

WROCŁAWSKA WYŻSZA SZKOŁA INFORMATYKI STOSOWANEJ

KARTA OPISU PRZEDMIOTU

Wydział		Informatyki	
Kierunek		Informatyka	
Specjalność		Programowanie	
Semestr	IV	Program studiów, dla którego obowiązuje sylabus	2023/2024
Stopień studiów	I		

Nazwa przedmiotu	Technologie backendowe			
Kod przedmiotu	TB			
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 języków programowania, takich jak Java lub C++. Umiejętność obsługi komputera i środowisk programistycznych. Znajomość podstawowych algorytmów i struktur danych.
Założenia i cele przedmiotu	Celem przedmiotu jest wprowadzenie studentów w zagadnienia projektowania i implementacji logiki po stronie serwera. Studenci poznają podstawy programowania backendowego, tworzenia API RESTful z wykorzystaniem popularnych frameworków (np. Express.js, Spring), metody integracji z bazami danych SQL i NoSQL oraz mechanizmy autentykacji i autoryzacji. Szczególny nacisk położony jest na bezpieczeństwo aplikacji backendowych i ochronę przed najczęstszymi atakami.
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 backendowego oraz różnice między językami i platformami (np. Node.js, Java) w kontekście	K_W02 K_W06	P6S_WG P6S_WG_INŻ

WROCŁAWSKA WYŻSZA SZKOŁA INFORMATYKI STOSOWANEJ

	<p>tworzenia serwerów.</p> <p>W02. Architektūrę i strukturę API RESTful oraz sposoby projektowania interfejsów do komunikacji między klientem a serwerem.</p> <p>W03. Zaawansowane metody projektowania, optymalizacji i zarządzania bazami danych relacyjnymi (SQL) oraz nierelacyjnymi (NoSQL).</p> <p>W04. Mechanizmy autentykacji i autoryzacji (JWT, OAuth) oraz ich znaczenie w zabezpieczeniu dostępu do aplikacji backendowych.</p> <p>W05. Rodzaje zagrożeń bezpieczeństwa aplikacji backendowych oraz zasady ochrony przed typowymi atakami (SQL Injection, XSS, CSRF).</p>	<p>K_W14 K_W16 K_W18</p>	
<p>UMIEJĘTNOŚCI – absolwent potrafi:</p>	<p>U01. Wykorzystać języki programowania backendowego i wzorce projektowe do tworzenia stabilnego i skalowalnego kodu po stronie serwera.</p> <p>U02. Zaprojektować, wdrożyć i przetestować API RESTful, w tym zdefiniować punkty końcowe, format danych oraz obsługę błędów.</p> <p>U03. Integrować aplikacje backendowe z bazami danych, korzystając z mechanizmów ORM, transakcji oraz indeksów dla optymalizacji wydajności.</p> <p>U04. Implementować mechanizmy autentykacji i autoryzacji, w tym generować i weryfikować tokeny JWT oraz korzystać z protokołu OAuth.</p> <p>U05. Zastosować dobre praktyki bezpieczeństwa, takie jak walidacja danych wejściowych, szyfrowanie połączeń i zarządzanie sekretami, w celu ochrony aplikacji przed atakami.</p>	<p>K_U01 K_U02 K_U03 K_U04 K_U09 K_U15 K_U18 K_U24 K_U25</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 możliwości urządzeń oprogramowania i systemów dostępnych na rynku IT.</p> <p>K03. Ciągłego samokształcenia się w celu dostosowywania się do dynamicznie zmieniających się technologii.</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	Podstawy programowania backendowego. Serwery, bazy danych, języki backendowe	8

WROCŁAWSKA WYŻSZA SZKOŁA INFORMATYKI STOSOWANEJ

	(Node.js, Java)	
2	Frameworki backendowe. Tworzenie API RESTful.	6
3	Bazy danych. Projektowanie, zarządzanie i optymalizacja baz danych SQL i NoSQL.	6
4	Autentykacja i autoryzacja. Implementacja mechanizmów logowania, tokenów JWT, OAuth.	6
5	Bezpieczeństwo aplikacji backendowych. Ochrona przed atakami typu SQL Injection, XSS, CSRF. Zaliczenie.	4

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 częściowych prezentacji.	W01-W05, U01-U05, K01-K03

Literatura podstawowa	<ol style="list-style-type: none"> 1. R. C. Martin, <i>Czysty kod. Podręcznik dobrego programisty</i>, Helion, Gliwice 2010. 2. J. Roszkowski, <i>Analiza i projektowanie strukturalne</i>, Helion, Gliwice, 2004. 3. N. Wirth, <i>Algorytmy + struktury danych = programy</i>, WNT, Warszawa 2002
Literatura uzupełniająca	<ol style="list-style-type: none"> 1. S. Prata, <i>Język C++. Szkoła programowania</i>. Wydanie VI, Helion, Gliwice 2019. 2. B. Eckel, <i>Thinking in Java. Edycja polska</i>, Helion, Gliwice 2006.

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