

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Inżynieria oprogramowania		Kod 1010331451010330109
Kierunek studiów Informatyka	Profil kształcenia (ogólnoakademicki, praktyczny) (brak)	Rok / Semestr 3 / 5
Ścieżka obieralności/specjalność -	Przedmiot oferowany w języku: polski	Kurs (obligatoryjny/obieralny) obligatoryjny
Stopień studiów: I stopień	Forma studiów (stacjonarna/niestacjonarna) stacjonarna	
Godziny Wykłady: 1 Ćwiczenia: - Laboratoria: 1 Projekty/seminaria: -		Liczba punktów 3
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) (brak)		(ogólnouczelniany, z innego kierunku) (brak)
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne		Podział ECTS (liczba i %) 3 100%
Odpowiedzialny za przedmiot / wykładowca:		
Dr hab. inż. Barbara Begier email: barbara.begier@put.poznan.pl tel. 61 665-3724 Elektryczny ul. Piotrowo 3A, 60-965 Poznań		
Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:		
1	Wiedza:	K_W00: ma podstawową wiedzę wynikającą z programu szkoły średniej K_W05: ma uporządkowaną i podbudowaną teoretycznie wiedzę w zakresie podstawowych konstrukcji programistycznych, implementacji algorytmów, paradygmatów i stylów programowania, metod weryfikacji poprawności programów, języków formalnych, kompilatorów, platform.
2	Umiejętności:	K_U01: potrafi pozyskiwać informacje z literatury, baz danych i innych źródeł; potrafi integrować uzyskane informacje, dokonywać ich interpretacji, a także wyciągać wnioski oraz formułować i uzasadniać opinie.
3	Kompetencje społeczne	K_K02: ma świadomość ważności i rozumie pozatechniczne aspekty i skutki działalności inżyniera-informatyka oraz związaną z tym odpowiedzialność za podejmowane decyzje.
Cel przedmiotu:		
Celem przedmiotu obejmującego dwa semestry jest zapoznanie studentów z inżynierskim podejściem do wytwarzania oprogramowania. Podczas pierwszego semestru tego przedmiotu celem jest poznanie zasad modelowania obiektowego zgodnie ze standardem UML 2.0, dokonanie przeglądu modeli cyklu rozwojowego oprogramowania oraz procesów podstawowych i wspomagających w produkcji oprogramowania.		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza:		
1. Ma uporządkowaną i podbudowaną metodologicznie wiedzę w zakresie inżynierii oprogramowania. - [K_W12] 2. Orientuje się w obecnym stanie oraz najnowszych trendach rozwojowych informatyki. - [K_W19]		
Umiejętności:		
1. Potrafi sformułować wymagania, opracować model obiektowy oraz ocenić prosty system informatyczny, uwzględniając realizowane funkcje i powiązania między elementami składowymi. - [K_U16] 2. Potrafi przygotować i przedstawić krótką prezentację poświęconą wynikom realizacji zadania inżynierskiego. - [K_U04]		
Kompetencje społeczne:		
1. Ma świadomość ważności i rozumie pozatechniczne aspekty i skutki działalności inżyniera-informatyka i związaną z tym odpowiedzialność za podejmowane decyzje. - [K_K02] 2. Ma świadomość odpowiedzialności za pracę własną oraz gotowość podporządkowania się zasadom pracy w zespole i ponoszenia odpowiedzialności za wspólnie realizowane zadania. - [K_K04]		
Sposoby sprawdzenia efektów kształcenia		

<p>Treści prezentowane podczas wykładu są przedmiotem egzaminu w następnym semestrze. Po serii wykładów w bieżącym semestrze ocena zostanie wystawiona na podstawie oceny aktywności studenta na zajęciach oraz sprawdzianu.</p> <p>Zaliczenie zajęć laboratoryjnych odbywa się na podstawie ocen cząstkowych wystawianych oddzielnie za każdy opracowany dokument oraz diagram UML.</p>		
Treści programowe		
<p>Wykład. Pojęcie inżynierii. Projektowanie oprogramowania oparte na modelu, zgodnie z koncepcją MDA. Dokument wymagań. Standard UML ? geneza, założenia, widoki i diagramy. Procesy wytwarzania oprogramowania i przegląd modeli cyklu rozwojowego oprogramowania. Cechy charakterystyczne metodyk twardej wytwarzania oprogramowania. Wymagane procesy wspomagające, w tym dokumentowanie. Podejście procesowe wg ISO 9000 odniesione do wytwarzania oprogramowania. Przeglądy i inspekcje wyrobu programowego. Model CMM oceny dojrzałości procesu wytwórczego oprogramowania.</p> <p>Laboratoria. Uściślanie wymagań, opracowanie modelu obiektowego tworzonego oprogramowania z użyciem standardu UML 2.0 (przypadki użycia, diagram obiektów i diagram klas).</p>		
Literatura podstawowa:		
<ol style="list-style-type: none"> Booch G., Jacobson I., Rumbaugh J., UML. Przewodnik użytkownika, WNT, Warszawa 2001 (i dalsze wydania). Wrycza St., Marcinkowski B., Wyrzykowski K., Język UML 2.0 w modelowaniu systemów informatycznych, Helion, Gliwice 2005. 		
Literatura uzupełniająca:		
<ol style="list-style-type: none"> Begier B., Inżynieria oprogramowania - problematyka jakości, Wydawnictwo Politechniki Pozn., Poznań 1999. Hamlet D., Maybee J., Podstawy techniczne inżynierii oprogramowania, WNT, Warszawa 2003. Metody wytwarzania oprogramowania, (red. St. Szejko), MIKOM, Warszawa 2002. Pressman R. S., Praktyczne podejście do inżynierii oprogramowania, WNT, Warszawa 2004. Pilone D., Pitman N., UML 2.0. Almanach, Helion, Gliwice 2007. 		
Bilans nakładu pracy przeciętnego studenta		
Czynność		Czas (godz.)
1. Uczestnictwo w wykładach		15
2. Uczestnictwo w zajęciach laboratoryjnych		15
3. Przygotowanie modelu obiektowego (diagramów), przygotowanie do zaliczenia pierwszej części przedmiotu		30
4. Konsultacje, zaliczanie		10
Obciążenie pracą studenta		
forma aktywności	godzin	ECTS
Łączny nakład pracy	70	3
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	45	1
Zajęcia o charakterze praktycznym	15	1