obiektowe – diagram klas UML.
12. Zasady projektowania interfejsu użytkownika.
13. Przegląd diagramów języka UML 2.0.

Metody kształcenia

Tradycyjny wykład, którego elementy objaśniane są na jednym spójnym i sukcesywnie doskonalonym przykładzie (system dziekanat); ćwiczenia projektowe (modelowanie w UML).

Efekty uczenia się i metody weryfikacji osiągania efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Student zna i rozumie zasady obiektowości.	K_W11	kolokwium	Wykład
Student ma podstawową wiedzę w zakresie analizy obiektowej i projektowania obiektowego oraz różnic pomiędzy nimi.	K_W11	kolokwium	Projekt

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Student posługuje się pojęciem metodyki rozwoju oprogramowania; zna cykl tworzenia oprogramowania wg RUP.	• K_W09	• kolokwium	• Wykład
Student zna podstawy języka modelowania UML.	• K_W10	• kolokwium	• Wykład • Projekt
Student potrafi zaprojektować modele statyki i dynamiki systemu z wykorzystaniem podstaw języka UML.	• K_U28 • K_K03	• projekt	• Projekt
Student potrafi projektować oprogramowanie wg zasad architektury trójwarstwowej.	• K_U27 • K_K09	• projekt	• Projekt

Warunki zaliczenia

- Kolokwium pisemne z progami punktowymi pozwalającymi ocenić, czy student osiągnął efekty kształcenia w stopniu zadawalającym lub zaliczenie w formie ustnej składające się z pytań teoretycznych i problemowych.
- Ocena jednego projektu w semestrze.

Na ocenę z przedmiotu składa się ocena z projektu (50%) oraz ocena z wykładu (50%). Warunkiem zaliczenia przedmiotu jest pozytywna ocena z projektu i wykładu.

Literatura podstawowa

1. S. Wrycza, B. Marcinkowski, K. Wyrzykowski, Język UML w modelowaniu systemów informatycznych, Helion, Gliwice 2005.
2. G. Booch, J. Rumbaugh, I. Jacobson, UML przewodnik użytkownika, Wydawnictwa Naukowo-Techniczne, 2002.
3. Sommerville, Inżynieria oprogramowania, Wydawnictwa Naukowo-Techniczne, Warszawa 2003.

Literatura uzupełniająca

1. E. Gamma, R. Helm, R. Johnson, J. Vlissides, Wzorce projektowe. Elementy oprogramowania obiektowego wielokrotnego użytku, Wydawnictwa Naukowo-Techniczne, Warszawa, 2005.
2. K. Subieta, Słownik terminów z zakresu obiektowości, Akademicka Oficyna Wydawnicza PLJ, Warszawa 1999.
3. R.V. Stumpf, L.C. Teague: Object-Oriented Systems Analysis and Design with UML, Prentice Hall, London, 2005.

Uwagi

Zmodyfikowane przez dr inż. Janusz Jabłoński (ostatnia modyfikacja: 03-10-2016 11:47)

Wygenerowano automatycznie z systemu SylabUZ