

IT systems design - opis przedmiotu

Informacje ogólne

Nazwa przedmiotu	IT systems design
Kod przedmiotu	11.3-WE-INFP-ITSD- Er
Wydział	Wydział Nauk Inżynieryjno-Technicznych
Kierunek	Informatyka
Profil	ogólnoakademicki
Rodzaj studiów	Program Erasmus pierwszego stopnia
Semestr rozpoczęcia	semestr zimowy 2021/2022

Informacje o przedmiocie

Semestr	5
Liczba punktów ECTS do zdobycia	6
Typ przedmiotu	obowiązkowy
Język nauczania	angielski
Sylabus opracował	<ul style="list-style-type: none">dr inż. Tomasz Gratkowskidr 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
Wykład	30	2	-	-	Egzamin
Laboratorium	30	2	-	-	Zaliczenie na ocenę

Cel przedmiotu

- To introduce students with the basics of software engineering and software modeling methods.
- Formation of the students understanding of the principles of object-oriented programming.
- To familiarize students with the principles of compiler design.

Wymagania wstępne

Principles of programming, Software engineering

Zakres tematyczny

Elements of software engineering.

Software development. Software crisis and ways to tackle them. Conceptual modeling. The role of modeling in software development. Historical background of modern modeling techniques. Object-oriented design methods and UML notation. Structured and object-oriented methodology.

Business process modeling in BPMN. Create a software model based on BPMN model. Analysis and requirements modeling. Domain analysis and modeling. Architecture design. Software life cycle. System Design and System Analysis. The basic object-oriented concepts and relationships between objects. Modeling links between objects. Messages and procedure calls. Classes, inheritance, generalization / specialization, polymorphism, interfaces.

Unified Modelling Language UML. The genesis of the uprising. Definition and objectives of the creation of UML. The scope of UML. UML diagrams. Characteristic diagrams.

Metody kształcenia

Lecture: conventional lecture

Laboratory: laboratory exercises, group work

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Student is able to prepare design documentation for IT system.		<ul style="list-style-type: none">bieżąca kontrola na zajęciach	<ul style="list-style-type: none">Laboratorium
Can develop customer requirements for the IT system.		<ul style="list-style-type: none">bieżąca kontrola na zajęciach	<ul style="list-style-type: none">Laboratorium
Can customize the methodology of project management to the design of the IT system.		<ul style="list-style-type: none">egzamin - ustny, opisowy, testowy i inne	<ul style="list-style-type: none">Wykład
Student can modeling any IT system using UML.		<ul style="list-style-type: none">bieżąca kontrola na zajęciach	<ul style="list-style-type: none">Laboratorium

Warunki zaliczenia

Lecture - obtaining a positive grade in written exam.

Laboratory - the main condition to get a pass are sufficient marks for all exercises and tests conducted during the semester.

Calculation of the final grade: = lecture 50% + laboratory 50%.

Literatura podstawowa

- Alan Dennis, Barbara Haley Wixom, David Tegarden: Systems Analysis and Design: An Object-Oriented Approach with UML, Wiley, 2015
- Brett D. McLaughlin, Gary Pollice, Dave West: Head First Object-Oriented Analysis and Design, O'Reilly Media, 2006
- Martin Fowler: UML Distilled: A Brief Guide to the Standard Object Modeling Language, Addison-Wesley, 2003
- By Martin Fowler: UML Distilled: A Brief Guide to the Standard Object Modeling Language, Addison-Wesley, 2003
- Stephen Schach: Object-Oriented and Classical Software Engineering (Irwin Computer Science), McGraw-Hill Education, 2010

Literatura uzupełniająca

Uwagi

Zmodyfikowane przez dr inż. Tomasz Gratkowski (ostatnia modyfikacja: 21-07-2021 10:06)

Wygenerowano automatycznie z systemu SylabUZ