



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez
Unię Europejską w ramach
Europejskiego Funduszu
Społecznego

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Nazwa przedmiotu		Kod ECTS					
Analiza i projektowanie systemów informatycznych		11.3.1313					
Nazwa jednostki prowadzącej przedmiot							
Katedra Informatyki Ekonomicznej							
Studia							
wydział	kierunek	poziom	pierwszego stopnia				
Wydział Matematyki, Fizyki i Informatyki	Informatyka	forma	stacjonarne				
		moduł	wszystkie				
		specjalnościowy	wszystkie				
		specjalizacja	wszystkie				
Nazwisko osoby prowadzącej (osób prowadzących)							
prof. UG, dr hab. Bartosz Marcinkowski							
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS					
Formy zajęć		3					
Wykład, Ćw. laboratoryjne							
Sposób realizacji zajęć							
zajęcia w sali dydaktycznej							
Liczba godzin							
Wykład: 15 godz., Ćw. laboratoryjne: 30 godz.							
Termin realizacji przedmiotu							
2022/2023 zimowy							
Status przedmiotu		Język wykładowy					
obowiązkowy		polski					
Metody dydaktyczne		Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne					
<ul style="list-style-type: none"> - Projektowanie doświadczeń - Wykład problemowy - ćwiczenia laboratoryjne -- sporządzanie i uruchamianie programów komputerowych 		Sposób zaliczenia					
		<ul style="list-style-type: none"> - Zaliczenie na ocenę - Egzamin 					
		Formy zaliczenia					
		<ul style="list-style-type: none"> - egzamin pisemny z pytaniami (zadaniami) otwartymi - ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru - kolokwium 					
Podstawowe kryteria oceny							
Sposób weryfikacji założonych efektów uczenia się							
zakładany efekt kształcenia	egzamin	kolokwium	projekt	referat	raport	aktywność	obserwacja postawy i umiejętności
	Wiedza						
K_W03	X	X					
	Umiejętności						
K_U03							X
	Kompetencje						
K_K03							X
K_K04							X
Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi							

<p>A. Wymagania formalne Brak wymagań formalnych</p>	
<p>B. Wymagania wstępne Brak wymagań wstępnych</p>	
<p>Cele kształcenia</p> <p>Opanowanie wiedzy i umiejętności z zakresu:</p> <ul style="list-style-type: none"> • podstaw metodologicznych tworzenia systemów informatycznych, • strukturalnych, obiektowych, społecznych i adaptacyjnych metodyk tworzenia systemów informatycznych, • liniowego, spiralnego i przyrostowo-iteracyjnego cyklu życia systemu, • metod i technik projektowania systemów informatycznych, • pakietów CASE (Computer Aided Software Engineering). 	
<p>Treści programowe</p> <ul style="list-style-type: none"> • Procesy analizy w kontekście inżynierii oprogramowania (fazy cyklu życia systemu, analiza funkcjonalna, niefunkcjonalna, relacja klient-dostawca, metodologiczne podstawy tworzenia systemów informatycznych • Budowa i rodzaje systemów informatycznych (rodzaje cykli życia systemu, budowa modułowa, architektura SOA, klasy systemów wykorzystywanych w biznesie) • Podstawy analizy strukturalnej (cele, znaczenie i założenia analizy strukturalnej, stosowane notacje w fazie analizy) • Podstawy analizy obiektowej (cele i znaczenie analizy obiektowej, procesy analizy obiektowej) • Modelowanie w analizie obiektowej (stosowane notacje, techniki i narzędzia, UML, BPMN, SysML) • Przykłady technologicznego wspierania analizy strukturalnej i obiektowej, wspomaganie analizy z wykorzystaniem systemów CASE (Computer Aided System Engineering) • Projektowanie interfejsów. Procesy projektowania formularzy i raportów. • Projektowanie okien dialogowych i sekwencji dialogowych • Zastosowanie przewodników i list kontrolnych w procesach projektowania • Diagramy przypadków użycia – identyfikacja PU, zaawansowana specyfikacja związków, wprowadzanie stereotypów do modelu, zarządzanie złożonością rozbudowanych modeli przypadków użycia z wykorzystaniem pakietów • Diagramy klas – modelowanie struktury danych w systemie, wdrażanie modelu danych, egzemplifikacja struktury danych z wykorzystaniem diagramów obiektów • Diagramy maszyn stanowych – śledzenie stanów obiektów w systemie, zagnieżdżanie maszyn stanowych, pseudostany • Modelowanie infrastruktury sprzętowej i osadzanie komponentów programowych z wykorzystaniem diagramów komponentów oraz rozlokowania UML • Generowanie kodu programu i jego analiza. Inżynieria w przód i w tył. Wzorce projektowe i ich dokumentacja. 	
<p>Wykaz literatury</p> <p>A. Literatura wymagana do ostatecznego zaliczenia zajęć (zdania egzaminu):</p> <ol style="list-style-type: none"> 1. Wrycza S., Marcinkowski B., Wyrzykowski K., Język UML 2.0 w modelowaniu systemów informatycznych, Helion, Gliwice 2005 2. Wrycza S., Marcinkowski B., Maślankowski J., UML 2.x. Ćwiczenia zaawansowane, Helion, Gliwice 2012 3. Valacich J., George J., Hoffer J., Essential of System Analysis and Design, Fifth Edition, Prentice Hall, 2012 <p>B. Literatura uzupełniająca</p> <ol style="list-style-type: none"> 1. Booch G., Rumbaugh J., Jacobson I., The Unified Modeling Language User Guide, Second Edition, Addison-Wesley, Boston 2005 2. Booch G., Maksimchuk R., Engle M., Young B., Conallen J., Houston L., Object-Oriented Analysis and Design with Applications (3rd Edition), Addison Wesley, 2007 	
<p>Kierunkowe efekty uczenia się</p> <p>K_W03: ma uporządkowaną wiedzę w zakresie inżynierii oprogramowania i metodyk zarządzania projektami informatycznymi, cyklu życia projektu informatycznego, specyfikacji, walidacji i weryfikacji oprogramowania, wzorców projektowych</p> <p>K_U03: potrafi pracować w zespole informatyków, zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminy, porozumiewać się przy użyciu różnych technik w środowisku zawodowym w tym z wykorzystaniem dedykowanych narzędzi</p> <p>K_K03: rozumie potrzebę i docenia zalety pracy zespołowej, rozumie konieczność systematycznej pracy nad zespołowymi projektami informatycznymi</p> <p>K_K04: rozumie i docenia znaczenie uczciwości intelektualnej w działaniach własnych i innych osób;</p>	<p>Wiedza</p> <p>ma uporządkowaną wiedzę w zakresie inżynierii oprogramowania i metodyk zarządzania projektami informatycznymi, cyklu życia projektu informatycznego, specyfikacji, walidacji i weryfikacji oprogramowania, wzorców projektowych definiuje podstawowe pojęcia i kategorie współczesnej metodologii tworzenia systemów informatycznych, wymienia rodzaje metodyk i narzędzi tworzenia systemów informatycznych, wyjaśnia procesy skutecznego projektowania i wdrażania projektów informatycznych</p> <p>Umiejętności</p> <p>rozwiązuje problemy, pozwalające na realizację potrzeb i celów informatycznych użytkowników planuje strategię informatyzacji firm i instytucji, analizuje obszar zastosowań pod kątem definiowania założeń i koncepcji systemów informatycznych, ustala kryteria i ocenia metodyki, metody, techniki i narzędzia tworzenia systemów informatycznych (TSI) pod kątem ich efektywnego doboru do specyfiki użytkownika, konstruuje funkcjonujące w praktyce rozwiązania informatyczne.</p>

postępuje etycznie	Kompetencje społeczne (postawy)
Kontakt	
Bartosz.Marcinkowski@ug.edu.pl	