

Object-oriented Analysis and Programming in UML - course description

General information	
Course name	Object-oriented Analysis and Programming in UML
Course ID	11.3-WK-IIeP-UML-P-S14_pNadGenYQ349
Faculty	Faculty of Mathematics, Computer Science and Econometrics
Field of study	Informatics and Econometrics
Education profile	academic
Level of studies	First-cycle studies leading to Bachelor's degree
Beginning semester	winter term 2018/2019

Course information	
Semester	6
ECTS credits to win	5
Course type	optional
Teaching language	polish
Author of syllabus	dr hab. inż. Silva Robak, prof. UZ

Classes forms					
The class form	Hours per semester (full-time)	Hours per week (full-time)	Hours per semester (part-time)	Hours per week (part-time)	Form of assignment
Project	30	2	-	-	Credit with grade
Lecture	30	2	-	-	Exam

Aim of the course

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).

Prerequisites

Znajomość technologii informacyjnych.

Scope

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.

Teaching methods

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

Learning outcomes and methods of theirs verification

Outcome description	Outcome symbols	Methods of verification	The class form
Student posługuje się pojęciem metodyki rozwoju oprogramowania; zna cykl tworzenia oprogramowania wg RUP.	K_W09	an evaluation test	Lecture
Student zna i rozumie zasady obiektowości.	K_W11	an evaluation test	Lecture

Outcome description	Outcome symbols	Methods of verification	The class form
Student potrafi projektować oprogramowanie wg zasad architektury trójwarstwowej.	• K_U27 • K_K09	• a project	• Project
Student potrafi zaprojektować modele statyki i dynamiki systemu z wykorzystaniem podstaw języka UML.	• K_U28 • K_K03	• a project	• Project
Student zna podstawy języka modelowania UML.	• K_W10	• an evaluation test	• Lecture • Project
Student ma podstawową wiedzę w zakresie analizy obiektowej i projektowania obiektowego oraz różnic pomiędzy nimi.	• K_W11	• an evaluation test	• Project

Assignment conditions

- 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.

Recommended reading

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.

Further reading

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.

Notes

Modified by dr Robert Dylewski, prof. UZ (last modification: 04-05-2018 19:24)

Generated automatically from SylabUZ computer system