



Akademia Nauk Stosowanych
im. Hipolita Cegielskiego w Gnieźnie Uczelnia Państwowa

SYLABUS

Pozycja przedmiotu w planie:		R.III / S.V
1. OGÓLNY OPIS PRZEDMIOTU		
1	Nazwa modułu	Moduł zajęć specjalnościowych
2	Nazwa przedmiotu	Podstawy manualnego testowania oprogramowania
3	Kierunek studiów	Bezpieczeństwo wewnętrzne
4	Poziom studiów	Licencjackie
5	Forma studiów	Stacjonarne
6	Profil studiów	Praktyczny
7	Rok studiów	Trzeci
8	Semestr przedmiotu	Piąty
9	Jednostka prowadząca kierunek studiów	Instytut Nauk o Bezpieczeństwie
10	Liczba punktów ECTS	2
11	Sposób zaliczenia:	Zaliczenie z oceną (wykład i laboratoria).
12	Imię i nazwisko nauczyciela (li) akademickiego (ich), stopień lub tytuł naukowy, adres e-mail	mgr Karol GEPPERT, k.geppert@ans-gniezno.edu.pl
13	Imię i nazwisko koordynatora(ów) przedmiotu, stopień lub tytuł naukowy, adres e-mail	mgr Karol GEPPERT, k.geppert@ans-gniezno.edu.pl
14	Język wykładowy	Polski
15	Tryb prowadzenia zajęć	Mieszany
16	Sposób prowadzenia zajęć	Synchroniczny
17	Narzędzia informatyczne wykorzystywane do prowadzenia zajęć, udostępniania materiałów i komunikacji ze studentami	Platforma MS Teams / poczta elektroniczna
18	Przedmiot wprowadzający	Podstawy modelowania systemów w języku UML, Podstawy cybernetyki społecznej, Techniki zabezpieczania (OPSEC) i pozyskiwania (OSINT) informacji.
19	Wymagania wstępne	Posługuje się komputerem, przeglądarką internetową i podstawowymi narzędziami biurowymi oraz sprawnie zarządza plikami. Rozumie podstawowe pojęcia systemu informatycznego, procesu, wymagania i modelu oraz potrafi odróżnić obserwację od interpretacji. Potrafi pracować w zespole, dokumentować wykonane czynności i komunikować problem w sposób zwięzły i rzeczowy.
20	Cele przedmiotu:	
C1	Cel ogólny i wiedza: przygotowanie studenta do rozumienia roli testowania manualnego w zapewnianiu jakości systemów informatycznych. Student poznaje terminologię, zasady, proces testowy, poziomy i typy testów, techniki projektowania oraz współczesne standardy jakości i dokumentowania wyników.	

C2	Umiejętności: rozwinięcie zdolności analizowania wymagań i ryzyka, projektowania przypadków testowych, wykonywania testów aplikacji webowych i API, rejestrowania dowodów, raportowania defektów oraz przedstawiania wyników z wykorzystaniem narzędzi stosowanych w pracy zespołów QA.		
C3	Kompetencje społeczne: kształtowanie odpowiedzialności za rzetelność i powtarzalność testów, ochronę danych i poufność, konstruktywną komunikację z zespołem wytwórczym, krytyczną ocenę ograniczeń pokrycia oraz stałe aktualizowanie wiedzy i narzędzi.		
21	Forma zajęć, liczba godzin wymagająca bezpośredniego udziału nauczyciela akademickiego, liczba godzin nakładu pracy studenta:		
Forma zajęć		Liczba godzin	
Wykład		15	
Laboratoria		30	
Suma godzin		45	
lp.	Całkowity nakład pracy studenta		
1	Nakład pracy związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela akademickiego wynosi:		Godzinowe obciążenie studenta
	Wykład		
	Laboratoria		
	Nakład pracy związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela akademickiego wynosi 45 godzin, co odpowiada 1,5 punktom ECTS.		
2	Bilans samodzielnej pracy studenta (część teoretyczna): 1. Lektura wskazanych fragmentów sylabusu ISTQB, norm i materiałów dydaktycznych - 4 godziny. 2. Uporządkowanie notatek oraz analiza przykładowych wymagań, technik i raportów defektów - 3 godziny. 3. Przygotowanie do testu wiedzy i krótkich zadań problemowych - 3 godziny. Łącznie praca własna: 10 godzin.		10 godzin
3	Łączny nakład pracy studenta		55 godzin
4	Punkty ECTS za przedmiot		2 ECTS
5	Liczba punktów ECTS osiąganych w ramach zajęć kształtujących umiejętności praktyczne, w tym laboratoriów, analiz przypadków i projektu		0,5 ECTS

2. EFEKTY UCZENIA SIĘ DLA PRZEDMIOTU			
Kod	Przedmiotowy efekt uczenia się	Kierunkowe efekty uczenia się	Metody weryfikacji
Wiedza - Student zna i rozumie:			
EU_W01	Wyjaśnia cele, zasady i terminologię testowania oprogramowania oraz rozróżnia błąd, defekt i awarię, testowanie i debugowanie, zapewnianie jakości i kontrolę jakości, a także poziomy i typy testów.	BW1_W02; BW1_W03	Test wiedzy i zadania sytuacyjne - minimum 51%.

EU_W02	Charakteryzuje proces testowy, miejsce testowania w cyklu życia oprogramowania, podejście oparte na ryzyku oraz techniki czarnoskrzynkowe, oparte na doświadczeniu i współpracy, wskazując ich zastosowania i ograniczenia.	BW1_W03	Test wiedzy; mapa procesu i dobór techniki - minimum 51%.
EU_W03	Objaśnia charakterystyki jakości produktu, zasady identyfikowalności i dokumentowania, a także prawne i etyczne uwarunkowania ochrony danych, dostępności, bezpieczeństwa, własności intelektualnej i wspomagania testów przez AI.	BW1_W02; BW1_W17	Audyt jakości, danych i użycia narzędzi - minimum 51%.
Umiejętności - Student potrafi:			
EU_U01	Analizuje wymagania, historyjki użytkownika i kryteria akceptacji, identyfikuje niejednoznaczności, ryzyka produktowe, warunki testowe i priorytety oraz tworzy macierz identyfikowalności.	BW1_U05; BW1_U07	Analiza podstawy testów i macierz ryzyka - poprawność >= 51%.
EU_U02	Projektuje przypadki i zestawy testowe z wykorzystaniem klas równoważności, wartości brzegowych, tablic decyzyjnych, przejść stanów, scenariuszy i technik kombinatorycznych oraz ocenia uzyskane pokrycie.	BW1_U05; BW1_U07	Indywidualny pakiet projektowania testów - minimum 51%.
EU_U03	Przygotowuje środowisko i dane, wykonuje manualne testy aplikacji webowej i API, rejestruje wyniki i dowody, odtwarza defekty oraz przeprowadza testy potwierdzające i regresję z użyciem właściwych narzędzi.	BW1_U05; BW1_U13	Dziennik wykonania, dowody i raporty defektów - minimum 51%.
EU_U04	Współpracuje w zespole QA, jasno raportuje defekty i status jakości, uzasadnia dotkliwość i priorytet, uczestniczy w triage oraz aktualizuje plan testów na podstawie wyników i informacji zwrotnej.	BW1_U04; BW1_U06	Projekt zespołowy, triage i prezentacja raportu.
Kompetencje społeczne - Student jest gotów do:			
EU_K01	Odpowiedzialnego, bezstronnego i etycznego wykonywania testów, ochrony danych i informacji poufnych, zachowania integralności dowodów oraz zgłaszania ryzyka bez przypisywania winy osobom.	BW1_K02; BW1_K06	Audyt etyczny, praca z danymi i jakość dokumentacji.
EU_K02	Krytycznej oceny kompletności i ograniczeń własnych testów, przyjmowania informacji zwrotnej, doskonalenia warsztatu oraz weryfikowania wyników narzędzi i treści generowanych przez AI.	BW1_K05; BW1_K06	Retrospektywa, ocena koleżeńska i dziennik uczenia się.

3. TREŚCI PROGRAMOWE ODNIESIONE DO EFEKTÓW UCZENIA SIĘ	
Wykłady	
Treści programowe	Efekty uczenia
BLOK I: Fundamenty jakości i testowania - 2 h - Terminologia: jakość, testowanie, zapewnianie jakości, kontrola jakości, błąd, defekt, awaria, przyczyna podstawowa i debugowanie. - Cele testowania i siedem zasad ISTQB; niezależność testowania, identyfikowalność oraz podejście całego zespołu. - Rola testera manualnego w ograniczaniu ryzyka i dostarczaniu informacji o jakości, a nie w dowodzeniu braku defektów.	[EU_W01; EU_K02]

BLOK II: Testowanie w cyklu życia oprogramowania - 2 h - Modele sekwencyjne, iteracyjne i zwinne; DevOps, ciągłe dostarczanie, shift-left i informacja zwrotna. - Poziomy i typy testów; testy statyczne i dynamiczne, funkcjonalne i нефункционалне, potwierdzające, regresyjne i utrzymaniowe. - Proces testowy według ISTQB i ISO/IEC/IEEE 29119: analiza, projektowanie, implementacja, wykonanie, monitorowanie, raportowanie i zamknięcie.	[EU_W01; EU_W02]
BLOK III: Analiza i projektowanie testów - 3 h - Podstawa testów, warunki testowe, przypadek testowy, procedura, dane, oczekiwany wynik, pokrycie i identyfikowalność. - Klasy równoważności i analiza wartości brzegowych; testy pozytywne, negatywne oraz dobór reprezentantów danych. - Tablice decyzyjne, przejścia stanów, scenariusze przypadków użycia i testowanie kombinatoryczne pairwise. - Techniki oparte na doświadczeniu: zgadywanie błędów, listy kontrolne i testowanie eksploracyjne.	[EU_W02; EU_U01; EU_U02]
BLOK IV: Jakość aplikacji webowych, mobilnych i API - 3 h - Model klient-serwer, podstawy HTTP, kontekst przeglądarki i urządzenia mobilnego oraz źródła danych obserwowalne przez testera. - Model jakości ISO/IEC 25010:2023: dziewięć charakterystyk produktu i ich przełożenie na cele, ryzyka i warunki testowe. - Manualne testowanie API: metody, parametry, nagłówki, autoryzacja, kody odpowiedzi, treść odpowiedzi i walidacja kontraktu. - Podstawy dostępności WCAG 2.2 i bezpiecznego testowania według OWASP; granice testów manualnych i konieczność pracy w kontrolowanym środowisku.	[EU_W03; EU_U03; EU_K01]
BLOK V: Zarządzanie testami, defektami i informacją o jakości - 2 h - Podejście oparte na ryzyku, priorytetyzacja, zakres, kryteria wejścia i wyjścia oraz dobór regresji. - Cykl życia defektu, dotykliwość i priorytet; cechy odtwarzalnego raportu: środowisko, kroki, dane, wynik oczekiwany i rzeczywisty, dowody. - Metryki pokrycia, wykonania i defektów, ich ograniczenia oraz zwięzłe raportowanie statusu interesariuszom.	[EU_W02; EU_U04; EU_K02]
BLOK VI: Testowanie eksploracyjne, narzędzia, AI i etyka - 3 h - Sesyjne testowanie eksploracyjne: misja, heurystyki, notatki, mapa pokrycia i debriefing. - Wsparcie generatywnej AI w analizie wymagań, tworzeniu pomysłów testowych i redakcji raportów; halucynacje, stronniczość i obowiązek weryfikacji przez człowieka. - Korzyści i ryzyka narzędzi testowych oraz automatyzacji; kryteria wyboru narzędzia i granice testowania manualnego. - Ochrona danych, poufność, własność intelektualna, kontrola dostępu, integralność dowodów i odpowiedzialność zawodowa testera.	[EU_W03; EU_K01; EU_K02]
Laboratoria	
Treści programowe	Efekty uczenia
BLOK I: Środowisko testowe i obserwacja jakości - 3 h - Rozpoznanie funkcji demonstracyjnej aplikacji, konfiguracji, wersji, przeglądarki i ograniczeń środowiska. - Zadanie: inwentaryzacja obszarów funkcjonalnych, pierwsza lista ryzyk i karta krótkiej sesji testowej. - Metody/narzędzia: aplikacja demonstracyjna, przeglądarka, DevTools, arkusz kontroli konfiguracji.	[EU_U01; EU_K01]
BLOK II: Analiza wymagań i planowanie oparte na ryzyku - 3 h - Przegląd wymagań, historyjek użytkownika i kryteriów akceptacji pod kątem kompletności, jednoznaczności i testowalności. - Zadanie: pytania do analityka, warunki testowe, macierz ryzyka i identyfikowalność wymagania-test. - Metody/narzędzia: case study, checklist review, Jira/YouTrack lub narzędzie równoważne, arkusz RTM.	[EU_U01; EU_U04; EU_K02]

BLOK III: Klasy równoważności i wartości brzegowe - 4 h - Projektowanie danych i przypadków dla formularzy, walidacji, zakresów liczbowych, dat i pól tekstowych. - Zadanie: testy pozytywne i negatywne z oczekiwanymi wynikami oraz obliczenie pokrycia klas i granic. - Metody/narzędzia: TestRail/Xray/Zephyr lub arkusz kalkulacyjny; wzajemny przegląd przypadków.	[EU_U02; EU_K02]
BLOK IV: Tablice decyzyjne, stany i kombinacje - 4 h - Modelowanie reguł biznesowych, uprawnień, rabatów, logowania i cyklu życia obiektu. - Zadanie: tablica decyzyjna, diagram przejść stanów i zestaw pairwise z uzasadnieniem ograniczenia liczby testów. - Metody/narzędzia: arkusz kalkulacyjny, PICT/AllPairs lub narzędzie równoważne, peer review.	[EU_U02; EU_U04]
BLOK V: Manualne testy aplikacji webowej i dostępności - 4 h - Wykonanie scenariuszy funkcjonalnych, testów responsywności i zgodności między przeglądarkami oraz obserwacja konsoli, sieci i pamięci. - Zadanie: dokumentacja wyników, zrzuty i logi, test klawiatury, fokusu, komunikatów błędów i podstawowego kontrastu. - Metody/narzędzia: Chrome/Firefox DevTools, Lighthouse/WAVE/axe DevTools lub narzędzia równoważne.	[EU_U03; EU_K01]
BLOK VI: Manualne testy API i danych - 4 h - Budowanie żądań GET, POST, PUT/PATCH i DELETE; parametry, nagłówki, uwierzytelnienie i kontrola kodów odpowiedzi. - Zadanie: przypadki pozytywne i negatywne, walidacja treści odpowiedzi, spójności danych i obsługi błędów. - Metody/narzędzia: Postman/Bruno/Insomnia lub narzędzie równoważne; wyłącznie dane syntetyczne i środowisko demonstracyjne.	[EU_U03; EU_K01]
BLOK VII: Raportowanie defektów, triage i regresja - 4 h - Redagowanie jednoznacznego tytułu, warunków wstępnych, kroków, wyników, dowodów, dotkliwości i priorytetu. - Zadanie: reprodukcja defektu innego zespołu, symulacja triage oraz dobór testów potwierdzających i regresyjnych. - Metody/narzędzia: Jira/YouTrack lub narzędzie równoważne, nagranie ekranu, logi i lista regresji.	[EU_U03; EU_U04; EU_K02]
BLOK VIII: Projekt integracyjny - test produktu i raport jakości - 4 h - Zespół planuje, projektuje i wykonuje testy wybranego zakresu aplikacji demonstracyjnej zgodnie z priorytetami ryzyka. - Zadanie: pakiet testowy, dowody wykonania, raporty defektów, podsumowanie pokrycia, ryzyka rezydualne i rekomendacja wydania. - Metody/narzędzia: repozytorium testów, system zgłoszeń, briefing, ocena koleżeńska, retrospektywa i audyt użycia AI.	[EU_U01; EU_U02; EU_U03; EU_U04; EU_K01; EU_K02]

4. LITERATURA

Materiały laboratoryjne, wymagania, aplikacje demonstracyjne i dane syntetyczne są udostępniane studentom przez Microsoft Teams. Literatura i dokumentacja narzędzi są okresowo aktualizowane.

Literatura podstawowa	1. ISTQB, Certified Tester Foundation Level Syllabus v4.0.1, International Software Testing Qualifications Board, 15.09.2024. 2. Roman A., Testowanie i jakość oprogramowania. Modele, techniki, narzędzia, wyd. 2, Wydawnictwo Naukowe PWN, Warszawa 2022. 3. Roman A., Zmitrowicz K. (red.), Testowanie oprogramowania w praktyce, Wydawnictwo Naukowe PWN, Warszawa 2017. 4. Gregory J., Crispin L., Agile Testing Condensed: A Brief Introduction, Leanpub, 2019. 5. Hendrickson E., Explore It!: Reduce Risk and Increase Confidence with Exploratory Testing, Pragmatic Bookshelf, 2013.
Literatura uzupełniająca	1. ISO/IEC/IEEE 29119-1:2022, Software and systems engineering - Software testing - Part 1: General concepts. 2. ISO/IEC/IEEE 29119-2:2021, Software and systems engineering - Software testing - Part 2: Test processes. 3. ISO/IEC 25010:2023, Systems and software engineering - SQuaRE - Product quality model. 4. W3C, Web Content Accessibility Guidelines (WCAG) 2.2, Recommendation, 2024. 5. Capgemini, Sogeti, OpenText, World Quality Report 2025-26, 2025.
Źródła internetowe	1. ISTQB, CTFL v4.0 oraz słownik terminów: https://www.istqb.org/certifications/certified-tester-foundation-level 2. OWASP, Web Security Testing Guide v4.2: https://owasp.org/www-project-web-security-testing-guide/v42/ 3. W3C, WCAG 2.2: https://www.w3.org/TR/WCAG22/ 4. Dokumentacja Chrome DevTools i Postman: https://developer.chrome.com/docs/devtools/ oraz https://learning.postman.com/ 5. Dokumentacja narzędzi Jira/YouTrack, TestRail/Xray/Zephyr, Bruno/Insomnia i axe/WAVE - zależnie od dostępności licencji; dopuszcza się rozwiązania równoważne.

5. METODY DYDAKTYCZNE

Forma	Metody dydaktyczne
Wykład	• Wykład problemowy i konwersatoryjny z krótkimi zadaniami diagnostycznymi oraz przykładami z procesu wytwarzania oprogramowania. • Analiza wymagań, raportów defektów, metryk i ryzyk; porównywanie technik projektowania testów. • Demonstracje pracy z aplikacją, narzędziami zarządzania testami, DevTools i klientem API. • Dyskusja nad jakością, dostępnością, bezpieczeństwem, ochroną danych, AI i odpowiedzialnością zawodową testera.
Laboratoria	• Problem Based Learning i praca na progresywnie rozwijanej aplikacji demonstracyjnej oraz zestawie wymagań. • Warsztaty projektowania testów, wykonania manualnego, dokumentowania dowodów i raportowania defektów. • Testowanie aplikacji webowej, podstaw dostępności i API z wykorzystaniem narzędzi branżowych lub równoważnych. • Praca w parach i zespołach, peer review, symulacja triage, briefing jakości i retrospektywa. • Testowanie eksploracyjne oparte na misjach, heurystykach, notatkach sesyjnych i debriefingu. • Retrieval practice i feedforward: krótkie quizy, rubryka kryterialna oraz poprawa przypadków i raportów po informacji zwrotnej.

6. METODY I KRYTERIA OCENIANIA

Forma zajęć: Wykład	Forma zaliczenia: Zaliczenie z oceną Test wiedzy i zadania problemowo-sytuacyjne
---------------------	---

Test wiedzy - 20% oceny końcowej Test obejmuje pytania problemowe, krótkie scenariusze i dobór technik weryfikujące EU_W01-EU_W03. Oceniane są: poprawność terminologiczna, rozumienie procesu testowego, dobór techniki, interpretacja ryzyka i zasady jakości, prawa oraz etyki. Skala ocen: 91-100% - 5,0; 85-90% - 4,5; 76-84% - 4,0; 66-75% - 3,5; 51-65% - 3,0; 0-50% - 2,0.			
Kryteria poziomu wykonania Niedostateczny (2,0): myli podstawowe pojęcia, nie wiąże testów z wymaganiami, tworzy nieodtwarzalne przypadki lub raporty i ignoruje ryzyka dla danych. Dostateczny (3,0): stosuje podstawowy proces i techniki, lecz zakres, oczekiwane wyniki, dowody lub uzasadnienie priorytetów są częściowe. Dobry (4,0): trafnie analizuje ryzyko, dobiera techniki, wykonuje testy powtarzalnie i komunikuje użyteczne wyniki; drobne braki nie zmieniają wniosku o jakości. Bardzo dobry (5,0): łączy wymagania, ryzyko, pokrycie i dowody, krytycznie ocenia ograniczenia, proponuje efektywną regresję oraz audytowalną rekomendację wydania.			
Warunek zaliczenia testu: co najmniej 51% punktów.			
Forma zajęć: Laboratoria		Forma zaliczenia: Zaliczenie z oceną Ocena ciągła i projekt integracyjny	
Składowe oceny końcowej (łącznie 100%): 1. Test wiedzy i zadania problemowo-sytuacyjne - 20%. 2. Indywidualny pakiet analizy i projektowania testów - 25%. 3. Zespołowy projekt testowy: wykonanie, dowody, defekty i raport jakości - 40%. 4. Aktywność laboratoryjna, peer review, triage i refleksja - 15%.			
Warunki konieczne: Uzyskanie co najmniej 51% z testu wiedzy i co najmniej 51% z projektu oraz wykazanie osiągnięcia wszystkich PEU. Dokumentacja projektu musi zapewniać identyfikowalność, odtwarzalność, dowody wykonania, ocenę pokrycia, ryzyka rezydualne i uzasadnioną rekomendację jakości. <i>Wykorzystanie generatywnej AI wymaga jawnego oznaczenia zakresu użycia i samodzielnej weryfikacji wyniku. Do publicznych narzędzi nie wolno wprowadzać danych osobowych, tajemnic organizacji, kluczy dostępowych ani materiałów ze środowiska produkcyjnego.</i>			
Arkusz szacowania oceny			
Kryterium oceny	Niedostateczny (2.0)	Dostateczny (3.0)	Dobry (4.0)/ Bardzo dobry (5.0)
Analiza i projektowanie testów	Nie identyfikuje podstawy i ryzyka; przypadki są przypadkowe, niekompletne lub bez oczekiwanych wyników.	Rozpoznaje podstawowe wymagania i stosuje prostą technikę, lecz pokrycie lub priorytety wymagają poprawy.	Zapewnia identyfikowalność, uzasadniony dobór technik, dane pozytywne i negatywne oraz przejrzystą ocenę pokrycia.
Wykonanie i dowody	Nie kontroluje środowiska i danych; wynik jest niepowtarzalny lub nie ma wiarygodnych dowodów.	Wykonuje główne kroki i zapisuje wynik, lecz opis konfiguracji, dane albo dowody są częściowe.	Prowadzi powtarzalne wykonanie, kontroluje wersję i dane, gromadzi adekwatne dowody oraz poprawnie wykonuje retest i regresję.
Defekty i komunikacja	Raport nie pozwala odtworzyć problemu; dotkliwość i priorytet są arbitralne, a komunikacja konfliktowa.	Raport jest zrozumiały, lecz wymaga doprecyzowania kroków, oczekiwań, dowodów lub wpływu biznesowego.	Raport jest zwięzły i odtwarzalny; triage opiera się na wpływie i ryzyku, a zespół przekazuje jasny status jakości.
Etyka, dane i krytycyzm	Używa nieuprawnionych danych lub bezkrytycznie przyjmuje wynik narzędzia	Rozpoznaje główne ryzyka i ograniczenia, lecz środki ochronne lub autorefleksja są ogólne.	Minimalizuje dane, chroni dostęp i dowody, weryfikuje wyniki narzędzi i AI oraz

Kryterium oceny	Niedostateczny (2.0)	Dostateczny (3.0)	Dobry (4.0)/ Bardzo dobry (5.0)
	albo AI; nie dostrzega ograniczeń.		jawnie komunikuje luki pokrycia i niepewność.

	Zatwierdzenie karty opisu zajęć	
	Stanowisko Tytuł/stopień naukowy, imię nazwisko	Podpis
Opracował	mgr Karol Geppert	
Zatwierdził	p.o. Dyrektora Instytutu mgr Karol Geppert	