

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	
Język wykładowy	Polski

Koordinator przedmiotu/osoba odpowiedzialna	Dr Rafał Stęgiński
---	--------------------

Forma zajęć (<i>katalog zamknięty ze słownika</i>)	Liczba godzin	semestr	Punkty ECTS
wykład	30	I/III	6
konwersatorium			
ćwiczenia	30	I/III	
laboratorium			
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	Rozumie współczesne znaczenie informatyki i jej zastosowań	K_W01
W_02	Zna podstawowe algorytmy oraz przykłady ich praktycznej implementacji	K_W03
W_03	Ma podstawową wiedzę z zakresu budowy i zarządzania systemami informatycznymi	K_W04
W_04	Ma ogólną wiedzę z zakresu algorytmiki, projektowania i programowania, systemów operacyjnych, sieci komputerowych, inżynierii oprogramowania, baz danych, sztucznej inteligencji i grafiki komputerowej	K_W06
UMIEJĘTNOŚCI		
U_01	Potrafi samodzielnie pozyskiwać i wykorzystywać informacje pomocne w rozwiązaniu określonych problemów informatycznych z dokumentacji technicznej, plików pomocy oraz zasobów Internetu i dostępnej literatury	K_U02
U_02	Potrafi projektować strony www	K_U05
U_03	U Potrafi stosować mechanizmy wspomaganie decyzji przy rozwiązywaniu problemów praktycznych mie ocenić poprawność aplikacji, zgodność z dokumentacją projektową, zaproponować poprawki uwzględniające zaobserwowane błędy	K_U018
KOMPETENCJE SPOŁECZNE		

IV. Opis przedmiotu/ treści programowe

<ol style="list-style-type: none"> 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 <i>(lista wyboru)</i>	Metody weryfikacji <i>(lista wyboru)</i>	Sposoby dokumentacji <i>(lista wyboru)</i>
WIEDZA			
W_01	Praca pod kierunkiem	Egzamin	Protokół
W_02	Praca pod kierunkiem	Egzamin	Protokół
W_03	Ćwiczenia praktyczne	Egzamin	Protokół

W_04	Ćwiczenia praktyczne	Egzamin	Protokół
UMIEJĘTNOŚCI			
U_01	Ćwiczenia laboratoryjne	Projekt	Karta oceny projektu
U_02	Ćwiczenia laboratoryjne	Projekt	Karta oceny projektu
U_03	Ćwiczenia laboratoryjne	Projekt	Karta oceny projektu
KOMPETENCJE SPOŁECZNE			

VI. Kryteria oceny, wagi...

Projekt korzystający z omawianych rozwiązań.

Egzamin ustny.

Na ocenę dostateczną:

- potrafi zarządzać projektem przy użyciu oprogramowania cvs, svn lub git
- zna podstawowe funkcjonalności platform Github i Bitbicket
- potrafi korzystać z Visual Studio Code
- potrafi tworzyć proste skrypty Javascript ES6
- zna architekturę rozwiązań fullstack Javascript
- zna wzorce projektowe frontendu aplikacji web

Na ocenę dobrą:

- potrafi zarządzać zespołem z użyciem platform Github i Bitbicket
- korzysta z dowolnych bibliotek Javascript ES6
- potrafi wdrażać architekturę rozwiązań fullstack Javascript
- zna składnię Node.js
- rozumie Restfull API

Na ocenę bardzo dobrą:

- swobodnie korzysta z Javascript ES6
- potrafi projektować aplikacje oparte o AngularJS
- zna wzorce projektowe frontendu aplikacji web
- zna koncepcje związane z projektowaniem opartym o Material Design

Obciążenie pracą studenta

Forma aktywności studenta	Liczba godzin
Liczba godzin kontaktowych z nauczycielem	50
Liczba godzin indywidualnej pracy studenta	40

VII. Literatura

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