

Inżynieria oprogramowania - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Inżynieria oprogramowania
Kod przedmiotu	11.3-WK-liED-IO-W-S14_pNadGenRB066
Wydział	Wydział Matematyki, Informatyki i Ekonometrii
Kierunek	Informatyka i ekonometria
Profil	ogólnoakademicki
Rodzaj studiów	drugiego stopnia z tyt. magistra
Semestr rozpoczęcia	semestr zimowy 2017/2018

Informacje o przedmiocie	
Semestr	1
Liczba punktów ECTS do zdobycia	7
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	dr inż. Janusz Jabłoński

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
Wykład	15	1	-	-	Egzamin
Laboratorium	30	2	-	-	Zaliczenie na ocenę

Cel przedmiotu

Zapoznanie studenta z wybranymi modelami technikami i metodami Inżynierii Oprogramowania w realizacji systemów informatycznych.

Wymagania wstępne

Znajomość podstaw koncepcji i metod obiektowych, programowania obiektowego oraz podstawy SQL i baz danych.

Zakres tematyczny

Wykład

1. Projektowanie oprogramowania. (1 godz)
2. Techniki planowania i szacowania kosztów - metoda punktów funkcyjnych. (2 godz)
3. Harmonogramowanie oraz monitorowanie przedsięwzięć informatycznych. (2 godz)
4. Procesy kontroli jakości systemów informatycznych oraz problemy niezawodności oprogramowania. (2 godz)
5. Techniki pracy zespołowej. (2 godz)
6. Weryfikacja, walidacja i testowanie oprogramowania. (2 godz)
7. Inspekcje oprogramowania. Interakcja człowiek – komputer. (2 godz)
8. Metody i techniki notacji w wytwarzaniu systemów informacyjnych oraz zarządzanie przedsięwzięciami informatycznymi. (2 godz)

LABORATORIUM

Wykorzystanie technologii Java oraz diagramów UML w środowisku Eclipse do projektowania i wytwarzania systemów informatycznych. Programowanie zdarzeniowe w Java. Narzędzia wspomagające projekt i realizację interakcji człowiek – komputer w oparciu o interfejs graficzny. J2SE oraz J2EE i technologie serwerowe w realizacji systemów informatycznych. Wykorzystanie możliwości JUnit oraz narzędzi SVN w testowaniu, weryfikacji i walidacji kodu. Organizowanie i zarządzanie pracą zespołu programistów w oparciu na przykładzie PRINCE 2.

Metody kształcenia

Tradycyjny wykład, ćwiczenia laboratoryjne z wykorzystaniem narzędzi informatycznych.

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Student rozróżnia grupy uczestników projektu informatycznego oraz potrafi przydzielać im zadania zgodnie z harmonogramem zadań. Student potrafi analizować, specyfikować i implementować wymagania jak również projektować strukturę i dynamikę systemu informatycznego w oparciu o nowoczesne narzędzia i technologie informatyczne, potrafi wykorzystywać specjalizowane biblioteki, moduły i programy komputerowe w zakresie analizy i przetwarzania danych.	• K_U14 • K_U15 • K_U16	• bieżąca kontrola na zajęciach • projekt • wykonanie sprawozdań laboratoryjnych	• Wykład • Laboratorium

Opis efektu	Symbol e efektów	Metody weryfikacji	Forma zajęć
Student zna podstawowe założenia organizowania pracy zespołowej w projektach informatycznych; rozumie konieczność systematycznego podejścia do organizowania pracy dla projektów o długofalowym charakterze oraz rozumie potrzebę ciągłego podnoszenia swoich kwalifikacji opartej na samokształceniu i śledzeniu literatury informatycznej oraz poznawania nowych technologii i narzędzi	• K_K02 • K_K03	• dyskusja • wykonanie sprawozdań laboratoryjnych	• Wykład • Laboratorium
Student zna w stopniu podstawowym metody, techniki oraz narzędzia modelowania i realizacji w JAVA systemów informatycznych, rozumie zakres i możliwości zastosowań baz danych i wzorców projektowych.	• K_W10 • K_W12	• egzamin - ustny, opisowy, testowy i inne • kolokwium • obserwacje i ocena umiejętności praktycznych studenta	• Wykład • Laboratorium

Warunki zaliczenia

Ostateczna ocena z przedmiotu uwzględnia ocenę z laboratorium (65%) i ocenę z egzaminu (35%), przy założeniu, że student osiągnął wszystkie zakładane efekty kształcenia w stopniu dostatecznym. Warunkiem zaliczenia przedmiotu jest uzyskanie oceny pozytywnej z laboratorium i wykładu.

Literatura podstawowa

1. B. Bruegge, A. H. Dutoit, Inżynieria oprogramowania w ujęciu obiektowym. UML wzorce projektowe i JAVA, Helion, Gliwice, 2011.
2. I. Sommerville, Inżynieria oprogramowania, Klasyka Informatyki, WNT, Warszawa, 2003.
3. I. Graham, Metody obiektowe w teorii i praktyce, WNT, 2004.
4. K. Sacha, Inżynieria Oprogramowania, PWN, Warszawa, 2010.
5. N. Dai, L. Mandel, A. Ryman, Eclipse Web Tools Platform. Tworzenie aplikacji WWW w języku Java, Helion, Gliwice, 2008.

Literatura uzupełniająca

1. D. Minter, L. Linwood, Hibernate od nowicjusza do profesjonalisty, Apress, Warszawa 2007.
2. D. Alur, J. Crupi, D. Malks, J2EE Wzorce projektowe, Helion, Gliwice 2004.
3. C. Horstmann, G. Cornell, JAVA 2. Techniki zaawansowane, Helion, Gliwice 2005.
4. A. Hemrajani, Java. Tworzenie aplikacji sieciowych za pomocą Springa, Hibernate i Eclipse, Helion, Gliwice 2007

Uwagi

Zmodyfikowane przez dr Robert Dylewski, prof. UZ (ostatnia modyfikacja: 09-04-2017 16:21)

Wygenerowano automatycznie z systemu SylabUZ