

Opis przedmiotu	
<b>GROUP PROJECT</b>	
Kod przedmiotu	1120-IN000-ISA-0473
Nazwa przedmiotu	Group project
Nazwa przedmiotu w jęz. polskim	Projekt zespołowy
<b>A. Usytuowanie przedmiotu w systemie studiów</b>	
Poziom kształcenia	Studia pierwszego stopnia
Forma i tryb prowadzenia studiów	Stacjonarne
Kierunek studiów	Informatyka i Systemy Informacyjne
Profil studiów	Profil ogólnoakademicki
Specjalność	-
Jednostka prowadząca	Wydział Matematyki i Nauk Informacyjnych
Jednostka realizująca	Wydział Matematyki i Nauk Informacyjnych
Koordynator przedmiotu	Dr inż. Krzysztof Kaczmarek
<b>B. Ogólna charakterystyka przedmiotu</b>	
Blok przedmiotów	Kierunkowe
Grupa przedmiotów	Obowiązkowe
Status przedmiotu	Obowiązkowy
Język prowadzenia zajęć	Angielski
Semestr nominalny	7
Usytuowanie realizacji w roku akademickim	Semestr zimowy
Wymagania wstępne/przedmioty poprzedzające	Software engineering Programming 1, 2 i 3, Programming in graphical environment Multilayer application development
Limit liczby studentów	Laboratoria (ćwiczenia komputerowe) – 15-24 os/grupa
<b>C. Efekty kształcenia i sposób prowadzenia zajęć</b>	
Cel przedmiotu	<p>Celem przedmiotu jest utrwalenie zasad przemysłowego tworzenia aplikacji (praca zespołowa), pomoc przy tworzeniu projektu dyplomowego oraz do-skonalenie umiejętności tworzenia oprogramowania. Po ukończeniu kursu studenci powinni:</p> <ul style="list-style-type: none"> <li>- posiadać wiedzę wystarczającą do tworzenia prostych aplikacji (do 2 000 linii kodu) w niedużym zespole (2 – 3 osoby);</li> <li>- powinni umieć wybrać odpowiedni model tworzenie aplikacji;</li> <li>- umieć - w ramach zespołu - dokonać podziału zadań na poszczególne oso-by;</li> <li>- umieć stworzyć harmonogram realizacji pracy;</li> <li>- umieć napisać i przetestować stworzoną przez siebie aplikację;</li> <li>- mieć przygotowaną (w 90%) aplikację będąca podstawą inżynierskiego projektu dyplomowego</li> </ul> <p><i>The aim of the course is to consolidate the principles of industrial application development (teamwork), help in creating a diploma project and improve software development skills. Upon completion of the course, students should:</i></p> <ul style="list-style-type: none"> <li>- have sufficient knowledge to create simple applications (up to 2000 lines of code) in a small team (2 - 3 people);</li> <li>- should be able to choose the right application development model;</li> </ul>

	<p>- <i>be able - within a team - to divide tasks between individuals;</i>  - <i>be able to create a work schedule;</i>  - <i>be able to write and test an application created by you;</i>  - <i>have prepared (in 90%) an application that is the basis for an engineering diploma project.</i></p>	
Efekty kształcenia	Patrz TABELA 38.	
Formy zajęć i ich wymiar	Wykład	15
	Ćwiczenia	0
	Laboratorium	0
	Projekt	15
Treści kształcenia	<p>Wykład:  Zasady przemysłowego tworzenia systemów komputerowych.  Przeprowadzanie testów jednostkowych i testów integracyjnych.  Tworzenie zestawu dokumentacji: harmonogram prac, specyfikacja wymagań (dokumentacja biznesowa), analiza ryzyka, dokumentacja architektoniczna, dokumentacja techniczna, plan testów akceptacyjnych, instrukcja obsługi i rejestr zmian. Podstawy projektowania interfejsu użytkownika.  Zastosowanie złożonego systemu składu tekstu do tworzenia profesjonalnych dokumentów.  Projekt:  Doskonalenie umiejętności tworzenia oprogramowania, podczas tworzenia aplikacji w niewielkim zespole (2 – 3 osoby). Przygotowanie dokumentacji dla przeprowadzonego procesu wytwarzania oprogramowania.</p> <p><i>Lecture:</i>  <i>Principles of industrial development of computer systems. Conducting unit tests and integration tests.</i>  <i>Creating a set of documentation: work schedule, requirements specification (business documentation), risk analysis, architectural documentation, technical documentation, acceptance test plan, user manual and change register. Basics of user interface design.</i>  <i>Use of a complex typesetting system to create professional documents.</i>  <i>Project:</i>  <i>Improving software development skills while creating applications in a small team (2-3 people). Preparation of documentation for the software development process.</i></p>	
Metody dydaktyczne	<p>Wykład: wykład informacyjny  Projekt: zespołowe rozwiązywanie problemów</p>	
Metody oceny	<p>Test na ostatnim wykładzie – od 0 do 30 pkt. (część A), zajęcia projektowe – 8 pkt za jeden etap, 5 etapów, razem od 0 do 40 pkt (część B), przygotowanie aplikacji w terminie (początek stycznia) – 0, 20, 25 lub 30 pkt, po terminie 0 pkt (część C). Studenci mają możliwość oddania każdej części projektu z ty-godniowym opóźnieniem. Jedno opóźnienie bez konsekwencji za każdym następnym otrzymują odpowiednio mniej punktów.  Ocena końcowa wynika z sumy A + B + C: 51-60 pkt – dostateczny, 61-70 pkt – trzy i pół, 71-80 pkt – dobry, 81-90 pkt – cztery i pół, od 91 pkt – bardzo dobry.</p> <p><i>Test at the last lecture - from 0 to 30 points. (part A), project classes - 8</i></p>	

	<p><i>points for one stage, 5 stages, a total of 0 to 40 points (part B), preparation of the application on time (beginning of January) - 0, 20, 25 or 30 points, 0 points after the deadline (part C). Students have the option of handing over any part of the project with a week's delay. One delay without consequences, each subsequent delay receives correspondingly fewer points.</i></p> <p><i>The final grade results from the sum of A + B + C: 51-60 points - satisfactory, 61-70 points - three and a half, 71-80 points - good, 81-90 points - four and a half, from 91 points - very good.</i></p>		
Metody sprawdzania efektów kształcenia	Patrz TABELA 1.		
Egzamin	Nie		
Literatura	<p>1. R. Pressman, Software Engineering, McGraw-Hill, 2004</p> <p>2. Software Developer's Journal</p> <p>Further literature/sources are chosen individually and depend on the topic of particular project.</p>		
Witryna www przedmiotu			
<b>D. Nakład pracy studenta</b>			
Liczba punktów ECTS	3		
Liczba godzin pracy studenta związanych z osiągnięciem efektów kształcenia	<p>1. godziny kontaktowe – 35 h; w tym</p> <p style="margin-left: 20px;">a) obecność na wykładach – 15 h</p> <p style="margin-left: 20px;">b) obecność na zajęciach projektowych – 15 h</p> <p style="margin-left: 20px;">c) konsultacje – 5 h</p> <p>2. praca własna studenta – 55 h; w tym</p> <p style="margin-left: 20px;">a) przygotowanie do zajęć projektowych – 10 h,</p> <p style="margin-left: 20px;">b) przygotowanie aplikacji, uruchomienie, testowanie – 25 h</p> <p style="margin-left: 20px;">c) przygotowanie dokumentacji – 20 h</p> <p>Razem 90 h, co odpowiada 3 pkt. ECTS</p>		
Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:	<p>1. obecność na wykładach – 15 h</p> <p>2. obecność na zajęciach projektowych – 15 h</p> <p>3. konsultacje – 5 h</p> <p>Razem 35 h, co odpowiada 2 pkt. ECTS</p>		
Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	<p>1. obecność na zajęciach projektowych – 15 h</p> <p>2. przygotowanie do zajęć projektowych – 10 h</p> <p>3. przygotowanie aplikacji, uruchomienie, testowanie – 25 h</p> <p>4. przygotowanie dokumentacji – 20 h</p> <p>Razem 70 h, co odpowiada 3 pkt. ECTS</p>		
<b>E. Informacje dodatkowe</b>			
Uwagi	-		
<b>TABELA 1 LEARNING OUTCOMES OF THE COURSE</b>			
1. Learning outcomes and their reference to the learning outcomes in the area of technical sciences and for the field of study Computer Science and Information Systems			
Learning outcomes of the module	DESCRIPTION OF THE LEARNING OUTCOMES A graduate of first level studies in the field <i>Computer Science and Information Systems</i> :	Odniesienie do efektów uczenia się dla kierunków	Sposób weryfikacji  Verification method
<b>KNOWLEDGE</b>			
W01	has a basic knowledge of the life cycle of computer systems;	I1A_W11	test project

W02	knows the basic methods, techniques and tools used to solve simple computer tasks;	I1A_W03 I1A_W04 I1A_W07 I1A_W09 I1A_W10 I1A_W11	project
W03	has a basic knowledge of quality management, including basic knowledge of standards	I1A_W11	test
SKILLS			
U01	can create a model of a simple system;	I1A_U02 I1A_U13	project
U02	is able to design simple computer systems;	I1A_U12	project
U03	can, according to given specifications, design and implement a simple computer system, using appropriate methods, techniques, and tools;	I1A_U11 I1A_U12	project
U04	is able to perform simple analysis of functioning of a computer system and evaluate the existing solutions, at least their functionality;	I1A_U06	test project
U05	can perform functional tests;	I1A_U13	test project
SOCIAL COMPETENCIES			
K01	is able to work individually and as a member of a programming team, can manage his/her time, make commitments, and meet deadlines;	I1A_K06	project
K02	knows examples and understands the cause of malfunctioning systems, which have led to serious financial or social losses or to a serious loss of health and even life.	I1A_K03	test