KARTA PRZEDMIOTU

I. Dane podstawowe

Nazwa przedmiotu	Zaawansowane aplikacje sieciowe
Nazwa przedmiotu w języku angielskim	
Kierunek studiów	Informatyka
Poziom studiów (I, II, jednolite magisterskie)	II
Forma studiów (stacjonarne, niestacjonarne)	Stacjonarne
Dyscyplina	Informatyka
Język wykładowy	Polski

Koordynator przedmiotu/osoba odpowiedzialna	Dr Rafał Stęgierski

Forma zajęć *(katalog zamknięty ze słownika)*	Liczba godzin	semestr	Punkty ECTS
wykład			6
konwersatorium	30	I	
ćwiczenia			
laboratorium	30	I	
warsztaty			
seminarium			
proseminarium			
lektorat			
praktyki			
zajęcia terenowe			
pracownia dyplomowa			
translatorium			
wizyta studyjna			

Wymagania wstępne	1. Podstawy algorytmiki i programowania 2. Bazy danych 3. Programowanie aplikacji internetowych

II. Cele kształcenia dla przedmiotu

C1: Opanowanie zaawansowanych rozwiązań fullstack C2: Znajomość Frameworków JS i PHP

III. Efekty uczenia się dla przedmiotu wraz z odniesieniem do efektów kierunkowych

Symbol	Opis efektu przedmiotowego	Odniesienie do efektu kierunkowego
WIEDZA		
W_01	Zna możliwości Javascript i frameworków JS w kontekście wykorzystania ich jako zaplecza i frontonu web	K_W01, K_W04
W_02	Zna składowe poprawnie zaprojektowanego serwisu webowego	K_W02
UMIEJĘTNOŚCI		
U_01	Zna aplikacje pozwalające na tworzenie elementów składowych witryn internetowych	K_U09
U_02	Potrafi korzystać z dokumentacji frameworków i bibliotek wykorzystywanych w procesie budowy witryny	K_U02, K_U09
U_03	Potrafi zaprojektować stronę internetową lub fronton	K_U05, K_U17
KOMPETENCJE SPOŁECZNE		
K_01	Potrafi ocenić jakość opracowanego rozwiązania i porównać z rozwiązaniami alternatywnymi	K_K01, K_K05
K_02	Potrafi wybrać tematykę i sposób realizacji projektu	K_K03, K_K06

IV. Opis przedmiotu/ treści programowe

1.Współdzielenie i praca grupowa nad projektem przy użyciu oprogramowania cvs, svn lub git
2. Zarządzanie projektem z użyciem platform Github i Bitbicket
4. Praca w Visual Studio Code
5. Javascript ES6
6. Architektura rozwiązań fullstack Javascript
7. Aplikacje Node.js
8. Restfull API
9. Użycie AngularJS
10. Wzorce projektowe frontendu aplikacji web
11. Material Design

V. Metody realizacji i weryfikacji efektów uczenia się

Symbol efektu	Metody dydaktyczne *(lista wyboru)*	Metody weryfikacji *(lista wyboru)*	Sposoby dokumentacji *(lista wyboru)*
WIEDZA			
W_01	Wykład konwersatoryjny	Egzamin	
W_02	Wykład konwersatoryjny	Egzamin	
UMIEJĘTNOŚCI			
U_01	Ćwiczenia laboratoryjne, metoda projektu	Przygotowanie projektu	Pliki projektu
U_02	Ćwiczenia laboratoryjne, metoda projektu	Przygotowanie projektu	Pliki projektu
U_03.	Ćwiczenia laboratoryjne, metoda projektu	Przygotowanie projektu	Pliki projektu
KOMPETENCJE SPOŁECZNE			
K_01	Ćwiczenia laboratoryjne		
K_02	Ćwiczenia laboratoryjne		

VI. Kryteria oceny, wagi

Na ocenę 3:

- Korzysta z Gita i managerów projektu.
- Potrafi śledzić zmiany w kodzie.
- Potrafi zaprojektować API aplikacji webowej.
- Zna koncepcję komunikacji asynchronicznej.
- Potrafi zaplanować model danych dla aplikacji.
- Ma wiedzę pozwalającą na stworzenie aplikacji i kontrolera AngularJS

Na ocenę 4:

- Potrafi udokumentować API i skorzystać z narzędzi weryfikujących jego poprawność.
- Zna koncepcję tworzenia aplikacji SPA.
- Potrafi zarządzać projektami Node.js.

Na ocenę 5:

- Potrafi w sposób szeroki korzystać z możliwości środowisk Fullstack.
- Umie zarządzać systemami nierelacyjnych baz danych.
- Zna i korzysta z bibliotek oferujących obsługę Material Design.

VII. Obciążenie pracą studenta

Forma aktywności studenta	Liczba godzin
Liczba godzin kontaktowych z nauczycielem	**90**
Liczba godzin indywidualnej pracy studenta	**70**

VIII. Literatura

Literatura podstawowa
1. Mike Cantelon, Marc Harter, TJ Holowaychuk, Nathan Rajlich, Node.js w akcji, Helion 2. Nicolas Bevacqua,Nowoczesny JavaScript. Poznaj ES6 i praktyczne zastosowania nowych rozwiązań, Helion 3. Dokumentacja AngularJS i Material Design
Literatura uzupełniająca