

POLITECHNIKA KRAKOWSKA IM. TADEUSZA KOŚCIUSZKI

KARTA PRZEDMIOTU

obowiązuje studentów rozpoczynających studia w roku akademickim 2025/2026

Wydział Mechaniczny

Kierunek studiów: Informatyka Stosowana

Profil: Ogólnoakademicki

Forma studiów: stacjonarne

Kod kierunku: S

Stopień studiów: I

Specjalności: Bez specjalności- blok A

1 INFORMACJE O PRZEDMIOCIE

NAZWA PRZEDMIOTU	Programowanie BackEnd
NAZWA PRZEDMIOTU W JĘZYKU ANGIELSKIM	
KOD PRZEDMIOTU	WM INFST oIS CB4 25/26
KATEGORIA PRZEDMIOTU	Przedmioty kierunkowe-bloki wybieralne
LICZBA PUNKTÓW ECTS	3.00
SEMESTRY	7

2 RODZAJ ZAJĘĆ, LICZBA GODZIN W PLANIE STUDIÓW

SEMESTR	WYKŁAD	ĆWICZENIA	LABORATORIUM	LABORATORIUM KOMPUTERO- WE	PROJEKT	SEMINARIUM
7	30	0	0	30	0	0

3 CELE PRZEDMIOTU

Cel 1 Student zna i rozumie strukturę i powiązania kompletnego stosu technologicznego aplikacji internetowej.

Cel 2 Student potrafi opracować aplikację serwerową współpracującą z front-endem.

4 WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

- 1 Kurs w zakresie dokumentacja techniczna II
- 2 Kurs w zakresie podstawy aplikacji internetowych
- 3 Kurs w zakresie programowanie front-end

5 EFEKTY KSZTAŁCENIA

- EK1 Wiedza** Absolwent zna i rozumie technologie programistyczne, metody projektowania i języki programowania do tworzenia aplikacji internetowych działających po stronie serwera.
- EK2 Umiejętności** Absolwent potrafi zaprojektować zgodnie ze specyfikacją aplikację serwerową o niewielkim lub średnim stopniu złożoności, oraz zaprojektować dla niej odpowiedni interfejs programistycznych wykorzystując informacje pozyskiwane z literatury, dokumentacji oraz serwisów internetowych.
- EK3 Umiejętności** Absolwent potrafi wykonać aplikację internetową po stronie serwera, w tym dobrać odpowiednie narzędzia programistyczne oraz sposób przechowywania i obsługi danych.
- EK4 Kompetencje społeczne** Absolwent jest gotów do ciągłego dokształcania się podnoszenia swoich kompetencji zawodowych i społecznych w zakresie i poszukiwania aktualnych rozwiązań w zakresie programowania back-end w warunkach szybko postępującego rozwoju informatyki.

6 TREŚCI PROGRAMOWE

WYKŁAD		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
W1	Struktura aplikacji typu full-stack. Konstrukcja serwerów aplikacji. Warstwy usług serwera. Warianty serwerów: Node.js, Java, .NET, PHP. Protokoły dostępowe serwerów. Techniki tworzenia komponentów serwerowych. Języki i biblioteki tworzenia komponentów serwerowych. Metody wdrażania aplikacji serwerowych. Utworzenie i konfiguracja serwera. Żądania: przekierowanie, parsowanie, obsługiwane. Moduły wspomagające na przykładzie biblioteki w rodzaju Express.js: instalacja, utworzenie warstwy middleware serwera, użycie komponentu router. Dynamiczne generowanie treści przez serwer, rodzaje szablonów treści (templating engines). Budowa serwera zgodnie ze wzorcem MVC, definiowanie poszczególnych komponentów i interakcji między nimi. Współpraca z relacyjnymi bazami danych z wykorzystaniem narzędzi wspomagających typu sequelize. Współpraca z bazami nierelacyjnymi na przykładzie mongoDB lub równoważnej. Zarządzanie sesjami i cookies, uwierzytelnianie użytkowników serwera, walidacja danych. Podstawy budowy interfejsów programistycznych zgodnie z metodologiami typu REST API i GraphQL oraz praktyczne zastosowania. Przetwarzanie asynchroniczne, gniazda sieciowe, biblioteka Socket.io. Techniki testowania i wdrażania aplikacji serwerowej.	30

LABORATORIUM KOMPUTEROWE		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
K1	Tworzenie i uruchamianie podstawowej struktury serwera z obsługą żądań. Tworzenie aplikacji serwerowej z wykorzystaniem bibliotek wspomagających typu Express.js. Tworzenie aplikacji serwerowej zgodnie z wzorcem MVC oraz dostępem do baz danych. Tworzenie interfejsów programistycznych według metodologii Rest API i GraphQL. Testowanie i wdrażanie serwera.	30

7 NARZĘDZIA DYDAKTYCZNE

N1 Wykłady

N2 Ćwiczenia laboratoryjne

8 OBCIĄŻENIE PRACĄ STUDENTA

FORMA AKTYWNOŚCI	ŚREDNIA LICZBA GODZIN NA ZREALIZOWANIE AKTYWNOŚCI
Godziny kontaktowe z nauczycielem akademickim, w tym:	
Godziny wynikające z planu studiów	60
Konsultacje przedmiotowe	6
Egzaminy i zaliczenia w sesji	0
Godziny bez udziału nauczyciela akademickiego wynikające z nakładu pracy studenta, w tym:	
Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	16
Opracowanie wyników	0
Przygotowanie raportu, projektu, prezentacji, dyskusji	0
zapoznawanie się z instrukcjami do ćwiczeń laboratoryjnych	8
SUMARYCZNA LICZBA GODZIN DLA PRZEDMIOTU WYNIKAJĄCA Z CAŁEGO NAKŁADU PRACY STUDENTA	90
SUMARYCZNA LICZBA PUNKTÓW ECTS DLA PRZEDMIOTU	3.00

9 SPOSOBY OCENY

OCENA FORMUJĄCA

F1 Egzamin z wykładu

F2 Kody źródłowe z zajęć laboratoryjnych

F3 Testy na zajęciach laboratoryjnych

OCENA PODSUMOWUJĄCA

P1 Średnia ważona ocen formujących

WARUNKI ZALICZENIA PRZEDMIOTU

W1 Pozytywny wynik egzaminu (powyżej 50% punktów)

W2 Pozytywny wynik laboratoriów (powyżej 50% punktów za kody źródłowe i testy)

W3 Obecność na min. 75% zajęć laboratoryjnych, w tym max. 1 nieobecność nieusprawiedliwiona

W4 Wszelkie niedotrzymania terminów skutkują obniżeniem liczby punktów

OCENA AKTYWNOŚCI BEZ UDZIAŁU NAUCZYCIELA

B1 Testy na zajęciach laboratoryjnych.

KRYTERIA OCENY

EFEKT KSZTAŁCENIA 1	
NA OCENĘ 2.0	Student uzyskał do 50 % punktów z egzaminu z wykładów.
NA OCENĘ 3.0	Student uzyskał więcej niż 50 %, do 60 % włącznie punktów z egzaminu z wykładów
NA OCENĘ 3.5	Student uzyskał więcej niż 60 %, do 70 % włącznie punktów z egzaminu z wykładów
NA OCENĘ 4.0	Student uzyskał więcej niż 70 %, do 80 % włącznie punktów z egzaminu z wykładów.
NA OCENĘ 4.5	Student uzyskał więcej niż 80 %, do 90 % włącznie punktów z egzaminu z wykładów.
NA OCENĘ 5.0	Student uzyskał więcej niż 90 % punktów z egzaminu z wykładów.
EFEKT KSZTAŁCENIA 2	
NA OCENĘ 2.0	Student uzyskał do 50 % punktów z zaliczeń testów na ćwiczeniach laboratoryjnych.
NA OCENĘ 3.0	Student uzyskał więcej niż 50 %, do 60 % włącznie punktów z zaliczeń testów na ćwiczeniach laboratoryjnych.
NA OCENĘ 3.5	Student uzyskał więcej niż 60 %, do 70 % włącznie punktów z zaliczeń testów na ćwiczeniach laboratoryjnych.
NA OCENĘ 4.0	Student uzyskał więcej niż 70 %, do 80 % włącznie punktów z zaliczeń testów na ćwiczeniach laboratoryjnych.
NA OCENĘ 4.5	Student uzyskał więcej niż 80 %, do 90 % włącznie punktów z zaliczeń testów na ćwiczeniach laboratoryjnych.

NA OCENĘ 5.0	Student uzyskał powyżej 90 % punktów z zaliczeń testów na ćwiczeniach laboratoryjnych.
EFEKT KSZTAŁCENIA 3	
NA OCENĘ 2.0	Student spełnia mniej niż 4 z 8 wymagań na ocenę 5.0
NA OCENĘ 3.0	Student spełnia 4 z 8 wymagań na ocenę 5.0
NA OCENĘ 3.5	Student spełnia 5 z 8 wymagań na ocenę 5.0
NA OCENĘ 4.0	Student spełnia 6 z 8 wymagań na ocenę 5.0
NA OCENĘ 4.5	Student spełnia 7 z 8 wymagań na ocenę 5.0
NA OCENĘ 5.0	Student potrafi poprawnie: (1)stworzyć i uruchomić podstawowe struktury serwera z obsługą żądań, (2)stworzyć aplikację serwerową z wykorzystaniem bibliotek wspomagających typu Express.js, (3)stworzyć aplikację serwerową zgodną z wzorcem MVC, (4)stworzyć aplikację serwerową z dostępem do baz danych, (5)stworzyć interfejs programistycznych według metodologii Rest API, (6)stworzyć interfejs programistycznych według metodologii GraphQL, (7)przetestować manualnie i w sposób zautomatyzowany działanie serwera, (8)wdrożyć serwer.
EFEKT KSZTAŁCENIA 4	
NA OCENĘ 2.0	Student uzyskał do 50% punktów za kody źródłowe z ćwiczeń laboratoryjnych, umieszczane w zdalnym repozytorium.
NA OCENĘ 3.0	Student uzyskał więcej niż 50 %, do 60 % włącznie punktów za kody źródłowe z ćwiczeń laboratoryjnych, umieszczane w zdalnym repozytorium.
NA OCENĘ 3.5	Student uzyskał więcej niż 60 %, do 70 % włącznie punktów za kody źródłowe z ćwiczeń laboratoryjnych, umieszczane w zdalnym repozytorium.
NA OCENĘ 4.0	Student uzyskał więcej niż 70 %, do 80 % włącznie punktów za kody źródłowe z ćwiczeń laboratoryjnych, umieszczane w zdalnym repozytorium.
NA OCENĘ 4.5	Student uzyskał więcej niż 80 %, do 90 % włącznie punktów za kody źródłowe z ćwiczeń laboratoryjnych, umieszczane w zdalnym repozytorium.
NA OCENĘ 5.0	Student uzyskał powyżej 90% punktów za kody źródłowe z ćwiczeń laboratoryjnych, umieszczane w zdalnym repozytorium.

10 MACIERZ REALIZACJI PRZEDMIOTU

EFEKT KSZTAŁCENIA	ODNIESIENIE DANEGO EFEKTU DO SZCZEGÓŁOWYCH EFEKTÓW ZDEFINIOWANYCH DLA PROGRAMU	CELE PRZEDMIOTU	TREŚCI PROGRAMOWE	NARZĘDZIA DYDAKTYCZNE	SPOSOBY OCENY
EK1		Cel 1 Cel 2	W1	N1	F1 P1
EK2		Cel 1 Cel 2	K1	N2	F3 P1
EK3		Cel 1 Cel 2	K1	N2	F2 F3 P1
EK4		Cel 1 Cel 2	W1 K1	N1 N2	F2 P1

11 WYKAZ LITERATURY

LITERATURA PODSTAWOWA

- [1] | **David Herron** — *Node.js Web Development, 5th Edition*, , 2020, Packt
- [2] | **Mardan, A** — *Full Stack JavaScript*, , 2018, Apress
- [3] | **Ethan Brown** — *Tworzenie aplikacji internetowych z użyciem Node i Express. Korzystanie ze stosu JavaScript. Wydanie II*, , 2020, Helion

LITERATURA UZUPEŁNIAJĄCA

- [1] | **Northwood, C** — *The Full Stack Developer*, , 2018, Apress
- [2] | **Zametti, F** — *Modern Full-Stack Development*, , 2020, Apress

LITERATURA DODATKOWA

- [1] | **Lech Lachowski** — *Single Page Web Applications. Programowanie aplikacji internetowych z JavaScript*, , 2015, Helion
- [2] | **Mariusz Walczak** — *Tworzenie nowoczesnych systemów webowych*, , 2016, Helion
- [3] | **Autor** — *Tytuł*, Miejscowość, 2022, Wydawnictwo

12 INFORMACJE O NAUCZYCIELACH AKADEMICKICH

OSOBA ODPOWIEDZIALNA ZA KARTĘ

dr inż. Paweł Lempa (kontakt: plempa@pk.edu.pl)

OSOBY PROWADZĄCE PRZEDMIOT

- 1 dr inż. Paweł Lempa (kontakt: pawel.lempe@pk.edu.pl)
- 2 dr hab. inż. Grzegorz Filo (kontakt: grzegorz.filo@pk.edu.pl)
- 3 pracownicy Katedry Informatyki Stosowanej (kontakt:)



13 ZATWIERDZENIE KARTY PRZEDMIOTU DO REALIZACJI

(miejscowość, data)

(odpowiedzialny za przedmiot)

(dziekan)

PRZYJMUJĘ DO REALIZACJI (data i podpisy osób prowadzących przedmiot)

.....
.....
.....