

AKADEMIA TECHNICZNO-INFORMATYCZNA W NAUKACH STOSOWANYCH

KARTA OPISU PRZEDMIOTU

Wydział		Informatyki	
Kierunek		Informatyka	
Specjalność			
Semestr	V,	Program studiów, dla którego obowiązuje sylabus	2025/2026
Stopień studiów	I		

Nazwa przedmiotu	Automatyzacja testów I			
Kod przedmiotu	ATI			
Łączna liczba godzin	30	Tryb	stacjonarny	niestacjonarny
Profil kształcenia	Ogólnoakademicki (A)		Praktyczny (P)	
Forma zajęć	laboratorium			
Język przedmiotu	polski			
Liczba punktów ECTS	2			

Prowadzący zajęcia	
Forma prowadzonych zajęć	Laboratorium
Wymiar zajęć	30 h
Stopień (tytuł) naukowy	
Imię	
Nazwisko	

Wymagania wstępne	Podstawowa znajomość programowania w co najmniej jednym języku, wiedza z zakresu podstaw inżynierii oprogramowania, znajomość podstawowych struktur danych i algorytmów.
Założenia i cele przedmiotu	Celem przedmiotu jest wprowadzenie studentów w zagadnienia automatyzacji testów oprogramowania, w tym poznanie podstawowych narzędzi i frameworków do testów UI (np. Selenium WebDriver), zrozumienie korzyści i wyzwań związanych z automatyzacją oraz opanowanie podstaw programowania w kontekście tworzenia skryptów testowych. Studenci nauczą się także tworzyć proste projekty testów automatycznych i generować raporty z wyników testów.
Metody dydaktyczne	<ol style="list-style-type: none"> 1. Prezentacje multimedialne. 2. Pokazy przykładowych rozwiązań problemów. 3. Rozwiązywanie zadań praktycznych.

Efekty uczenia się (odniesienie do charakterystyk poziomów Polskiej Ramy Kwalifikacji)		Odniesienie do efektów dla kierunku	Odniesienie do efektów uczenia się wg Polskiej Ramy Kwalifikacji
WIEDZA – absolwent zna i rozumie:	W01. Podstawowe zasady programowania obiektowego i strukturalnego, niezbędne do tworzenia skryptów testowych.	K_W02 K_W06 K_W10	P6S_WG P6S_WG_INŻ

AKADEMIA TECHNICZNO-INFORMATYCZNA W NAUKACH STOSOWANYCH

	<p>W02. Koncepcje i etapy procesu automatyzacji testów w kontekście projektowania systemów informatycznych.</p> <p>W03. Specyfikę testowania interfejsów webowych, w tym znaczenie HTML, CSS, JavaScript dla stabilności i skuteczności testów UI.</p> <p>W04. Rolę wzorców projektowych i metody parametryzacji testów w zwiększaniu efektywności i łatwości utrzymania projektów testowych.</p> <p>W05. Znaczenie automatyzacji testów w cyklu życia projektu programistycznego oraz wpływ testów automatycznych na jakość i terminowość dostarczania oprogramowania.</p>	<p>K_W20 K_W21</p>	
<p>UMIEJĘTNOŚCI – absolwent potrafi:</p>	<p>U01. Pozyskiwać informacje o narzędziach i praktykach automatyzacji testów z dokumentacji i repozytoriów, a następnie krytycznie je ocenić i wdrożyć.</p> <p>U02. Pisać proste skrypty testowe w wybranym języku programowania z wykorzystaniem narzędzi automatyzujących testy (np. Selenium WebDriver).</p> <p>U03. Zaplanować proces automatyzacji testów prostego modułu aplikacji, wskazując testy najbardziej opłacalne do zautomatyzowania.</p> <p>U04. Zintegrować testy automatyczne z aplikacją webową, uwzględniając specyfikę front-endu i dynamikę elementów interfejsu.</p> <p>U05. Opracować dokumentację projektów testowych, w tym opisy środowiska testowego, strukturę testów automatycznych i instrukcje uruchamiania.</p>	<p>K_U01 K_U02 K_U03 K_U04 K_U09 K_U15 K_U23</p>	<p>P6S_UW P6S_UW_INŻ P6S_UO P6S_KK P6S_UK</p>
<p>KOMPETENCJE SPOŁECZNE – absolwent jest gotów do</p>	<p>K01. Pracy w zespole, przyjmując w nim różne role.</p> <p>K02. Krytycznej oceny dostępnego oprogramowania i rozwiązań programistycznych.</p> <p>K03. Propagowania kodeksu etyki zawodowej i dbania o jakość oprogramowania.</p>	<p>K_K04 K_K05 K_K06</p>	<p>P6S_UO P6S_KR P6S_KK</p>

Lp.	Tematyka zajęć	Liczba godzin
Forma zajęć – laboratorium		
1	Wprowadzenie do automatyzacji testów: cele i korzyści. Dlaczego automatyzować testy? Kiedy automatyzacja jest opłacalna? Wyzwania w automatyzacji testów.	3
2	Przegląd narzędzi do automatyzacji testów. Narzędzia open-source vs. komercyjne. Wprowadzenie do Selenium WebDriver. Inne popularne narzędzia: Appium, JUnit,	4

AKADEMIA TECHNICZNO-INFORMATYCZNA W NAUKACH STOSOWANYCH

	TestNG.	
3	Podstawy programowania w kontekście testów automatycznych. Wprowadzenie do języka programowania (np. Java, Python). Struktury danych i kontrola przepływu. Pisanie prostych skryptów testowych.	7
4	Automatyzacja testów interfejsu użytkownika. Elementy DOM i interakcja z nimi. Identyfikacja obiektów na stronie (lokatory). Synchronizacja i oczekiwania.	6
5	Tworzenie skryptów automatyzujących testy. Struktura projektu testowego. Pisanie i uruchamianie testów automatycznych. Generowanie raportów z testów. Zaliczenie.	10

Forma i warunki zaliczenia przedmiotu	Wykonanie projektów. Częstkowe prezentacje, zdawanie raportów, obrona projektów.	
Metody weryfikacji efektów uczenia się		Nr efektu uczenia się z sylabusu
	Ocena projektów i częstkowych prezentacji.	W01-W05, U01-U05, K01-K03

Literatura podstawowa	<ol style="list-style-type: none"> 1. A. Roman, <i>Testowanie i jakość oprogramowania. Modele, techniki, narzędzia</i>, Wydawnictwo Naukowe PWN, Warszawa 2015. 2. R. Smilgin, <i>Zawód tester. Od decyzji do zdobycia doświadczenia</i>, Wydawnictwo Naukowe PWN, Warszawa 2016.
Literatura uzupełniająca	<ol style="list-style-type: none"> 1. K. Zmitrowicz, <i>Automatyzacja testów aplikacji webowych z wykorzystaniem Selenium WebDriver</i>, Helion, Gliwice 2020. 2. J. Osowski, <i>Testowanie aplikacji mobilnych w praktyce</i>, Helion, Gliwice 2021.

Nakład pracy studenta	
	Liczba godzin
Zajęcia dydaktyczne	30
Przygotowanie się do zajęć	5
Studiowanie literatury	5
Udział w konsultacjach	2
Przygotowanie projektu / eseju / prezentacji itp.	18
Przygotowanie się do egzaminu / zaliczenia	-
Inne	-
ŁĄCZNY nakład pracy studenta w godz.	60
Liczba punktów ECTS	2