

KARTA PRZEDMIOTU DLA NABORU 2021/2022						
FORMA: STUDIA STACJONARNE						
INFORMACJE OGÓLNE						
1. Nazwa przedmiotu Testowanie oprogramowania						
2. Nazwa kierunku Informatyka						
3. Poziom studiów studia pierwszego stopnia						
4. Liczba punktów ECTS 2						
5. Liczba godzin w semestrze						
semestr	w	ćw	lab/lek	prj/zp	pws	prk
VII	15		15			
6. Język wykładowy polski						
7. Wykładowca dr inż. Marcin Klimek						
INFORMACJE SZCZEGÓŁOWE						
8. Wymagania wstępne						
1. Posiadanie podstawowej wiedzy na temat metod i narzędzi inżynierii oprogramowania.						
2. Posiadanie podstawowej wiedzy programistycznej						
3. Wiedza z zakresu baz danych, logiki						
9. Cele przedmiotu						
C1 nabycie umiejętności tworzenia i zarządzania testami.						
C2 zapoznanie studentów z podstawami testowania oprogramowania, a także z metodami, technikami i narzędziami inżynierii jakości						
10. Efekty uczenia się w zakresie wiedzy, umiejętności i kompetencji społecznych						
Student, który zaliczył przedmiot:					odniesienie do kierunkowych efektów uczenia się	
WIEDZA						
EU01	zna standardy i dobre praktyki wykonywania testów				K_W13 K_W14	
EU02	zna zagadnienia związane z technikami testowania oprogramowania, podziałem testów i dostępnymi standardami testowania oprogramowania.					
EU03	zna zasady zarządzania rozległymi projektami informatycznymi.					
UMIEJĘTNOŚCI						
EU04	umie testować oprogramowanie oraz stosuje właściwe testy na odpowiednim etapie procesu wytwórczego oprogramowania				K_U03 K_U09 K_U14 K_U18 K_U20	
EU05	umie sporządzać dokumentację specyfikującą testy i raporty o ich przebiegu.					
EU06	posiada umiejętność weryfikacji projektów informatycznych pod kątem jakościowym					
EU07	potrafi stosować narzędzia testujące w procesach wytwarzania oprogramowania.					
KOMPETENCJE SPOŁECZNE						

EU08	potrafi zespołowo przeprowadzić testy	K_K03 K_K04
EU09	zna kodeks etyczny i zagadnienia etyczne związane z pracą testera	
11. Treści programowe		
Forma zajęć – wykłady/laboratoria		
Wykłady: <ul style="list-style-type: none">1. Wprowadzenie do testowania. Testowanie w cyklu życia oprogramowania.2. Charakterystyka i podział testów oprogramowania: Błędy i awarie. Testy funkcjonalne i testy właściwości. Testy białoskrzynkowe (strukturalne) i czarnoskrzynkowe (funkcjonalne), testy poprawek błędów, testy regresywne. Testowanie statyczne i dynamiczne.3. Zastosowania testów w procesie rozwoju oprogramowania: Poziomy testowania oprogramowania. Testy jednostkowe, testy integracyjne wewnętrzne, testy systemowe, testy integracyjne zewnętrzne, testy akceptacyjne.4. Standardy w testowaniu oprogramowania.5. Testowanie w fazie utrzymania: Testowanie po rozbudowie i udoskonalaniu systemu. Modyfikacje naprawcze i awaryjne.6. Systemy raportowania i śledzenia błędów. Proces usuwania błędów7. Techniki scalania rozwiązań informatycznych: Integracja aplikacji, integracja danych, integracja sieciowa, integracja systemowa.8. Testowanie нефункционалне.		
Laboratorium: <ul style="list-style-type: none">1. Statyczna inspekcja i przeglądy kodu: Sprawdzanie kodu zgodnie z listą potencjalnych błędów.2. Projektowanie przypadków testowych w testach czarno skrzynkowych.3. Projektowanie przypadków testowych w testach biało skrzynkowych.4. Testy jednostkowe.5. Przygotowanie planu testów i raportu testów.6. Wersje oprogramowania i testy akceptacyjne.7. Testowanie aplikacji internetowych i systemów rozproszonych.8. Wykorzystanie narzędzi do automatyzacji: xUnit, Selenium, JIRA (zarządzanie incydentami) itp.9. Testowanie eksploracyjne10. Analiza statyczna i dynamiczna11. Testowanie нефункционалне (testy użyteczności, testy wydajności)		
12. Narzędzia/metody dydaktyczne		
<ul style="list-style-type: none">1. Podczas laboratoriów praca indywidualna studentów: pisanie testów np. jednostkowych. Studenci samodzielnie rozwiązują zadany problem praktyczny, dobierając odpowiednie narzędzia.<ul style="list-style-type: none">a. dyskusje wokół tematyki poruszanej na wykładzie,b. praktyczne projektowanie przypadków testowych,c. manualne wykonywanie testów,d. automatyzacja wykonania testów.2. Laboratoria w części odbywają się przy tablicy: studenci np. analizują różne strategie dotyczące testowania oprogramowanie3. Wykład prowadzony z wykorzystaniem projektora multimedialnego. Wykłady przygotowane w formie prezentacji PowerPoint.4. Treści prezentowane na wykładzie są wzbogacone o pokazy praktycznych rozwiązań odnoszących się do prezentowanych zagadnień.		
13. Sposoby oceny (częstkowe, końcowe)		
<ul style="list-style-type: none">1. Kolokwium przy komputerze: implementacja testów jednostkowych2. Kartkówki na laboratoriach		

3. Ocena realizacji projektu zaliczeniowego.	
4. Ocena ciągła pracy indywidualnej podczas laboratoriów.	
5. Egzamin w formie pisemnej sprawdzający wiedzę z wykładów i laboratoriów	
14. Obciążenie pracą studenta	
Forma aktywności	liczba godzin
1. Zajęcia z bezpośrednim udziałem nauczyciela oraz konsultacje	35
2. Nakład pracy studenta	15
suma	50
liczba punktów ECTS	2
15. Literatura	
Literatura podstawowa:	
1. Janusz Górski, Cezary Orłowski, Inżynieria Oprogramowania w procesach integracji systemów informatycznych , PWNT, 2010.	
2. Vladimir Khorikov, Testy jednostkowe : zasady, praktyki i wzorce, Gliwice : Wydawnictwo Helion, 2020.	
3. Praca zbiorowa pod redakcją Jana Werewki, Zarządzanie projektami w przedsiębiorstwie informatycznym, AGH UWND, 2012.	
4. Adam Roman, Testowanie i jakość oprogramowania. Modele, techniki, narzędzia, PWN 2015.	
5. Stephen H. Kan, Metryki i modele w inżynierii jakości oprogramowania, PWN 2006	
Literatura uzupełniająca:	
1. Roy Osheroove, Testy jednostkowe - świat niezawodnych aplikacji , Helion 2014	
2. Glenford Myers i in., Sztuka testowania oprogramowania, 2005	
3. Black Mitchell, Advanced Software Testing, Helion 2010	
16. Formy oceny – szczegóły	
Warunki uzyskania zaliczenia przedmiotu: zajęcia kończą się zaliczeniem z oceną. Sposób weryfikacji efektów uczenia się: Ocena stopnia osiągniętych przez studenta efektów uczenia się następuje wg poniższych kryteriów: 5.0 – zakładany efekt uczenia się został osiągnięty bez zastrzeżeń 4.5 – zakładany efekt uczenia się został osiągnięty z pojedynczymi brakami/błędami 4.0 – zakładany efekt uczenia się został osiągnięty z nielicznymi brakami/błędami 3.5 – zakładany efekt uczenia się został osiągnięty z wieloma brakami/błędami 3.0 – zakładany efekt kształcenia został osiągnięty z licznymi i istotnymi brakami/błędami (minimalnie wymagany poziom osiągnięcia efektu) 2.0 – zakładany efekt uczenia się nie został osiągnięty	
17. Inne przydatne informacje o przedmiocie	
1. Bezpośrednich informacji o problematyce zajęć i treściach programowych udziela prowadzący w trakcie zajęć i podczas konsultacji	
2. Zajęcia odbywać się będą w Akademii Białskiej im. Jana Pawła II lub na platformie Teams	
3. Zajęcia odbywać się będą zgodnie z aktualnym planem zajęć	
4. Konsultacje odbywać się będą zgodnie z obowiązującym terminarzem	