

KARTA PRZEDMIOTU

Nazwa przedmiotu: Inżynieria oprogramowania (InfKAu>SI4INZOPR19)

Nazwa w języku polskim:

Nazwa w jęz. angielskim: Software Engineering

Dane dotyczące przedmiotu:

Jednostka oferująca przedmiot: Wydział Automatyki, Elektroniki i Informatyki

Przedmiot dla jednostki: Politechnika Śląska

Domyślny typ protokołu dla przedmiotu:

ZAL

Język wykładowy:

polski

Strona WWW:

<https://platforma2.polsl.pl/rau2/course/view.php?id=1077>

Skrócony opis:

Celem przedmiotu jest przedstawienie wybranych zagadnień wchodzących w zakres inżynierii programowania, ze szczególnym uwzględnieniem tych, które odnoszą się do cyklu życia jednostek oprogramowania o ograniczonych rozmiarach. Między innymi studenci zapoznają się ogólnie z problematyką projektowania systemów informatycznych z wykorzystaniem nowoczesnych metodologii i narzędzi CASE.

Opis:

Zagadnienia omawiane w ramach wykładu:

1. Modele wytwarzania oprogramowania
2. Metody zwinnego wytwarzania oprogramowania
3. Inżynieria wymagań
4. Tworzenie modelu systemu
5. Język UML
6. Projektowanie architektury systemu
7. Projekt i implementacja
8. Wzorce i antywzorce projektowe
9. Dobre praktyki programowania
10. Testowanie oprogramowania
11. Ewolucja oprogramowania

Zagadnienia realizowane w trakcie ćwiczeń laboratoryjnych:

1. Optymalizacja czasowa
2. Rozwój podyktowany wzorcami
3. Wzorce projektowe

Liczba godzin zajęć z bezpośrednim udziałem nauczycieli akademickich lub innych osób prowadzących zajęcia i studentów

Wykład: 30h

Ćwiczenia laboratoryjne: 15h

Liczba godzin przeznaczonych na pracę własną studenta

Przygotowanie do zaliczenia ćwiczeń laboratoryjnych: 3h

Przygotowanie do zaliczenia wykładu: 12h

Całkowita liczba godzin: 60

Liczba punktów ECTS: 2

Literatura:

Inżynieria oprogramowania:

1. I. Sommerville: Software Engineering, 10th edition. Pearson, 2015.
2. R. S. Pressman, B. R. Maxim: Software Engineering A Practitioner's Approach. McGraw-Hill Education, NY, 2015.
3. S. R. Schach: Object-Oriented and Classical Software Engineering. McGraw-Hill Education, NY, 2010.
4. R. J. Leach: Introduction to Software Engineering. CRC Press, Taylor & Francis Group, 2016.
5. G. O'Regan: Concise Guide to Software Engineering. From Fundamentals to Application Methods. Springer, 2017.
6. R. Stephens: Beginning Software Engineering. John Wiley & Sons, Inc., 2015.
7. K. Schwaber & J. Sutherland: Przewodnik po Scrumie, XI 2020.

UML:

8. K. Hamilton, R. Miles: Learning UML 2.0. O'Reilly, 2006.

Wzorce projektowe:

9. E. Gamma, R. Helm, R. Johnson, J. Vlissides: Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley Professional Computing, 1997.
10. A. Shvets: Dive into Design Patterns, Refactoring.Guru, Dec 5th, 2018
11. Refactoring GURU
12. Source Making
13. W.J. Brown, R. Malveau, S. McCormick, T. Mowbray: AntiPatterns: Refactoring Software, Architectures, and Projects in Crisis. USA: John Wiley & Sons, 1998.

Efekty uczenia się:

Po zakończeniu kursu, student zna:

- szczegółowe zagadnienia algorytmiki, programowania, projektowania i modelowania systemów informatycznych (test), K1A_W11

- podstawowe zasady tworzenia i rozwoju różnych form indywidualnej przedsiębiorczości, wykorzystującej wiedzę z zakresu informatyki (test) K1A_W21

Po zakończeniu kursu, student potrafi:

- przeprowadzać systematyczną weryfikację zarówno sprzętu, jak i oprogramowania (laboratorium) K1A_U15

- dokonywać krytycznej analizy sposobu funkcjonowania istniejących rozwiązań technicznych i oceniać te rozwiązania (laboratorium) K1A_U19

- ocenić przydatność rutynowych metod i narzędzi informatycznych oraz wybrać i zastosować właściwą metodę i narzędzia do typowych zadań informatycznych lub realizacji operacji arytmetycznych (laboratorium) K1A_U21

- brać udział w debacie – przedstawiać i oceniać różne opinie i stanowiska oraz dyskutować o nich (laboratorium wykład) K1A_U29

Metody i kryteria oceniania:

Na zaliczenie przedmiotu składa się:

1. Zaliczenie wykładu odbywa się w formie kolokwium.

2. Zaliczenie ćwiczeń laboratoryjnych polega na realizacji zadań wyznaczonych przez prowadzących w trakcie zajęć. Praca na poszczególnych zajęciach jest oceniana.

Ocena końcowa jest średnią z ocen uzyskanych z kolokwium wykładowego oraz średniej z ocen uzyskanych na poszczególnych laboratoriach.

Sylabus obowiązuje od roku akademickiego 2025/2026, a jego zawartość nie podlega zmianom w trakcie trwania semestru.

Punkty przedmiotu w cyklach:

Informatyka, niestacjonarne (zaoczne) I stopnia inżynierskie 7 sem. (InfAu-NZI7)			
Typ punktów	Liczba	Cykl pocz.	Cykl kon.
Europejski System Transferu Punktów (ECTS)	2	2020/2021-L	

Informatyka, stacjonarne I stopnia inżynierskie 7 sem. (InfKAu-SI7)			
Typ punktów	Liczba	Cykl pocz.	Cykl kon.
Europejski System Transferu Punktów (ECTS)	2	2020/2021-L	