**Nazwa przedmiotu:**

Inżynieria oprogramowania

**Koordynator przedmiotu:**

Michał Śmiałek

**Status przedmiotu:**

Fakultatywny ograniczonego wyboru

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Elektronika i Telekomunikacja

**Grupa przedmiotów:**

Przedmioty informatyki - obieralne

**Kod przedmiotu:**

IOZ

**Semestr nominalny:**

7 / rok ak. 2016/2017

**Liczba punktów ECTS:**

5

**Liczba godzin pracy studenta związanych z osiągnięciem efektów uczenia się:**

samodzielna lektura podręcznika 25h
opracowanie założeń projektu 13h
wykonanie sprawozdań projektowych 6\*10=60h
analiza uwag nauczyciela do projektu 15h
przygotowanie do egzaminu 15h

**Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:**

konsultacje osobiste 4h
konsultacje mailowe 22h

**Język prowadzenia zajęć:**

polski

**Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:**

opracowanie założeń projektu 13h
wykonanie sprawozdań projektowych 6\*10=60h
analiza uwag nauczyciela do projektu 15h

**Formy zajęć i ich wymiar w semestrze:**

|  |  |
| --- | --- |
| Wykład: | 30h |
| Ćwiczenia: | 15h |
| Laboratorium: | 0h |
| Projekt: | 15h |
| Lekcje komputerowe: | 0h |

**Wymagania wstępne:**

-

**Limit liczby studentów:**

-

**Cel przedmiotu:**

Celem przedmiotu jest wprowadzenie w tematykę metod wytwarzania i eksploatacji oprogramowania oraz wykształcenie praktycznych umiejętności wykorzystania wybranych metod i narzędzi inżynierii oprogramowania. Po ukończeniu zajęć, student powinien znać i rozumieć najważniejsze procesy wytwarzania oprogramowania, umieć zastosować podstawowe zasady obiektowego modelowania oprogramowania w języku UML oraz umieć podjąć współpracę z analitykami i projektantami systemów IT.

**Treści kształcenia:**

Treści przekazywane w ramach przedmiotu można podzielić na trzy części. W pierwszej części przedstawiono wprowadzenie do inżynierii oprogramowania, podstawowych cykli wytwórczych i metodyk. Zaprezentowano kwestie złożoności systemów oraz metody stosowane w celu ich opanowania. Przedstawiono podział cyklu wytwórczego na dyscypliny i fazy. Określono różne sposoby uporządkowania tych elementów w cykle wytwórcze. Przedstawiono także najpopularniejsze metodyki wytwarzania oprogramowania w podziale na metodyki agilne (zwinne) i formalne oraz sposób ich implementacji w organizacjach wytwarzających oprogramowanie.
W drugiej części skoncentrowano się na prezentacji zasad modelowania złożonych systemów oprogramowania. Przedstawiono zasadę abstrakcji i jej realizację w postaci modelowania obiektowego. Dokonano przeglądu i zaprezentowano bliżej podstawowe modele wraz z ich notacją w języku UML. Pokazano, w jaki sposób modelować strukturę i dynamikę systemu przy pomocy różnych modeli języka UML.
Trzecia część przedmiotu zawiera prezentację najważniejszych dyscyplin inżynierii oprogramowania. Przedstawiono w niej podstawowe zasady inżynierii wymagań oraz projektowania systemów, łącznie z zasadami transformacji tworzonych w ich ramach modeli. Przedstawiono także dyscypliny implementacji systemu, zarządzania konfiguracji i zmianami oraz testowania. Opis uzupełniono prezentacją zasad stosowania narzędzi CASE.

**Metody oceny:**

Ocena za egzamin: maksimum 60 pkt.; ocena za projekt: maksimum 40 pkt. Ocena końcowa wynika z sumy punktów za wykład i projekt: od 51 pkt, co 10 pkt. kolejna ocena od 3,0 do 5,0. Uwaga: należy zaliczyć (51%) zarówno wykład, jak i projekt. W kolejnych edycjach przedmiotu proporcja punktów za wykład i projekt może ulec zmianie.

**Egzamin:**

tak

**Literatura:**

I. Somerville, Inżynieria oprogramowania, WNT, 2003
R. S. Pressmann, Praktyczne podejście do inżynierii oprogramowania, WNT, 2004
M. Śmiałek, Zrozumieć UML 2.0, Helion, 2005
W.Dąbrowski, A.Stasiak, M.Wolski, Modelowanie systemów informatycznych w języku UML 2.1, PWN 2007

**Witryna www przedmiotu:**

-

**Uwagi:**

-

## Efekty przedmiotowe

### Profil ogólnoakademicki - wiedza

**Efekt IO\_W01:**

wiedza w zakresie wszystkich istotnych aspektów inżynierii oprogramowania, w tym podbudowana teoretycznie

Weryfikacja:

egzamin

**Powiązane efekty kierunkowe:** K\_W19

**Powiązane efekty obszarowe:** T1A\_W04

### Profil ogólnoakademicki - umiejętności

**Efekt IO\_U01:**

umiejętność porozumiewania się przy pomocy odpowiednich notacji inżynierii oprogramowania

Weryfikacja:

projekt - ocena jakości przekazu wykorzystującego poznane notacje

**Powiązane efekty kierunkowe:** K\_U02

**Powiązane efekty obszarowe:** T1A\_U02

### Profil ogólnoakademicki - kompetencje społeczne

**Efekt IO\_K01:**

umiejętność oceny wpływu inżynierii oprogramowania na społeczeństwo

Weryfikacja:

egzamin, projekt - ocena świadomości studenta w zakresie wpływu na społeczeństwo wybranych elementów inżynierii oprogramowania

**Powiązane efekty kierunkowe:** K\_K02

**Powiązane efekty obszarowe:** T1A\_K02