

OPIS PRZEDMIOTU KSZTAŁCENIA

Nazwa przedmiotu Inżynieria oprogramowania II					
Nazwa jednostki prowadzącej przedmiot Instytut Matematyki					
Kierunek	Specjalność	Specjalizacja	Semestr/y	Poziom kształcenia/ profil kształcenia	Forma studiów
Informatyka	Programowanie	-	6	SPS/ praktyczny	stacjonarne/ niestacjonarne
Nazwisko osoby prowadzącej (osób prowadzących) dr inż. Zbigniew Ledóchowski, dr Ryszard Motyka					
Formy zajęć	Liczba godzin				Liczba punktów ECTS
	N (nauczyciel)		S (student)		
	studia stacjonarne	studia niestacjonarne	studia stacjonarne	studia niestacjonarne	
(CL) ćwiczenia laboratoryjne	20	12	80	88	4
Przygotowanie do zajęć (w tym rozwiązywanie zadań domowych)			20	20	
Przygotowanie projektu (także zespołowego)			40	48	
Przygotowanie do kolokwium			20	20	
Razem	20	12	80	88	4
Metody dydaktyczne					
<ul style="list-style-type: none"> • ćwiczenia laboratoryjne: ćwiczenia praktyczne - rozwiązywanie zadań z wykorzystaniem komputera, metoda projektu, metoda problemowa 					
Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi					
<p>A. Wymagania formalne: Algorytmy i struktury danych, Programowanie I-V, Wstęp do gromadzenia i przetwarzania danych, Bazy danych</p> <p>B. Wymagania wstępne: wiadomości i umiejętności z przedmiotów związanych z architekturą komputera i podstawami informatyki oraz wiedza z zakresu podstaw programowania</p>					
Cele przedmiotu					
<ul style="list-style-type: none"> • zapoznanie z możliwościami wykorzystania narzędzi informatycznych wspierających definiowanie, prowadzenie i organizację projektów informatycznych • wskazanie na wagę zapewniania jakości jako niezbędnego elementu w procesie wytwarzania oprogramowania. • przekazanie wiedzy dotyczącej różnych metodyk prowadzenia projektu informatycznego. • zapoznanie z wybranymi zasadami zarządzania przedsięwzięciami programistycznymi • zapoznanie się z podstawowymi pojęciami i zagadnieniami związanymi z zarządzaniem zmianą w projekcie informatycznym oraz systemami kontroli wersji • promowanie wiedzy i umiejętności związanych z pracą w zespołach projektowych 					
Treści przedmiotu					
<ol style="list-style-type: none"> 1. Zapewnianie jakości na etapie analizy, projektowania i wytwarzania oprogramowania. 2. Metody formalne i empiryczne oceny jakości oprogramowania. Testowanie. Kategorie testów. Organizacja i zasady testowania oprogramowania. 3. Tworzenie przykładowych plików konfiguracyjnych i instalacyjnych – elementy wiedzy o zarządzaniu konfiguracją oprogramowania. 4. Wybrane problemy związane z wdrożeniem oprogramowania. 5. Metodyki tworzenia oprogramowania. Klasyczny model kaskadowy i metodyki zwinne. PRINCE2, SCRUM oraz XP-uwagi ogólne. 6. Elementy wiedzy o prowadzeniu przedsięwzięć informatycznych. Struktury organizacyjne w projektach informatycznych. 					

7. Ewolucja projektów informatycznych. Zapewnienie jakości projektu informatycznego w kontekście jego cyklu życia. Rola refaktoryzacji.

<p>Efekty kształcenia</p> <p>Wiedza</p> <p>W_01 charakteryzuje formalne i empiryczne metody oceny jakości oprogramowania, w tym organizację i zasady testowania oprogramowania</p> <p>W_02 ocenia jakość systemów informatycznych ze szczególnym uwzględnieniem interfejsu użytkownika</p> <p>W_03 opisuje najważniejsze metodyki realizacji projektów informatycznych, w tym możliwości ich doboru do konkretnych przedsięwzięć</p> <p>W_04 opisuje zasady i struktury organizacyjne charakterystyczne dla projektów informatycznych</p> <p>Umiejętności</p> <p>U_01 konstruuje plan testowania oprogramowania zgodnie z uzyskaną wiedzą</p> <p>U_02 tworzy pliki będące elementem konfigurowania i zarządzania oprogramowaniem</p> <p>U_03 konstruuje plany wdrożenia oprogramowania</p> <p>U_04 dobiera odpowiednie metodyki realizacji projektów informatycznych do zaaranżowanych problemów</p> <p>Kompetencje społeczne</p> <p>K_01 wykazuje kreatywność przy rozwiązywaniu problemów</p> <p>K_02 pracuje w zespole</p> <p>K_03 ma świadomość ograniczeń wynikających z niedostatecznej wiedzy i umiejętności i rozumie potrzebę dalszego kształcenia, jest otwarty na poszukiwanie niestandardowych rozwiązań</p>	<p>Sposób zaliczenia oraz formy i podstawowe kryteria oceny/wymagania egzaminacyjne</p> <p>A. Sposób zaliczenia</p> <p>CL – zaliczenie z oceną</p> <p>B. Sposoby weryfikacji i oceny efektów</p> <p>(CL) Ćwiczenia laboratoryjne - kolokwium dotyczące problematyki tworzenia planu testowego oraz związane z tworzeniem plików konfiguracyjnych – efekty: U_01,U_02, W_01,W_04 K_01,K_03 (40 %)</p> <p>- praca projektowa zespołowa dotycząca planu wdrożeniowego dla oprogramowania oraz doboru metodyki jego tworzenia - efekty: U_03, U_04,W_02, W_03, K_01, K_02,K_03 (60%)</p> <p>Każda z form oceny CL jest punktowana, a suma punktów możliwych do uzyskania to y. W nawiasach przy każdej z form oceniania CL podano jaki procent y można uzyskać maksymalnie z tej formy oceniania. Ocena B dla zaliczenia CL jest ustalana na podstawie uzyskanej przez studenta sumy punktów P według zasady:</p> <table border="0"> <tr> <td>P ∈ [0% y, 50% y)</td> <td>niedostateczna</td> <td></td> </tr> <tr> <td>P ∈ [50% y, 60% y)</td> <td></td> <td>dostateczna</td> </tr> <tr> <td>P ∈ [60% y, 70% y)</td> <td></td> <td>dostateczna plus</td> </tr> <tr> <td>P ∈ [70% y, 80% y)</td> <td></td> <td>dobra</td> </tr> <tr> <td>P ∈ [80% y, 90% y)</td> <td></td> <td>db plus</td> </tr> <tr> <td>P ∈ [90% y, 100% y]</td> <td></td> <td>bardzo dobra</td> </tr> </table> <p>Końcową oceną z zaliczenia przedmiotu jest ocena za ćwiczenia laboratoryjne.</p>	P ∈ [0% y, 50% y)	niedostateczna		P ∈ [50% y, 60% y)		dostateczna	P ∈ [60% y, 70% y)		dostateczna plus	P ∈ [70% y, 80% y)		dobra	P ∈ [80% y, 90% y)		db plus	P ∈ [90% y, 100% y]		bardzo dobra
P ∈ [0% y, 50% y)	niedostateczna																		
P ∈ [50% y, 60% y)		dostateczna																	
P ∈ [60% y, 70% y)		dostateczna plus																	
P ∈ [70% y, 80% y)		dobra																	
P ∈ [80% y, 90% y)		db plus																	
P ∈ [90% y, 100% y]		bardzo dobra																	

Matryca efektów kształcenia dla przedmiotu		
Numer (symbol) efektu kształcenia	Odniesienie do efektów kształcenia dla programu	Odniesienie do charakterystyki drugiego stopnia PRK dla obszaru/ obszarów
W_01	K1 _W13, K1 _W22, K1 _W24	P6S_WG, P6S_WK
W_02	K1 _W13, K1 _W22, K1 _W24	P6S_WG ,P6S_WK
W_03	K1 _W24	P6S_WG
W_04	K1 _W11, K1 _W13, K1_W24	P6S_WG ,P6S_WK
U_01	K1_U14, K1_U27	P6S_UW
U_02	K1_U28, K1_U32	P6S_UW
U_03	K1_U08, K1_U27	P6S_UW
U_04	K1_U06, K1_U08, K1_U19, K1_U31	P6S_UW

K_01	K1_K02,K1_08	P6S_KO
K_02	K1_K04, K1_K07, K1_K08	P6S_KR, P6S_KO
K_03	K1_K01,K1_08	P6S_KK, P6S_KO

Wykaz literatury

A. Literatura wymagana do ostatecznego zaliczenia zajęć (zdania egzaminu):

1. Adzic G. *Specyfikacja na przykładach. Poznaj zwinne metody pracy i dostarczaj właściwe oprogramowanie.*, Helion, Gliwice 2014
2. Sacha K. *Inżynieria oprogramowania*, PWN, Warszawa 2010
3. Shalloway A., Trott J.R. *Programowanie zorientowane obiektowo. Wzorce projektowe*. Helion, Gliwice 2005
4. Wrycza S. , Marcinkowski B., Wyrzykowski K. *Język UML 2.0 w modelowaniu systemów informatycznych, modelowanie biznesowe, metodyki projektowe oparte na UML, narzędzia CASE*, Helion, Gliwice 2005

B. Literatura uzupełniająca

1. Jaskiewicz A. *Inżynieria oprogramowania* , Helion. Gliwice 1997
2. Phillips J., *Zarządzanie projektami IT*, Wydanie III, Helion Gliwice 2011,

Kontakt

dr inż. Zbigniew Ledóchowski, dr Ryszard Motyka
 zbigniew.ledochowski@apsl.edu.pl, ryszard.motyka@apsl.edu.pl