Nazwa przedmiotu						
Modelowanie i analiza systemów informatycznych Modeling and analysis of information systems.						
Dyscyplina					Oznaczenie przedmiotu	
Informatyka					1K_IS2	
Rodzaj przedmiotu	Stopień studiów	Tryb studiów	Język zajęć		Rok	Semestr
obowiązkowy	2	stacjonarne	polski		I	I
Rodzaj zajęć		Wyk.	Ćw.	Lab.	Sem.	Proj.	Liczba punktów ECTS
Liczba godzin w semestrze		30	0	15	0	0	5 ECTS
Koordynator	dr inż. Łukasz Piątek l_piatek@el.pcz.czest.pl						
Prowadzący	dr inż. Łukasz Piątek l_piatek@el.pcz.czest.pl dr inż. Piotr Szeląg szelag@el.pcz.czest.pl mgr inż. Marcjan Nowak m.nowak@el.pcz.czest.pl						

I. KARTA PRZEDMIOTU

Cel przedmiotu

C1. Zapoznanie studenta z cechami systemów informatycznych.

C2. Zapoznanie studenta z wybranymi metodami modelowania.

C3. Zapoznanie studenta z etapami cyklu życia systemów informatycznych.

Wymagania wstępne w zakresie wiedzy, umiejętności i innych kompetencji

1. Wiedza z inżynierii oprogramowania z zakresu studiów I stopnia kierunku Informatyka

Efekty kształcenia

EK1. Student potrafi analizować cechy systemów informatycznych.
EK2. Student potrafi posługiwać się modelami systemów informatycznych implementowanymi w języku UML
EK3. Student ma wiedzę na temat cyklu życia systemów informatycznych.

Treści programowe: wykłady	Liczba godzin
W 1 – Wprowadzenie. Cechy systemów informatycznych i związanych z nimi artefaktów. Cykl życia systemów informatycznych. Przegląd modeli procesów wytwarzania oprogramowania. Modelowanie obiektowe i strukturalne	2
W 2 – Modelowanie wymagań, Modele przypadków użycia.	2
W 3 – Język UML. Modelowanie przepływu czynności,	2
W 4 – Język UML. Modelowanie struktury logicznej systemu	2
W 5 – Język UML. Modelowanie uporządkowanej interakcji	2
W 6 – Język UML. Modelowanie połączeń. Modelowanie harmonogramowania interakcji	2
W 7 – Język UML. Zarządzanie częściami systemu	2
W 8 – Język UML. Modelowanie stanu obiektów	2
W 9 – Język UML. Modelowanie wdrożenia systemu	2
W 10 – Klasyczne wzorce projektowe.	2
W 11 - Iteracyjny model wytwarzania aplikacji	2
W 12 – Zapewnienie jakości w procesie wytwarzania oprogramowania.	2
W 13 – Testowanie systemów informatycznych	3
W 14 – Warstwowa budowa systemów informatycznych	2
Test zaliczeniowy	1
SUMA	**30**

Treści programowe: laboratorium	Liczba godzin
L 1 - Język UML. Modelowanie wymagań.	**1**
L 2 – Język UML. Modelowanie przepływu czynności przy pomocy diagramów aktywności	**1**
L 3 – Język UML. Modelowanie struktury logicznej systemu przy pomocy klas i ich diagramów	**2**
L 4 – Język UML. Modelowanie uporządkowanej interakcji przy pomocy diagramów interakcji	**2**
L 5 – Język UML. Modelowanie połączeń przy pomocy diagramów komunikacji.	**1**
L 6 – Język UML. Modelowanie harmonogramowania interakcji przy pomocy diagramów czasowych	**2**
L 7 – Język UML. Zarządzanie częściami systemu przy pomocy diagramów komponentów,	**2**
L 8 – Język UML. Modelowanie stanu obiektów za pomocą diagramów maszyny stanowej	**2**

L 9 – Język UML. Modelowanie wdrożenia systemu przy pomocy diagramów wdrożenia	**1**
Test zaliczeniowy	**1**
SUMA	**15**

Narzędzia dydaktyczne

1. Prezentacja multimedialna
2. Laboratorium – praca samodzielna/programowanie

Sposoby oceny efektów kształcenia (F – ocena Formująca, P – ocena Podsumowująca)

F1. Test z zakresu wymagań wstępnych przedmiotu Języki skryptowe. Test nie ma wpływu na ocenę końcową z przedmiotu.
P1. Wykład – pisemny test zaliczeniowy. Ocena z testu stanowi 100% końcowej oceny z wykładu
P2. Laboratorium : 50% punktów oceny końcowej przyznawane za realizacje zadań podstawowych i dodatkowych w trakcie zajęć laboratoryjnych
P3. Laboratorium : 50% punktów oceny końcowej przyznawane na podstawie rezultatów pisemnego testu zaliczeniowego

Obciążenie pracą studenta

Forma aktywności	Średnia liczba godzin na zrealizowanie aktywności
Godziny kontaktowe z prowadzącym	45
Zapoznanie się ze wskazaną literaturą	25
Przygotowanie do zajęć	30
Przygotowanie do testu / kolokwium / egzaminu	20
Przygotowanie sprawozdań/prezentacji	5
Sumaryczna liczba godzin/punktów ECTS dla przedmiotu	**125 / 5 ECTS**

Wykaz literatury podstawowej i uzupełniającej

1. Bernd Bruegge, Allen H. Dutoit, Inżynieria oprogramowania w ujęciu obiektowym. UML, wzorce projektowe i Java, Wydawnictwo Helion, Gliwice 2011
2. Craig Larman, UML i wzorce projektowe. Analiza i projektowanie obiektowe oraz iteracyjny model wytwarzania aplikacji. Wydanie III, Wydawnictwo Helion, Gliwice 2011
3. Russ Miles, Kim Hamilton, UML 2.0. Wprowadzenie, Wydawnictwo Helion, Gliwice 2011
4. Len Bass, Paul Clements, Rick Kazman: Architektura oprogramowania w praktyce, Wydawnictwo Helion, Gliwice 2011

Macierz realizacji efektów kształcenia

Efekt kształcenia	Odniesienie efektu do efektów kształcenia dla dyscypliny naukowej Informatyka	Cele przedmiotu	Forma zajęć	Narzędzia dydaktyczne	Sposób oceny
EK1	KINF1_U17	C1,C3	W, Lab	1,2	P1, P2, P3
EK2	KINF1_U17	C2,C3	W, Lab	1,2	P1, P2, P3
EK3	KINF1_U17	C3	W, Lab	1,2	P1, P2, P3

* – wg załącznika

II. FORMY OCENY – SZCZEGÓŁY

Ocena	Efekty
EK1	**Student potrafi analizować cechy systemów informatycznych**
2	Student nie potrafi analizować cech systemów informatycznych
3	Student potrafi analizować cechy prostych systemów informatycznych przy pomocy modeli obiektowych
3.5	Student potrafi analizować cechy złożonych systemów informatycznych przy pomocy modeli obiektowych
4	Student potrafi analizować cechy prostych systemów informatycznych przy pomocy modeli strukturalnych
4.5	Student potrafi analizować cechy średnio złożonych systemów informatycznych przy pomocy modeli strukturalnych
5	Student potrafi analizować cechy złożonych systemów informatycznych przy pomocy modeli strukturalnych
EK2	**Student potrafi posługiwać się modelami systemów informatycznych implementowanymi w języku UML**
2	Student nie potrafi posługiwać się modelami systemów informatycznych implementowanymi w języku UML
3	Student zna zasady modelowania wymagań, modelowania przepływu czynności
3.5	Student zna zasady modelowania struktury logicznej systemu i uporządkowanej interakcji
4	Student zna zasady modelowania połączeń
4.5	Student zna zasady modelowania harmonogramowania interakcji
5	Student zna zasady modelowania stanu obiektów
EK3	**Student ma wiedzę na temat cyklu życia systemów informatycznych**
2	Student nie ma wiedzy na temat cyklu życia systemów informatycznych
3	Student potrafi wymienić etapy cyklu życia systemów informatycznych
3.5	Student potrafi scharakteryzować etapy cyklu życia systemów informatycznych dotyczące określenia założeń i celów systemu
4	Student potrafi scharakteryzować etapy cyklu życia systemów informatycznych dotyczące projektowania
4.5	Student potrafi scharakteryzować etapy cyklu życia systemów informatycznych dotyczące wdrożenia
5	Student potrafi scharakteryzować etapy cyklu życia systemów informatycznych dotyczące eksploatacji oraz oceny efektów wdrożenia

III. INNE PRZYDATNE INFORMACJE O PRZEDMIOCIE

1. Wszelkie informacje dla studentów na temat planu zajęć dostępne są na tablicy ogłoszeń oraz na stronie el.pcz.pl.
2. Informacje na temat warunków zaliczania zajęć przekazywana jest studentom podczas pierwszych zajęć.