

STUDIA I STOPNIA
stacjonarne/~~niestacjonarne~~
KIERUNEK ELEKTROTECHNIKA

PRZEDMIOT:	Projektowanie systemów informatycznych	
ROK:	III	
SEMESTR:	5	
RODZAJ ZAJĘĆ I LICZBA GODZIN:	<input checked="" type="checkbox"/> Wykład	30
	<input type="checkbox"/> Ćwiczenia	-
	<input type="checkbox"/> Laboratorium	-
	<input checked="" type="checkbox"/> Projekt	30
LICZBA PUNKTÓW ECTS:	4	
RODZAJ PRZEDMIOTU:	podstawowy	

PROWADZĄCY PRZEDMIOT:	Dr inż. Marek Miłosz
--------------------------	----------------------

WYMAGANIA WSTĘPNE:	brak
--------------------	------

CELE I EFEKTY KSZTAŁCENIA
<p>Celem kształcenia jest zapoznanie studentów z modelami wytwarzania oprogramowania, ich etapami, stosowanymi metodykami i narzędziami wspomagającymi.</p> <p>Efektami kształcenia jest umiejętność specyfikowania wymagań do oprogramowania, tworzenia modelu/projektu oprogramowania w różnych metodykach (strukturalnych i obiektowych) i narzędziach wspomagających, posługiwaniu się wzorcami projektowymi, projektowania interfejsu, uczestniczenia w procesach implementowania, testowania i walidacji, także rozwoju oprogramowania.</p>

TREŚCI PROGRAMOWE PRZEDMIOTU
Wykład
<ol style="list-style-type: none"> 1. Typowe etapy wytwarzania oprogramowania i ich zawartość. Modele wytwarzania oprogramowania (kaskadowy, ewolucyjne, iteracyjne, agile, XP). Metoda prototypowania i użycie wzorców projektowych. 2. Analiza systemów informacyjno-decyzyjnych. Modelowanie procesów biznesowych i specyfikacja zawartości dokumentów. Notacje BPMN i BNF. 3. Typy wymagań. Pozyskiwanie wymagań, konsolidacja i redakcja wymagań. Przygotowanie specyfikacji oprogramowania wg standardu IEEE i kryteria jej akceptacji. Notacje formalizacji wymagań (szablony, scenariusze, przypadki użycia, listy hierarchiczne). Zarządzanie wymaganiami. 4. Projektowanie/modelowanie oprogramowania. Metodyki strukturalne i obiektowe. Notacje. 5. Projekt architektury systemu. Przegląd współczesnych architektur. 6. Szczegółowe techniki modelowania procesów i struktur danych oprogramowania. Modele konceptualne, logiczne i implementacyjne. Mapowanie modeli. 7. Charakterystyka środowisk CASE. Rola słownika danych i repozytorium. Techniki pracy w narzędziach CASE. Zintegrowane środowiska rozwoju oprogramowania. 8. Standardy i projekt interfejsu z użytkownikiem. Zasady konstrukcji poprawnego interfejsu. Narzędzia wspomagające. 9. Kodowanie danych. Konstrukcja kodów. Cyfry kontrolne.

10. Testowanie i walidacja oprogramowania. Cele, zakres i typy metod testowania oprogramowania. Problem organizacji procesu testowania.
11. Rozwój oprogramowania na etapie eksploatacji. Proces wnoszenia zmian do oprogramowania. Zarządzanie konfiguracjami.
12. Systemy zapewnienia jakości oprogramowania. Modele TQM, ISO 9000x, CMM i EFQM.

Projekt

Praktyczne zapoznawanie się z technikami wykorzystywanymi w inżynierii oprogramowania: BPMN, BNF, DFD/ERD, UCD oraz narzędziami CASE je wspierającymi.

W trakcie zajęć laboratoryjnych studenci wykonują projekt prostej aplikacji ewidencyjno-rozliczeniowej z wykorzystaniem prezentowanych technik w cyklu: analiza obszaru – specyfikowanie wymagań – projekt architektury – projekt szczegółowy (w tym systemu kodowania, interfejsu) – projekt przypadków testowych.

WARUNKI ZALICZENIA

WYKŁAD

Egzamin

ĆWICZENIA/PROJEKT/LABORATORIUM

Zaliczenie

WYKAZ ZALECANEJ LITERATURY PODSTAWOWEJ

1.	Sommerville I., <i>Inżynieria oprogramowania</i> , WNT, Warszawa 2003
2.	Pilone D., Miles R., <i>Head First. Software Development. Edycja polska</i> , Helion, Gliwice, 2008
3.	Pressman Roger S., <i>Praktyczne podejście do inżynierii oprogramowania</i> , WNT, Warszawa 2004
4.	<i>SWEBOK. Guide to the Software Engineering Body of Knowledge</i> . IEEE CS (www.computer.org)
5.	Dąbrowski W., Subieta K., <i>Podstawy inżynierii oprogramowania</i> , Wydawnictwo PJWSTK, Warszawa, 2005
6.	

WYKAZ ZALECANEJ LITERATURY UZUPEŁNIAJĄCEJ

1.	Gamma E., Helm R., Johnson R., Vlissides J., <i>Wzorce projektowe. Elementy oprogramowania obiektowego wielokrotnego użytku</i> , WNT, Warszawa, 2008
2.	Hofmeister Ch., Nord R., Soni D., <i>Tworzenie architektury oprogramowania</i> , WNT, Warszawa, 2006
3.	Szyperski C., <i>Oprogramowanie komponentowe. Obiekty to za mało</i> , WNT, Warszawa, 2001
4.	Jaskiewicz A., <i>Inżynieria oprogramowania</i> , Helion, Gliwice, 1997
5.	Booch G., Rumbaugh J., Jacobson I., <i>UML. Przewodnik użytkownika</i> , WNT, Warszawa, 2001