

**ANALIZA I MODELOWANIE SYSTEMÓW**

**Kod przedmiotu: IO-AMS2**

**Rodzaj przedmiotu: kierunkowy, obieralny**

**Specjalność: Inżynieria oprogramowania**

**Wydział: Informatyki**

**Kierunek: Informatyka**

**Poziom studiów: drugiego stopnia**

**Profil studiów: praktyczny**

**Forma studiów: stacjonarna/niestacjonarna**

**Rok: 1**

**Semestr: 1**

**Formy zajęć i liczba godzin:**

**Forma stacjonarna**

    wykłady – 15

    laboratorium – 25

**Forma niestacjonarna**

    wykłady – 10

    laboratorium – 18

**Zajęcia prowadzone są w języku polskim.**

**Liczba punktów ECTS: 3**

**Osoby prowadzące:**

    wykład:

    laboratorium:

---

**1. Założenia i cele przedmiotu:**

Przedmiot poświęcony jest przekazaniu wiedzy i wyrobieniu umiejętności w zakresie analizy systemowej oraz tworzenia modeli systemu informatycznego, pozwalających na spójne opracowanie założeń projektowych i rozpoczęcie procesu tworzenia oprogramowania. Tworzenie współczesnych systemów informatycznych jest zagadnieniem złożonym i wielowymiarowym — oprócz biegłości w zakresie programowania, istotnym elementem jest umiejętność wykonywania analizy problemu, opracowania wymagań dla systemu, ich weryfikacja, opracowanie specyfikacji wymagań, zarządzanie zmianami.

**2. Określenie przedmiotów wprowadzających wraz z wymaganiami wstępnymi:**

Przedmioty wprowadzające to: Programowanie obiektowe.

### 3. Opis form zajęć

#### a) Wykłady

- **Treści programowe:**
  - Metody inżynierii oprogramowania.
  - Analiza wymagań dla systemu informatycznego, opracowywanie specyfikacji wymagań, wymagania funkcjonalne i нефункционалне.
  - Modelowanie wymagań, podejścia, diagramy.
  - Metodyka a metoda projektowania systemu, cykl życia systemu, fazy cyklu życia systemu, modele cyklu życia systemu, klasyfikacja metodyk projektowania.
  - Metodyki strukturalne, podejścia, diagramy.
  - Modelowanie struktury baz danych, modele, notacje, diagramy.
  - Metodyki obiektowe, podejścia, diagramy.
  - Wzorce projektowe a metodyki projektowania.
  - Testowanie i weryfikacja oprogramowania.
- **Metody dydaktyczne:**
  - Wykład prowadzony metodą tradycyjną z wykorzystaniem rzutnika multimedialnego, z wykorzystaniem materiałów udostępnianych studentom w postaci elektronicznej.
- **Forma i warunki zaliczenia:**
  - Pozytywna ocena testu egzaminacyjnego.
- **Wykaz literatury podstawowej:**
  1. Booch G., Rumbaugh J., Jacobson I.: UML – przewodnik użytkownika. WNT,.
  2. Gamma E., Helms R., Johnson R., Vlissides J.: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley.
  3. Flasiński M. : Wstęp do analitycznych metod projektowania systemów informatycznych, WNT.
- **Wykaz literatury uzupełniającej:**
  1. Miękina L.: Inżynieria Oprogramowania, UWND AGH, Kraków.
  2. Pressman R.S., Praktyczne podejście do inżynierii oprogramowania, WNT.
  3. Havey M. , Essential Buiseness Process Modeling, O'REILLY

#### b) Laboratorium

- **Treści programowe:**
  - Przegląd systemów wspomagających analizę i projektowanie systemów informatycznych.
  - Analiza wymagań, ćwiczenia w opracowaniu i weryfikacji specyfikacji wymagań, modelowanie wymagań dla systemu informatycznego.
  - Metodyki strukturalne – ćwiczenia praktyczne, dyskusja wad i zalet.
  - Projektowanie struktury relacyjnych baz danych, modele, narzędzia.
  - Metodyki obiektowe — ćwiczenia praktyczne, projektowanie obiektowe a programowanie obiektowe.
  - Zarządzanie zmianami w projekcie systemu.
  - Testowanie i weryfikacja oprogramowanie.
  - Wykorzystanie narzędzi wspomagających testowanie.

- Opracowanie projektu.
- **Metody dydaktyczne:**
  - Prezentacja treści i dyskusja moderowana.
  - Metoda problemowa – studium przypadku, burza mózgów.
  - Metoda laboratoryjna – ćwiczenia laboratoryjne z wykorzystaniem komputerów.
- **Forma i warunki zaliczenia:**
  - Pozytywna średnia ocena realizacji wskazanych zadań w ramach 3 prac kontrolnych.
  - Pozytywna ocena aktywności studenta podczas zajęć, w tym ocena biegłości w stosowaniu narzędzi wspomagających projektowanie systemów.
- **Wykaz literatury podstawowej:**
  - Dąbrowski W., Stasiak A., Wolski M., Modelowanie systemów informatycznych w języku UML 2.1, PWN.
  - Wrycza S., Marcinkowski B., Wyrzykowski K., Język UML 2.0 w modelowaniu systemów informatycznych, Helion.
- **Wykaz literatury uzupełniającej:**
  - Śmiałek M., Zrozumieć UML 2.0. Metody modelowania obiektowego, Helion.
  - Dumnicki R., Kasprzyk A., Kozłowski M., Analiza i projektowanie obiektowe, Helion.

#### 4. Opis sposobu wyznaczania punktów ECTS

##### a. forma stacjonarna

Forma zajęć	Formy aktywności studenta	Średnia liczba godzin na zrealizowanie aktywności
Wykład	Kontakt z nauczycielem	15
	Przygotowanie do egzaminu	15
Laboratorium	Kontakt z nauczycielem	25
	Czytanie wskazanej literatury	10
	Przygotowanie do pracy kontrolnej	10

<b>Całkowita ilość godzin aktywności studenta</b>	<b>75</b>
<b>Liczba punktów ECTS dla modułu/przedmiotu</b>	<b>3</b>

##### b. forma niestacjonarna

Forma zajęć	Formy aktywności studenta	Średnia liczba godzin na zrealizowanie aktywności
Wykład	Kontakt z nauczycielem	10
	Przygotowanie do egzaminu	20
Laboratorium	Kontakt z nauczycielem	18
	Czytanie wskazanej literatury	17
	Przygotowanie do pracy kontrolnej	10

<b>Całkowita ilość godzin aktywności studenta</b>	<b>75</b>
<b>Liczba punktów ECTS dla modułu/przedmiotu</b>	<b>3</b>

## 5. Wskaźniki sumaryczne

### a. forma stacjonarna

- a) liczba godzin dydaktycznych (tzw. kontaktowych) i liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich
  - Liczba godzin kontaktowych – 35
  - Liczba punktów ECTS – 1,4
- b) liczba godzin dydaktycznych (tzw. kontaktowych) i liczba punktów ECTS na zajęciach o charakterze praktycznym.
  - Liczba godzin kontaktowych – 25
  - Liczba punktów ECTS – 1,8

### b. forma niestacjonarna

- a) liczba godzin dydaktycznych (tzw. kontaktowych) i liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich
  - Liczba godzin kontaktowych – 28
  - Liczba punktów ECTS – 1,2
- b) liczba godzin dydaktycznych (tzw. kontaktowych) i liczba punktów ECTS na zajęciach o charakterze praktycznym.
  - Liczba godzin kontaktowych – 18
  - Liczba punktów ECTS – 1,8

## 6. Zakładane efekty uczenia się

<b>Efekt przedmiotowy (Symbol)</b>	<b>Efekty uczenia się dla przedmiotu</b>	<b>Odniesienie do kierunkowych efektów uczenia się</b>
IO-AMS2_W1	Student posiada wiedzę z zakresu modeli cyklu życia systemu informatycznego oraz zakresu metodyk projektowania strukturalnego i obiektowego.	IIK_W03 IIK_W05
IO-AMS2_W2	Student zna metody i zasady analizy i modelowania wymagań funkcjonalnych i нефункциональных.	IIK_W07 IIK_W08
IO-AMS2_U1	Student potrafi dokonać analizy wymagań użytkownika i opracować specyfikację wymagań dla systemu informatycznego.	IIK_U12 IIK_U13
IO-AMS2_U2	Student potrafi wykorzystywać techniki projektowania strukturalnego i obiektowego, potrafi posługiwać się programami wspomagającymi prace projektowe.	IIK_U14 IIK_U15
IO-AMS2_K1	Student posiada kompetencje interpersonalne, potrafi pracować indywidualnie oraz w zespole, dokonuje właściwego podziału pracy w zespole.	IIK_K01 IIK_K02 IIK_K03

**7. Odniesienie efektów uczenia się do form zajęć i sposób oceny osiągnięcia przez studenta efektów uczenia się.**

Efekt przedmiotowy (Symbol)	Forma zajęć		Sposób sprawdzenia osiągnięcia efektu
	Wykład	Laboratorium	
IO-AMS2_W1	v		Egzamin
IO-AMS2_W2	v	v	Egzamin
IO-AMS2_U1	v	v	Prace kontrolne
IO-AMS2_U2		v	Prace kontrolne
IO-AMS2_K1		v	Prace kontrolne, ocena aktywności

**8. Kryteria uznania osiągnięcia przez studenta efektów uczenia się.**

Efekt przedmiotowy (Symbol)	Efekt jest uznawany za osiągnięty, gdy student:
IO-AMS2_W1	Poprawnie odpowiada na ponad 50% pytań testu egzaminacyjnego
IO-AMS2_W2	Poprawnie odpowiada na ponad 50% pytań testu egzaminacyjnego
IO-AMS2_U1	Osiąga ponad 50% punktów w pracach kontrolnych.
IO-AMS2_U2	Osiąga ponad 50% punktów w pracach kontrolnych.
IO-AMS2_K1	Osiąga ponad 50% punktów w pracach kontrolnych.