

Projektowanie systemów informatycznych - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Projektowanie systemów informatycznych
Kod przedmiotu	11.3-WK-IIEP-PSI-W-S14_pNadGenLP3ME
Wydział	Wydział Matematyki, Informatyki i Ekonometrii
Kierunek	Informatyka i ekonometria
Profil	ogólnoakademicki
Rodzaj studiów	pierwszego stopnia z tyt. licencjata
Semestr rozpoczęcia	semestr zimowy 2019/2020

Informacje o przedmiocie	
Semestr	5
Liczba punktów ECTS do zdobycia	4
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	dr inż. Janusz Jabłoński

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

Zapoznanie z teoretycznymi i praktycznymi aspektami analizy i projektowania systemów informatycznych z wykorzystaniem metodyk, technik i narzędzi wspomagających zarządzanie projektami.

Wymagania wstępne

Programowanie komputerów, Technologie Informatyczne.

Zakres tematyczny

Wykład

1. Zakres materiału, literatura oraz omówienie formy i treści zaliczenia przedmiotu (2 godz.)
2. Metodologiczne podstawy tworzenia systemów informatycznych: rodzaje systemów, zakres i składniki metodyki tworzenia. Klasyfikacja metodyk tworzenia. (4 godz.)
3. Cykl życia systemu: kaskadowy, spiralny, przyrostowy. (2 godz.)
4. Fazy cyklu: planowanie, analiza, projektowanie, wdrażanie, użytkowanie systemów. (2 godz.)
5. Modyfikacje cyklu życia systemu: prototypowanie, pakiety zastosowań, metodyki „zwinne”. (2 godz.)
6. Planowanie systemów informatycznych: cele planowania, proces planowania, formułowanie strategii informatyzacji, studium wykonalności, metody analizy sytuacyjnej, zespół projektowy. (2 godz.)
7. Model klient – server oraz architektura trójwarstwowa w realizacji systemów informatycznych. (2 godz.)
8. Metody i techniki projektowania systemów informatycznych – modele związków encji, diagramy przepływu danych, słowniki/skorowidze danych, techniki decyzyjne, diagramy struktur. Diagramy UML w modelowaniu struktury i dynamiki systemów obiektowych. (4 godz.)
9. Metodyka obiektowa - techniki i modele UML w projektowaniu systemów informatycznych. (4 godz.)
10. Metodyki społeczne. (2 godz.)
11. Komputerowo wspomagane tworzenie systemów informatycznych (2 godz.)
12. Pakiety wspomagające zarządzanie projektami: istota i generacje, rodzaje pakietów na rynku oprogramowania. (2 godz.)

Laboratorium

1. Wprowadzenie, omówienie zakres materiału i formy zaliczenia laboratorium (2 godz.)
2. Etapy cyklu życia oprogramowania jako dokumentacja projektowa - przykłady (2 godz.)
3. Podstawowe funkcjonalności systemu informatycznego w praktyce gospodarczej (2 godz.)
4. Cykl życia systemu informatycznego - Diagram Ganta w MicrosoftProject (2 godz.)
5. Modelowanie systemu transakcyjnego "Sklep" - MSVisio w modelowaniu (2 godz.)
6. Projektowanie bazy danych dla zadanych wymagań - profil klienta, profil towaru (2 godz.)
7. Projektowanie interfejsu użytkownika i programowanie zdarzeniowe w MS Excel (2 godz.)
8. Implementacja funkcjonalności systemu "Sklep" w VisualBasic-u (2 godz.)
9. Modelowanie struktury systemu - Eclipse i projektowanie diagramu klas (2 godz.)
10. JDBC i bazy danych - połączenie z bazą danych i zapytania z poziomu Eclipse (2 godz.)
11. Serwer aplikacji i architektura trójwarstwowa w realizacji systemu transakcyjnego (6 godz.)
12. Testowanie i weryfikacja systemu "Sklep" (2 godz.)
13. Sprawdzenie wiadomości i wystawienie zaliczenia laboratorium (2 godz.)

Metody kształcenia

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Student zna podstawowe cykle życia oprogramowania oraz techniki metodyk strukturalnych i obiektowych w wytwarzaniu oprogramowania	K_W09	- egzamin - ustny, opisowy, testowy i inne - kolokwium. sprawdzanie stopnia przygotowania do ćwiczeń laboratoryjnych oraz sprawozdań z realizowanych laboratoriów	- Wykład - Laboratorium
Student rozumie konieczność systematycznej pracy, także w zespole, nad projektami o długofalowym charakterze.	K_K05	kolokwium	Laboratorium
Student posiada podstawowe umiejętności analityczne i projektowe niezbędne w kontaktach ze specjalistami z dziedziny proble-mu oraz branży IT, potrafi specyfikować wymagania wobec systemu informatycznego oraz potrafi korzystać z notacji obiektowych zgodnych z UML w zakresie niezbędnym do implementacji systemów informatycznych w architekturze trójwarstwowej, korzysta z oraz narzędzia CASE w realizacji prostych systemów informatycznych.	K_U23 K_U27 K_U28 K_U34	- egzamin - ustny, opisowy, testowy i inne - kolokwium. sprawdzanie stopnia przygotowania do ćwiczeń laboratoryjnych oraz sprawozdań z realizowanych laboratorió	- Wykład - Laboratorium
Student potrafi współdziałać i pracować w zespole.	K_K09	- sprawdzian z progami punktowymi - sprawdzanie sprawozdań z realizowanych laboratoriów	Laboratorium
Student umie dobrać cykl życia oprogramowania do zaistniałych wymagań i ograniczeń.	K_U21	- egzamin - ustny, opisowy, testowy i inne - kolokwium. sprawdzanie stopnia przygotowania do ćwiczeń laboratoryjnych oraz sprawozdań z realizowanych laboratoriów	- Wykład - Laboratorium
Student posiada ogólną wiedzę na temat systemów informatycznych.	K_W01	- egzamin - ustny, opisowy, testowy i inne - kolokwium. sprawdzanie stopnia przygotowania do ćwiczeń laboratoryjnych oraz sprawozdań z realizowanych laboratoriów	- Wykład - Laboratorium

Warunki zaliczenia

1. Ocena stopnia przygotowania do ćwiczeń laboratoryjnych oraz sprawozdań z realizowanych laboratoriów.
2. Kolokwium pisemne z progami punktowymi oraz zadaniami pozwalającymi ocenić, czy student osiągnął efekty kształcenia w stopniu minimalnym.
3. Egzamin pisemny składający się z pytań testowych i zadań, weryfikujący znajomość modeli, technik i metod.

Warunkiem zaliczenia przedmiotu jest pozytywna ocena z laboratorium i z egzaminu. Ostateczna ocena z przedmiotu uwzględnia ocenę z laboratorium (60%) i ocenę z egzaminu (40%), przy założeniu, że student osiągnął wszystkie zakładane efekty kształcenia w stopniu dostatecznym.

Literatura podstawowa

1. R. Barker, C. Longman, CASE Method, modelowanie funkcji i procesów, WNT, 1996.
2. V. Sthern, C++ Inżynieria Programowania, Helion, Gliwice 2004.
3. J. Cogswell, Tworzenie użytecznego oprogramowania, Warszawa 2005.
4. J. Górski, Inżynieria oprogramowania w projekcie informatycznym, Warszawa 2000.

Literatura uzupełniająca

1. P. Benon-Davies, Inżynieria Systemów Informacyjnych, WNT, 1999.
2. J. Roszkowski, Analiza i projektowanie strukturalne, Wydawnictwo Helion, 1998.
3. C. L. Hall, Techniczne podstawy systemów klient–serwer, WNT, 1996.

Uwagi

Zmodyfikowane przez dr Robert Dylewski, prof. UZ (ostatnia modyfikacja: 17-09-2019 09:32)

Wygenerowano automatycznie z systemu SylabUZ