



## SYLABUS PRZEDMIOTU

Nazwa przedmiotu:	Technologie internetu
Kod przedmiotu:	TIN
Kierunek / Profil:	Informatyka / praktyczny
Tryb studiów:	niestacjonarny
Rok / Semestr:	1 / 2
Charakter:	obowiązkowy
Odpowiedzialny:	dr Tadeusz Puśniakowski
Wersja z dnia:	19.02.2026

### 1. Godziny zajęć i punkty ECTS

Wykłady	Ćwiczenia	Laboratoria	Z prowadzącym	Praca własna	Łącznie	ECTS
16 h	—	16 h	32 h	68 h	100 h	4

### 2. Forma zajęć

Forma zajęć	Sposób zaliczenia
Laboratorium	Zaliczenie z oceną

### 3. Cel dydaktyczny

Zapoznanie studentów z nowoczesnymi technologiami internetowymi, które są kluczowe dla tworzenia i zarządzania dynamicznymi stronami oraz aplikacjami internetowymi. Zdobycie wiedzy na temat tworzenia responsywnych interfejsów, wykorzystania zaawansowanych technik CSS oraz programowania w języku JavaScript, w tym programowania funkcyjnego i obiektowego. Przedmiot obejmuje także tematykę asynchronicznego programowania, modyfikacji elementów DOM oraz korzystania z narzędzi sieciowych, takich jak HTTP i

WebSocket. Dodatkowo, studenci poznają podstawy pracy z popularnym narzędziem do budowy aplikacji serwerowych.

## 4. Przedmioty wprowadzające

---

Przedmiot	Wymagane zagadnienia
Warsztaty programistyczne	Bazy danych
Znajomość HTML, obsługi bazy danych za pomocą języka zapytań	—

## 5. Treści programowe

---

1. Przegląd technologii internetowych i ich rola w tworzeniu nowoczesnych stron internetowych.
2. Zasady responsywności stron internetowych i ich implementacja.
3. Techniki zaawansowanego stylowania CSS oraz organizacja i optymalizacja kodu.
4. Zastosowanie narzędzi do tworzenia układów i stylizacji stron internetowych.
5. Wprowadzenie do języka JavaScript: zmienne oraz operatory.
6. Funkcje i paradygmat programowania funkcyjnego w języku JavaScript.
7. Programowanie obiektowe w JavaScript oraz manipulacja obiektami.
8. Zarządzanie zdarzeniami w JavaScript oraz modyfikacja struktury DOM.
9. Asynchroniczne programowanie w JavaScript: podstawy i praktyczne zastosowania.
10. Zaawansowane techniki programowania w JavaScript: rozprzestrzenianie i destrukturyzacja.
11. Wprowadzenie do mechanizmów obsługi zdarzeń w aplikacjach internetowych.
12. Obsługa komunikacji sieciowej w JavaScript, w tym zapytania HTTP i mechanizmy komunikacji w czasie rzeczywistym.
13. Podstawy pracy z technologiami serwerowymi oraz budowa aplikacji internetowych.
14. Wprowadzenie do tworzenia aplikacji internetowych z wykorzystaniem popularnych narzędzi serwerowych.

## 6. Efekty kształcenia

---

### Wiedza

- Student rozumie podstawowe technologie internetowe oraz ich rolę w tworzeniu nowoczesnych stron i aplikacji internetowych.
- Student potrafi wyjaśnić zasady responsywności stron internetowych oraz zastosować odpowiednie techniki do ich implementacji.
- Student rozumie działanie preprocesorów CSS i potrafi wykorzystać je do optymalizacji oraz organizacji kodu CSS.

- Student rozumie zasady działania języka JavaScript, w tym zmienne, operatory oraz funkcje, i potrafi zastosować je w praktycznych zadaniach.
- Student potrafi wyjaśnić zasady asynchronicznego programowania w JavaScript i zastosować je w praktyce.
- Student rozumie podstawy działania narzędzi serwerowych, takich jak Node.js i potrafi je zastosować w kontekście budowy aplikacji internetowych.

## Umiejętności

- Student umie zastosować frameworki CSS w celu tworzenia złożonych i estetycznych interfejsów użytkownika.
- Student potrafi wyjaśnić pojęcia programowania funkcyjnego oraz obiektowego w JavaScript i zastosować je w kodzie.
- Student umie pisać i optymalizować kod w języku JavaScript, stosując zmienne, operatory oraz funkcje.
- Student umie wykorzystać narzędzia sieciowe w JavaScript, w tym obsługę protokołów HTTP i WebSocket, do budowy dynamicznych aplikacji internetowych.

## Kompetencje społeczne

- Student jest gotów do efektywnej współpracy w zespole projektowym, wnosząc wkład w tworzenie oraz rozwój aplikacji internetowych.
- Student jest gotów do krytycznego oceniania własnych umiejętności i poszerzania wiedzy w zakresie nowoczesnych technologii internetowych, podejmując inicjatywy samokształcenia.
- Student jest gotów do rozwiązywania problemów programistycznych w sposób kreatywny i efektywny, korzystając z nowoczesnych narzędzi i technologii.
- Student jest gotów do stosowania najlepszych praktyk programistycznych oraz standardów branżowych przy tworzeniu rozwiązań internetowych, dbając o ich jakość i bezpieczeństwo.

## 7. Kryteria oceny

---

- wykład z elementami dyskusji z prezentacją multimedialną
- burza mózgów
- rozwiązywanie zadań
- analiza przypadków
- projekt praktyczny
- Kryteria oceny
- 50% Kolokwium pisemne
- 50% Projekt praktyczny

## 8. Metody dydaktyczne

---

Wykład, laboratoria, praca własna studenta.

## 9. Literatura

---

### Podstawowa:

- D. Herron, Node.js Web Development, Fifth Edition, Apress, 2020.
- E. Brown, Web Development with Node and Express, 2nd Edition, O'Reilly Media, 2019.

### Uzupełniająca:

- F. Zammetti, Modern Full Stack Development using Typescript, React, Node.js, Webpack and Docker, Apress, 2020.