

1. Nazwa przedmiotu: INŻYNIERIA OPROGRAMOWANIA		2. Kod przedmiotu: IO		
3. Karta przedmiotu ważna od roku akademickiego: 2019/20				
4. Forma kształcenia: studia pierwszego stopnia				
5. Forma studiów: studia stacjonarne				
6. Kierunek studiów: INFORMATYKA (SYMBOL WYDZIAŁU) RMS				
7. Profil studiów: praktyczny				
8. Specjalność: WSZYSTKIE				
9. Semestr: V				
10. Jednostka prowadząca przedmiot: Instytut Matematyki				
11. Prowadzący przedmiot: dr inż. Zdzisław Sroczyński				
12. Przynależność do grupy przedmiotów: Wybrane działy informatyki (WDI)				
13. Status przedmiotu: obowiązkowy				
14. Język prowadzenia zajęć: polski				
15. Przedmioty wprowadzające oraz wymagania wstępne: Programowanie I-III, Algorytmy i struktury danych, Programowanie obiektowe i graficzne, Bazy danych				
16. Cel przedmiotu: Nabycie umiejętności projektowania systemów informatycznych, doboru języków i bibliotek programistycznych, technik i narzędzi oraz poznanie metod uruchamiania i testowania oprogramowania.				
17. Efekty kształcenia				
Student który zaliczy przedmiot:				
Nr	Opis efektu kształcenia	Metoda sprawdzenia efektu kształcenia	Forma prowadzenia zajęć	Odniesienie do efektów dla kierunku studiów
1	potrafi zaprojektować z uwzględnieniem wiedzy o cyklu życia złożonych systemów informatycznych, zrealizować i udokumentować system informatyczny posługując się właściwie dobraną metodyką i odpowiednimi narzędziami	proj	wykład, projekt	K1P_W15, K1P_W17 K1P_U35
2	potrafi wykorzystywać narzędzia i systemy wspomagające zarządzanie zespołowym projektem informatycznym, w tym systemy kontroli wersji	proj	projekt	K1P_U13, K1P_U29, K1P_K04
3	potrafi dobrać właściwe wzorce architektoniczne oprogramowania, w szczególności stosując zasady projektowania obiektowego	proj	wykład, projekt	K1P_U19
4	potrafi przeprowadzać systematyczne testy oprogramowania z wykorzystaniem właściwie dobranych przypadków i danych testowych	proj	wykład, projekt	K1P_U27

5	zna i potrafi zastosować metody szacowania złożoności i prędkości produkcji oprogramowania w celu uzyskania systemów o właściwej wydajności i odpowiednim zapotrzebowaniu na pamięć	proj	wykład, projekt	K1P_U32
6	potrafi identyfikować czynniki społeczne, prawne i organizacyjne wpływające na jakość i bezpieczeństwo projektowanego systemu	proj	wykład, projekt	K1P_U11, K1P_U16, K1P_K02

18. Formy zajęć dydaktycznych i ich wymiar (liczba godzin)

Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium
30			75	

19. Treści kształcenia:

Wykład: Projektowanie oprogramowania. Modele cyklu życia oprogramowania. Projektowanie niezawodnego oprogramowania, programowanie zwinne (agile) oraz ekstremalne. Specyfikacja wymagań. Koncepty tworzenia złożonego oprogramowania - podejście strukturalne i obiektowe. Korzystanie i projektowanie API (Application Programming Interface). Narzędzia i środowiska wytwarzania oprogramowania. Modelowanie. UML. Narzędzia CASE. Dobre praktyki notacyjne, przenośność kodu źródłowego. Uruchamianie i testowanie oprogramowania. Refaktoring kodu. Zarządzanie projektem informatycznym. Systemy pracy grupowej, hostingowe, systemy kontroli wersji. Projektowanie interakcji człowiek-komputer: komunikacja program-użytkownik. Dokumentacja procesu tworzenia oprogramowania oraz dokumentacja użytkowa. Optymalizacja czasowa i pamięciowa programów. Przygotowywanie aplikacji do wykorzystania przez osoby niepełnosprawne. Wdrożenie, utrzymanie i konserwacja oprogramowania. Ocena jakości i złożoności oprogramowania (kryteria, metryki). Wzorce projektowe. Standardy i metodyki zarządzania jakością.

Projekt: Założenia projektu zespołowego, oszacowanie prędkości zadań. Dobór narzędzi, przygotowanie środowiska pracy. Wirtualizacja systemu operacyjnego, uruchamianie aplikacji. Przeprowadzanie systematycznych testów. Dobór struktur danych, przygotowanie modelu obiektowego. Ocena i poprawa efektywności oprogramowania, zastosowanie właściwych algorytmów bibliotecznych. Komunikacja programów z użytkownikiem. Zarządzanie projektem informatycznym, notacje wspomagające modelowanie. Tworzenie dokumentacji użytkowej oprogramowania. Narzędzia pracy grupowej, systemy hostingowe dla projektów informatycznych, wykorzystanie systemu kontroli wersji.

20. Egzamin: nie

21. Literatura podstawowa:

1. K. Sacha: „Inżynieria oprogramowania”, PWN, 2010.
2. J. Nielsen, R. Budi: „Funkcjonalność aplikacji mobilnych. Nowoczesne standardy UX i UI”. Helion 2013.
3. K. Kaczor: „SCRUM i nie tylko. Teoria i praktyka w metodach Agile”, PWN 2014.
4. P. Szmaj (red.): „Inżynieria programowania. Metody i ćwiczenia laboratoryjne”, Wydawnictwo Politechniki Śląskiej, Gliwice, 2003.
5. A. Jaskiewicz: „Inżynieria oprogramowania”, Helion, 1997.

22. Literatura uzupełniająca:

1. W. Dąbrowski, A. Stasiak, M. Wolski: „Modelowanie systemów informatycznych w języku UML 2.1”, PWN, 2009.
2. S. H. Kan: „Metryki i modele w inżynierii jakości oprogramowania”, PWN, 2006.
3. P. Wróblewski: „Zarządzanie projektami z wykorzystaniem darmowego oprogramowania”, Helion 2009.
4. I. Sommerville: „Inżynieria Oprogramowania”, WNT, 2003.
5. B. Wiszniewski, B. Bereza-Jarociński: „Teoria i praktyka testowania programów”, PWN, 2006.
6. J. Spolsky: „Projektowanie interfejsu użytkownika. Poradnik dla programistów”, Mikom 2001.
7. S. Krug: „Nie każ mi myśleć. O życiowym podejściu do funkcjonalności stron internetowych”, Helion, 2006.

23. Nakład pracy studenta potrzebny do osiągnięcia efektów kształcenia

Lp.	Forma zajęć	Liczba godzin kontaktowych / pracy studenta
1	Wykład	30/10
2	Ćwiczenia	/
3	Laboratorium	/
4	Projekt	75/155
5	Seminarium	/
6	Inne	/
	Suma godzin	105/165

24.

Suma wszystkich godzin	270
-------------------------------	------------

25.

Liczba punktów ECTS	9
----------------------------	----------

26.

Liczba punktów ECTS uzyskanych na zajęciach z bezpośrednim udziałem nauczyciela akademickiego	9
--	----------

27.

Liczba punktów ECTS uzyskanych na zajęciach o charakterze praktycznym (laboratoria, projekty)	8
--	----------

28. Uwagi:

Aktywność na wykładzie 0-10 pkt. Projekt 0-90 pkt. W ramach projektu obowiązkowa praca w zespole lub międzyzespołowa, z wykorzystaniem narzędzi pracy grupowej.
 Student powinien uzyskać powyżej 40% punktów z każdego efektu kształcenia. Końcowa ocena wg skali punktowej: 41-55: dost, 56-70: dost plus, 71-80: dobry, 81-90: dobry plus, 91-100: bdb.

Zatwierdzono:

.....
 (data i podpis prowadzącego)

.....
 (data i podpis dyrektora instytutu/kierownika katedry/
 Dyrektora Kolegium Języków Obcych/kierownika lub
 dyrektora jednostki międzywydziałowej)