

WYDZIAŁ PODSTAWOWYCH PROBLEMÓW TECHNIKI

KARTA PRZEDMIOTU

Nazwa w języku polskim : INŻYNIERIA OPROGRAMOWANIA
Nazwa w języku angielskim: SOFTWARE ENGINEERING
Kierunek studiów (jeśli dotyczy): INŻYNIERIA BIOMEDYCZNA
Specjalność (jeśli dotyczy): INFORMATYKA MEDYCZNA
Stopień studiów i forma: I / II stopień*, stacjonarna / niestacjonarna*
Rodzaj przedmiotu: obowiązkowy / wybieralny / ogólnouczelniany*
Kod przedmiotu INP002017W, INP002017L, INP002017P
Grupa kursów TAK / NIE*

	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium
Liczba godzin zajęć zorganizowanych w Uczelni (ZZU)	30		30	15	
Liczba godzin całkowitego nakładu pracy studenta (CNPS)	90		60	60	
Forma zaliczenia	Egzamin / zaliczenie na ocenę*	Egzamin / zaliczenie na ocenę*	Egzamin / zaliczenie na ocenę*	Egzamin / zaliczenie na ocenę*	Egzamin / zaliczenie na ocenę*
Dla grupy kursów zaznaczyć kurs końcowy (X)					
Liczba punktów ECTS	3		2	2	
w tym liczba punktów odpowiadająca zajęciom o charakterze praktycznym (P)			2	1,5	
w tym liczba punktów ECTS odpowiadająca zajęciom wymagającym bezpośredniego kontaktu (BK)	2		1,5	0.5	

WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

Student ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną obejmującą kluczowe zagadnienia z zakresu języków programowania, w tym zorientowanych obiektowo, potrafi analizować podstawowe struktury danych i algorytmy, potrafi analizować, pisać i uruchamiać praktycznie proste programy; potrafi pisać i uruchamiać programy w języku zorientowanym obiektowo, potrafi samodzielnie znajdować i usuwać błędy w tworzonych programach, potrafi budować wielopoziomą strukturę programu realizującą zadanie programistyczne;
 Zaliczone kursy: Technologie informacyjne (wykład i laboratorium), Języki programowania (wykład i laboratorium), Zaawansowane programowanie obiektowe (wykład i laboratorium).

CELE PRZEDMIOTU

- C1 Uzyskanie podstawowej wiedzy z zakresu tworzenia oprogramowania oraz kierowania projektem programistycznym
 C2 Nabycie podstawowych umiejętności w zakresie niektórych współczesnych technik projektowania oprogramowania oraz zarządzania projektem programistycznym

PRZEDMIOTOWE EFEKTY KSZTAŁCENIA

Z zakresu wiedzy:

PEK_W01 Ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną obejmującą kluczowe zagadnienia z zakresu inżynierii oprogramowania;

Z zakresu umiejętności:

PEK_U01 Potrafi specyfikować wymagania w projekcie programistyczny;

PEK_U02 Potrafi zaprojektować system informatyczny w języku UML;

PEK_U03 Potrafi korzystać ze współczesnych technik tworzenie systemów informatycznych;

PEK_U04 Potrafi sprawdzić poprawność oprogramowania;

Z zakresu kompetencji społecznych:

PEK_K01 potrafi współdziałać i współpracować w grupie przyjmując w niej różne role

PEK_K02 potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania

TREŚCI PROGRAMOWE

Forma zajęć - wykład		Liczba godzin
Wy1	Wprowadzenie Zasady zaliczenia kursu. Wprowadzenie do inżynierii oprogramowania.(IO) Podstawowe pojęcia: specyfikacja, analiza, projekt, implementacja, test, wdrożenie, pielęgnacja. Rola IO w inżynierii biomedycznej.	2
Wy2	Zasady skutecznego działania 7 nawyków skutecznego działania wg S. Coveya jako zasada efektywnej realizacji złożonych projektów informatycznych	2
Wy3	Specyfikowanie wymagań Faza strategiczna: cele, koszty, harmonogram, ryzyko. Zbieranie, analiza i negocjowanie wymagań. Wymagania funkcjonalne i pozafunkcjonalne . Opis oraz zarządzanie wymaganiami.	2
Wy4	Przypadki użycia Specyfikacja wymagań funkcjonalnych jako przypadków użycia	2
Wy5	Modelowanie i projektowanie Faza analizy i projektu. Strategie i metody budowy modelu. Zunifikowany język modelowania (UML): notacja i semantyka, perspektywy, diagramy strukturalne i behawioralne	2
Wy6	Język UML Wybrane diagramy opisu systemu w języku UML	2
Wy7	Wybrane aspekty projektowania Wzorce projektowe: założenia, podział, wybrane przykłady. Projektowanie interfejsu użytkownika.	2
Wy8	Kolokwium sprawdzające I	2
Wy9	Prowadzenie projektu programistycznego Modele tworzenia oprogramowania. Programowanie kaskadowe i zwinne.	2
Wy10	Środowiska programistyczne Zintegrowane środowisko programistyczne. Zarządzanie konfiguracją oprogramowania	2

Wy11	Sprawdzanie poprawności oprogramowania Weryfikacja i walidacja. Testy statyczne i dynamiczne. Pomiar poprawności oprogramowania. Testowanie automatyczne. Systemy odporne na błędy	2
Wy12	Wdrażanie i pielęgnacja oprogramowania Tworzenie dokumentacji technicznej. Ewolucja, pielęgnacja i refaktoryzacja	2
Wy13	Podsumowanie Spójnienie z lotu ptaka na IO. Jakość oprogramowania. Charakterystyczne cechy projektów programistycznych z zakresu Inżynierii Biomedycznej	2
Wy14	Kolokwium sprawdzające II	2
Wy15	Wybrane aspekty IO	2
	Suma godzin	30

Forma zajęć - laboratorium		Liczba godzin
La1	Wprowadzenie do laboratorium. Zasady obowiązujące na zajęciach. Wprowadzenie do narzędzi wykorzystywanych podczas laboratorium	2
La2	Praktyczna ewaluacja umiejętności. Badanie poprawności kodu.	2
La3	Specyfikowanie wymagań – przypadki użycia	4
La4	Projektowanie oprogramowania – diagramy klas i in.	6
La5	Implementacja w oparciu o wzorce projektowe	4
La6	Sprawdzanie poprawności oprogramowania	4
La7	Tworzenie dokumentacji systemu	2
La8	Refaktoryzowanie systemu	4
La9	Zaliczenie	2
	Suma godzin	30

Forma zajęć - projekt		Liczba godzin
Pr1	Wprowadzenie do projektu. Zasady obowiązujące na zajęciach.	1
Pr2	Faza strategiczna projektu	2
Pr3	Specyfikowanie wymagań – przypadki użycia	2
Pr4	Projektowanie oprogramowania – diagramy klas i in.	2
Pr5	Implementacja w oparciu o wzorce projektowe	2
Pr6	Sprawdzanie poprawności oprogramowania	2
Pr7	Tworzenie dokumentacji systemu	2
Pr8	Zaliczenie	2
	Suma godzin	15

STOSOWANE NARZĘDZIA DYDAKTYCZNE
N1. Wykład – prezentacja multimedialna wspomaganą metodą tradycyjną (tablica i pisak)
N2. Projekt – zadanie projektowe grupowe w ramach zajęć i godzin CNPS
N3. Laboratorium – listy zadań do samodzielnej realizacji w ramach zajęć i godzin CNPS
N4. Laboratorium – przykładowe zadania rozwiązywane wspólnie w ramach zajęć
N5. Laboratorium – krótkie testy sprawdzające – w formie pisemnej lub elektronicznej
N6. Laboratorium i projekt – komputer i oprogramowanie: m.in. narzędzie do modelowania UML (ArgoUML), zintegrowane środowisko programistyczne (Eclipse / NetBeans), system kontroli wersji (SVN / GitHub)

OCENA OSIĄGNIĘCIA PRZEDMIOTOWYCH EFEKTÓW KSZTAŁCENIA

Oceny (F – formująca (w trakcie semestru), P – podsumowująca (na koniec semestru))	Numer efektu kształcenia	Sposób oceny osiągnięcia efektu kształcenia
F1	PEK_W01 PEK_U01 PEK_U02	Egzamin
F2	PEK_W01 PEK_U01 PEK_U02	Kolokwia sprawdzające
F3	PEK_U01 PEK_U02 PEK_U03 PEK_U04 PEK_K01 PEK_K02	Projekt programistyczny
F4	PEK_U01 PEK_U02 PEK_U03 PEK_U04	Listy zadań do samodzielnej realizacji
F5	PEK_W01 PEK_U01 PEK_U02 PEK_U03 PEK_U04	Krótkie testy sprawdzające
P – wykład = max(F1, F2) P – projekt = F3 P – laboratorium = 0.5*F4 + 0.5*F5		

LITERATURA PODSTAWOWA I UZUPEŁNIAJĄCA

LITERATURA PODSTAWOWA:

- [1] Jaskiewicz A., Inżynieria oprogramowania, Helion 2000.
- [2] Sacha K., Inżynieria oprogramowania, PWN 2010
- [3] Somerville J., Inżynieria oprogramowania, WNT 2003.

LITERATURA UZUPEŁNIAJĄCA:

- [1] Fowler M., UML w kropelce, LTP, 2005.
- [2] Fowler M., i in., Refaktoryzacja: ulepszanie struktury istniejącego kodu, Helion 1991, 2011
- [3] Martin R.C., Czysty kod. Podręcznik dobrego programisty, Helion 2010
- [4] Pressman R.S., Praktyczne podejście do inżynierii oprogramowania, WNT 2005, 2010.

OPIEKUN PRZEDMIOTU (IMIĘ, NAZWISKO, ADRES E-MAIL)

Dr inż. Witold Dyrka witold.dyrka@pwr.wroc.pl

MACIERZ POWIĄZANIA EFEKTÓW KSZTAŁCENIA DLA PRZEDMIOTU
Inżynieria oprogramowania
Z EFEKTAMI KSZTAŁCENIA NA KIERUNKU Inżynieria Biomedyczna
I SPECJALNOŚCI Informatyka Medyczna

Przedmiotowy efekt kształcenia	Odniesienie przedmiotowego efektu do efektów kształcenia zdefiniowanych dla kierunku studiów i specjalności (o ile dotyczy)	Cele przedmiotu**	Treści programowe**	Numer narzędzia dydaktycznego**
PEK_W01 (wiedza)	K1IBM_W09_S4INM	C1, C2	Wy1 – Wy15	N1, N5
PEK_U01 (umiejętności)	K1IBM_U13_S4INM K1IBM_U16_S4INM	C1, C2	La3, La7, La9 Pr2 – Pr4, Pr6 – Pr8, Wy1 – Wy4	N1 – N5
PEK_U02	K1IBM_U15_S4INM	C1, C2	La3 – La5, La7 – La9, Pr2 – Pr5, Pr7, Pr8 Wy5 – Wy7	N1 – N6
PEK_U03	K1IBM_U16_S4INM	C1, C2	La5 – La9, Pr5 – Pr8, Wy9 – Wy11	N1 – N6
PEK_U04	K1IBM_U13_S4INM, K1IBM_U16_S4INM	C1, C2	La2, La6 – La9, Pr6 – Pr8 Wy12 – Wy13	N1 – N6
PEK_K01 (kompetencje)	K1IBM_K03	C1, C2	La1 – La8, Pr1 – Pr8	N2, N4, N6
PEK_K02	K1IBM_K04	C1, C2	La1 – La9, Pr1 – Pr8	N1 – N4