**Nazwa przedmiotu:**

Inżynieria oprogramowania

**Koordynator przedmiotu:**

Marcin Szlenk

**Status przedmiotu:**

Fakultatywny ograniczonego wyboru

**Poziom kształcenia:**

Studia II stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Przedmioty techniczne - podstawowe

**Kod przedmiotu:**

IOP

**Semestr nominalny:**

1 / rok ak. 2018/2019

**Liczba punktów ECTS:**

4

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

100
- 30 godzin wykładu
- 15 godzin ćwiczeń laboratoryjnych
- 5 godzin konsultacji
- 50 godzin samodzielnej pracy zwiazanej z przygotowaniem do kolokwium i przygotowaniem dokumentacji projektowej niezbędnej do zaliczenia ćwiczeń laboratoryjnych

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

2

**Język prowadzenia zajęć:**

polski

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

2

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

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

**Wymagania wstępne:**

Znajomość programowania w języku obiektowym i strukturalnym.

**Limit liczby studentów:**

120

**Cel przedmiotu:**

Celem przedmiotu jest wprowadzenie słuchaczy w podstawowe zagadnienia inżynierii oprogramowania, obejmujące organizację cyklu wytwarzania aplikacji, metodykę projektowania i weryfikacji programów oraz zarządzania projektem programistycznym. Związane z wykładem laboratorium pozwala studentom nabyć praktyczną umiejętność obiektowego modelowania i projektowania programów oraz planowania projektu.

**Treści kształcenia:**

Wprowadzenie (2). Zakres i cel inżynierii oprogramowania. Podstawowe działania i rozkład kosztów projektu. Projektowanie systemu a projektowanie programu. Kaskadowy i iteracyjny model cyklu życia oprogramowania. Metody i narzędzia.
Analiza strategiczna (2). Modelowanie procesów biznesowych. Studium wykonalności, procedura przetargu. Szacowanie kosztów projektu – metoda punktów funkcyjnych i model COCOMO.
Analiza i projektowanie obiektowe (12). Analiza i modelowanie wymagań metodą przypadków użycia (use cases): przypadki użycia, scenariusze, diagram, reguły biznesowe, schemat dokumentacji; przykład. Modelowanie dziedziny problemu: klasy, obiekty i ich relacje, diagram klas, perspektywy widzenia modelu; diagram maszyny stanowej; przykład. Modelowanie architektury w języku UML (Unified Modeling Language): diagramy pakietów, komponentów i rozmieszczenia; diagramy sekwencji i komunikacji; przeznaczenie modeli. Projektowanie architektury oprogramowania: analiza niewrażliwości (robustness analysis), warstwy danych, logiki biznesowej i prezentacji; modelowanie warstwy danych, projektowanie warstwy logiki biznesowej, wzorce projektowe; komunikacja między warstwami, technologie obiektowe, technologia EJB; przykład. Proces RUP; przykład.
Analiza i projektowanie strukturalne (2). Podstawowe pojęcia i modele strukturalne: hierarchia funkcji, diagram przepływu danych, diagram encji, diagram struktury. Wykorzystanie modeli w projekcie: ustalenie zakresu systemu, budowa modelu funkcjonalnego, specyfikowanie danych, budowa modelu implementacyjnego; przykład.
Testowanie oprogramowania (4). Weryfikacja i zatwierdzanie (verification and validation). Poziomy testowania: testowanie jednostkowe, integracyjne i systemowe; testowanie akceptacyjne. Organizacja procesu testowania: plan testów, specyfikacja testów, procedura testowania; usuwanie błędów. Metryki i metody projektowania testów i testowania; automatyzacja testów. Inne techniki weryfikacji: inspekcje kodu, przeglądy, dowody poprawności.
Zarządzanie projektem informatycznym (4). Struktura organizacyjna projektu. Planowanie projektu: struktura podziału pracy i struktura podziału produktu, tworzenie harmonogramu, narzędzia wspomagające. Tworzenie budżetu projektu. Plan projektu. Zarządzanie ryzykiem, zarządzanie biegiem projektu, metoda PRINCE2.
Kierunki rozwoju inżynierii oprogramowania (2). Metody zwinne. Architektura usługowa (SOA), korporacyjna magistrala usług.
Kolokwia zaliczeniowe (2).

**Metody oceny:**

Wykład: 2 kolokwia.
Laboratorium: ocena wyników pracy studenta.
Wymagane zaliczenie obydwu części przedmiotu.

**Egzamin:**

nie

**Literatura:**

1. Sacha K., Inżynieria oprogramowania, WNT, Warszawa 2010.
2. Fowler M., Scot K.: UML w kropelce; LTP 2002.
3. Booch G., Rumbaugh J., Jacobson I.: UML przewodnik użytkownika, WNT 2001.

**Witryna www przedmiotu:**

https://usosweb.usos.pw.edu.pl/kontroler.php?\_action=katalog2/przedmioty/pokazPrzedmiot&prz\_kod=103C-INIIT-ISP-IOP

**Uwagi:**

Przedmiot prowadzony w każdym semestrze (letnim i zimowym).

## Charakterystyki przedmiotowe

### Profil ogólnoakademicki - wiedza

**Charakterystyka IOP\_W01:**

Ma szczegółową wiedzę z zakresu metod zarządzania projektami

Weryfikacja:

Kolokwium

**Powiązane charakterystyki kierunkowe:** K\_W04

**Powiązane charakterystyki obszarowe:** I.P7S\_WG

**Charakterystyka IOP\_W02:**

Ma teoretyczną i praktyczną wiedzę szczegółową z zakresu inżynierii oprogramowania

Weryfikacja:

Kolokwium + zaliczenie laboratorium

**Powiązane charakterystyki kierunkowe:** K\_W05

**Powiązane charakterystyki obszarowe:** I.P7S\_WG, III.P7S\_WG.o

**Charakterystyka IOP\_W02:**

Ma wiedzę o nowych metodach i trendach rozwojowych w inżynierii oprogramowania

Weryfikacja:

Kolokwium

**Powiązane charakterystyki kierunkowe:** K\_W07

**Powiązane charakterystyki obszarowe:** I.P7S\_WG

**Charakterystyka IOP\_W04:**

Ma uporządkowaną wiedzę na temat istniejących technologii informatycznych i kierunkach ich rozwoju

Weryfikacja:

Kolokwium

**Powiązane charakterystyki kierunkowe:** K\_W10

**Powiązane charakterystyki obszarowe:** I.P7S\_WG, III.P7S\_WG.o

### Profil ogólnoakademicki - umiejętności

**Charakterystyka IOP\_U01:**

Potrafi tworzyć i odczytywać modele w języku UML, używane do komunikacji w środowisku informatycznym

Weryfikacja:

Kolokwium + zaliczenie laboratorium

**Powiązane charakterystyki kierunkowe:** K\_U02

**Powiązane charakterystyki obszarowe:** I.P7S\_UK

**Charakterystyka IOP\_U02:**

Potrafi opracować specyfikację wymagań oraz stworzyć i udokumentować projekt systemu zgodny ze specyfikacją wymagań

Weryfikacja:

Kolokwium + zaliczenie laboratorium

**Powiązane charakterystyki kierunkowe:** K\_U05

**Powiązane charakterystyki obszarowe:** I.P7S\_UW

**Charakterystyka IOP\_U03:**

Potrafi zebrać pozatechniczne wymagania stawiane systemom informatycznym w dziedzinie aplikacji

Weryfikacja:

Kolokwium + zaliczenie laboratorium

**Powiązane charakterystyki kierunkowe:** K\_U07

**Powiązane charakterystyki obszarowe:** III.P7S\_UW.2.o, I.P7S\_UW

### Profil ogólnoakademicki - kompetencje społeczne

**Charakterystyka IOP\_KK01:**

Ma świadomość roli inżyniera w procesie podejmowania decyzji o realizacji bądź zaniechaniu przedsięwzięcia informatycznego

Weryfikacja:

zaliczenie laboratorium

**Powiązane charakterystyki kierunkowe:** K\_K02

**Powiązane charakterystyki obszarowe:** I.P7S\_KK, I.P7S\_KR