

Inżynieria oprogramowania - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Inżynieria oprogramowania
Kod przedmiotu	11.3-WI-INFP-IO
Wydział	Wydział Informatyki, Elektrotechniki i Automatyki
Kierunek	Informatyka
Profil	ogólnoakademicki
Rodzaj studiów	pierwszego stopnia z tyt. inżyniera
Semestr rozpoczęcia	semestr zimowy 2022/2023

Informacje o przedmiocie	
Semestr	4
Liczba punktów ECTS do zdobycia	4
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	dr inż. Michał Doligalski

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	18	1,2	Zaliczenie na ocenę
Wykład	30	2	18	1,2	Zaliczenie na ocenę

Cel przedmiotu

- zapoznanie studenta z metodami projektowania, analizy i metodami testowania programów,
- nabycie przez studenta umiejętności specyfikacji wymagań, planowania, dokumentacji projektów informatycznych,
- zapoznanie studenta z narzędziami do zorientowanego obiektowo modelowania i weryfikacji programów.

Wymagania wstępne

Teoretyczne podstawy informatyki, Algorytmy i struktury danych, Programowanie obiektowe.

Zakres tematyczny

Wprowadzenie do inżynierii oprogramowania i inżynierii systemowej. Podstawowe definicje, cykle produkcji oprogramowania. Modele wytwarzania oprogramowania - podejście klasyczne i zwinne. Problematyka projektowania systemów informatycznych. Strategie projektowania systemów informatycznych. Modelowanie procesów biznesowych w projekcie informatycznym.

Modele systemów informatycznych. Przykładowe systemy informatyczne w edukacji, rozrywce, architekturze, przemyśle lotniczym, motoryzacyjnym oraz w medycynie. Projekty informatyczne. Specyfika projektów informatycznych. Plan projektu. Dokument wymagań. Definicja wymagań, Specyfikacja wymagań, Specyfikacja funkcjonalna i programowa. Projektowanie i prototypowanie aplikacji. Wprowadzenie do zagadnień niezawodności systemów informatycznych. Testowanie systemów informatycznych. Wyszukiwanie defektów. Proces testowania.

Systemy komputerowego wspomaganie inżynierii oprogramowania (ang. Computer Aided Software Engineering). Upper i Lower CASE, Warsztaty CASE. Konfiguracja. Konfiguracja systemów informatycznych. Zarządzanie w projekcie informatycznym. Zarządzanie projektem grupowym. Zarządzanie kosztem. Konserwacja i ewolucja systemów informatycznych. Ryzyko w projekcie informatycznym i zarządzanie ryzykiem.

Metody kształcenia

Wykład: wykład konwencjonalny/tradycyjny.

Projekt: praca w grupach.

Efekty uczenia się i metody weryfikacji osiągania efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
rozumie problematykę dystrybucji i konserwacji oprogramowania i posiada umiejętności pracy i komunikacji w zespole programistycznym	- K_W09 - K_K04	sprawdzian z progami punktowymi	- Wykład - Projekt
potrafi zdefiniować i scharakteryzować podstawowe cykle wytwarzania oprogramowania	- K_W09 - K_U17 - K_K04 - K_K06	- obserwacje i ocena umiejętności praktycznych studenta - projekt	- Wykład - Projekt

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
potrafi opracować plan projektu, dokumentację wymagań, specyfikacje wymagań oraz specyfikacje funkcjonalną i programową, a także ocenić jakość projektu z wykorzystaniem odpowiednich narzędzi	• K_W09 • K_U16 • K_U18 • K_K04 • K_K06	• obserwacje i ocena umiejętności praktycznych studenta • projekt	• Projekt

Warunki zaliczenia

Wykład - warunkiem zaliczenia jest uzyskanie pozytywnych ocen z kolokwium.

Projekt - warunkiem zaliczenia jest uzyskanie pozytywnych ocen ze wszystkich zadań projektowych, przewidzianych do realizacji w ramach zajęć projektowych.

Składowe oceny końcowej = wykład: 50% + projekt: 50%

Literatura podstawowa

1. Sommerville I.: Inżynieria Oprogramowania, WNT, Warszawa, wydanie 10 (1 w WN PWN), 2020
2. Sacha K.: Inżynieria Oprogramowania, PWN, 2021.
3. Karl E Wieggers, Joy Beatty, Specyfikacja oprogramowania. Inżynieria wymagań. Wydanie III (ebook) Helion 2014
4. Spolsky L.: Sztuka pisania oprogramowania, Helion, 2007.

Literatura uzupełniająca

1. Bass L.: Architektura oprogramowania w praktyce, WNT, 2006.
2. Brooks F.: Mityczny osobomiesiąc. Eseje o inżynierii oprogramowania, WNT, 2000.
3. Bernd Bruegge, Allen H. Dutoit - Inżynieria oprogramowania w ujęciu obiektowym. UML, wzorce projektowe i Java

Uwagi

Zmodyfikowane przez dr inż. Michał Doligalski (ostatnia modyfikacja: 23-04-2022 07:40)

Wygenerowano automatycznie z systemu SylabUZ