


KAPITAŁ LUDZKI
 NARODOWA STRATEGIA SPÓJNOŚCI

 Projekt współfinansowany przez
 Unię Europejską w ramach
 Europejskiego Funduszu
 Społecznego

UNIA EUROPEJSKA
 EUROPEJSKI
 FUNDUSZ SPOŁECZNY


Nazwa przedmiotu		Kod ECTS	
Technologie Front-End		11.3.1889	
Nazwa jednostki prowadzącej przedmiot			
Instytut Informatyki			
Studia			
wydział	kierunek	poziom	pierwszego stopnia
Wydział Matematyki, Fizyki i Informatyki	Informatyka	forma	stacjonarne
		moduł	wszystkie
		specjalnościowy	wszystkie
		specjalizacja	wszystkie
Nazwisko osoby prowadzącej (osób prowadzących)			
mgr Mateusz Miotk			
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS	
Formy zajęć		6 Przedmiot w wymiarze 30h wykładu i 30h lab. + praca własna studenta	
Wykład, Ćw. laboratoryjne			
Sposób realizacji zajęć			
zajęcia w sali dydaktycznej			
Liczba godzin			
Wykład: 30 godz., Ćw. laboratoryjne: 30 godz.			
Termin realizacji przedmiotu			
2022/2023 letni			
Status przedmiotu		Język wykładowy	
fakultatywny (do wyboru)		polski	
Metody dydaktyczne		Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne	
<ul style="list-style-type: none"> - Rozwiązywanie zadań - Wykład z prezentacją multimedialną - laboratorium - pisanie i testowanie programów projekt praktyczny 		Sposób zaliczenia	
		<ul style="list-style-type: none"> - Zaliczenie na ocenę - Egzamin 	
		Formy zaliczenia	
		<ul style="list-style-type: none"> - wykonanie pracy zaliczeniowej - projekt lub prezentacja - egzamin pisemny z pytaniami (zadaniami) otwartymi - egzamin pisemny testowy - ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru - kolokwium - wykonanie pracy zaliczeniowej - wykonanie określonej pracy praktycznej 	
		Podstawowe kryteria oceny	
		Wykład: <ul style="list-style-type: none"> • (50%) - Ocena z laboratorium • (50%) - Ocena z egzaminu Laboratorium: <ul style="list-style-type: none"> • (20%) - Zadania praktyczne • (40%) - Kolokwium • (40%) - Projekt programistyczny 	
Sposób weryfikacji założonych efektów uczenia się			

zakładany efekt kształcenia	egzamin	odpowiedź ustna	program/projekt	kolokwium	referat	raport	aktywność w dyskusji	obserwacja postawy studenta
Wiedza								
K_W05	X		X	X				
K_W08	X		X	X				
P_W1	X		X	X				
P_W2	X		X	X				
P_W3	X		X	X				
Umiejętności								
K_U02	X		X	X				X
K_U06	X		X	X				
P_U1	X		X	X				X
P_U2	X		X	X				
P_U3	X		X	X				
Kompetencje								
K_K01								X
K_K03							X	X
P_K1							X	X

Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi

A. Wymagania formalne

Zaliczony przedmiot Aplikacje bazodanowe.

B. Wymagania wstępne

Podstawy programowania w języku JavaScript. Umiejętność tworzenia statycznych stron internetowych za pomocą języka znaczników HTML oraz kaskadowych arkuszy stylów CSS.

Cele kształcenia

Zapoznanie studentów z projektowaniem i tworzeniem aplikacji typu SPA (single-page applications) na przykładzie React.js.

Treści programowe

Treści wykładów (z podziałem na tygodnie):

1. Historia frontendu. Język znaczników HTML i CSS
2. Preprocesory CSS: Less oraz Sass
3. Język JavaScript jako język budowania interfejsów webowych
4. Programowanie funkcyjne w języku JavaScript
5. Architektura aplikacji SPA: Serwer Node.js
6. Zarządzanie serwerem Node.js. Wprowadzenie do frameworka React.js
7. JSX w React.js. Integracja CSS w aplikacji
8. Pojęcie stanu aplikacji oraz wprowadzenie do hooków w React.js
9. Formularze oraz ich walidacja w React.js. Narzędzie Formik.
10. Inne rodzaje hooków w React.js
11. Integracja danych w React.js
12. Trasowanie aplikacji na podstawie pakietu React Router
13. Testowanie aplikacji frontendowych
14. Technologia suspense w React.js
15. Przegląd najciekawszych pakietów używanych w aplikacjach React.js

Wykaz literatury

Literatura podstawowa (wymagana do ostatecznego zaliczenia zajęć, zdania egzaminu wykorzystywana podczas zajęć oraz studiowana samodzielnie przez studenta):

- Node.js, <https://nodejs.org/en/>
- React – a javascript library for building user interfaces, <https://reactjs.org/>
- Reactjs tutorial - tutorialspoint, <https://www.tutorialspoint.com/reactjs/index.htm>
- W3schools online web tutorials, <https://www.w3schools.com/>
- Frontend developer handbook - <https://frontendmasters.com/books/front-end-handbook>

- E. Porcello and A. Banks, Learning react: Modern patterns for developing react apps, O'Reilly Media, Incorporated, 2020.
- A. Prabhu, Beggining CSS Preprocessors. With Sass, Compass and Less, Apress, 2015.
- C. Rippon, Learn react with typescript 3: Beginner's guide to modern react web development with typescript 3, Packt Publishing, 2018.
- M.T. Thomas, React in action, Manning Publications, 2018.
- F. Zammetti, Modern full-stack development: Using typescript, react, node.js, webpack, and docker, Apress, 2020.

Literatura uzupełniająca:

- A. Boduch and R. Derks, React and react native: A complete hands-on guide to modern web and mobile development with react.js, Packt Publishing, 2020.
- E. Brown, Web Development with Node and Express, O'Reilly Media, Incorporated, 2020.
- A. Freeman, Pro react 16, Apress, 2019.
- D. Griffiths and D. Griffiths, React Cookbook. Recipes for Mastering the React Framework, O'Reilly, 2021.
- G. Lim, Beginning react with hooks, Independently Published, 2020.
- C.S. Roldán, React cookbook: Create dynamic web apps with react using redux, webpack, node.js, and graphql, Packt Publishing, 2018.
- R. Wieruch, The road to react: Your journey to master plain yet pragmatic react.js, Independently Published, 2017.

Kierunkowe efekty uczenia się K_W05: ma ogólną wiedzę na temat różnych paradygmatów programowania i języków programowania; szczegółowo zna metody i wzorce projektowania i programowania obiektowego K_W08: ma wiedzę na temat technologii sieciowych, w tym podstawowych protokołów komunikacyjnych, bezpieczeństwa i budowy aplikacji sieciowych K_U02: potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania K_U06: potrafi projektować, tworzyć, uruchamiać i testować programy przy wykorzystaniu dedykowanych narzędzi oraz adekwatnych wzorców K_K01: zna ograniczenia własnej wiedzy i rozumie potrzebę dalszego uczenia się K_K03: potrafi i jest gotów formułować opinie na temat podstawowych zagadnień informatycznych	Wiedza Student zna klasyczne problemy i pojęcia z programowania współbieżnego, wybrane techniki programistyczne służące do tworzenia programów współbieżnych oraz wybrane narzędzia teoretyczne służące do wnioskowania o ich poprawności Efekty przedmiotowe P_W1: Student zna i rozumie pojęcia związane z pojęciem Frontend Development. Student zna i rozumie pojęcie preprocesora CSS oraz tworzenia aplikacji webowej z wykorzystaniem narzędzia Node.js oraz React.js (K_W05, K_W08). P_W2: Student zna i rozumie elementy programowania funkcyjnego na przykładzie języka JavaScript. Student zna i rozumie techniki oraz metody programistyczne wykorzystywane w języku JavaScript (K_W05, K_W08). P_W3: Student zna i rozumie pojęcia z zakresu tworzenia interaktywnych aplikacji internetowych, umożliwiających komunikację typu człowiek-komputer (K_W05, K_W08).
	Umiejętności Student potrafi zaprojektować i zbudować aplikacji typu SPA za pomocą narzędzia React i sprawdzić jego poprawność. Efekty przedmiotowe: P_U1: Student potrafi ocenić przydatność różnych podejść programistycznych na podstawie języka JavaScript i związanych z nimi środowisk na przykładzie narzędzia Node.js (K_U02, K_U06). P_U2: Student potrafi wyspecyfikować, zaprojektować, zaimplementować, przetestować oraz zdebugować aplikację webową; potrafi korzystać z bibliotek przy użyciu narzędzia npm, środowisk programistycznych, integrujących i uruchomieniowych (K_U02, K_U06). P_U3: Student potrafi wytworzyć warstwową aplikację webową w oparciu o wybrane wzorce architektoniczne, przy pomocy narzędzia Node.js oraz React.js (K_U02, K_U06).
	Kompetencje społeczne (postawy) Student jest w stanie współpracować z zespołem programistów tworzących wspólnie program współbieżny. Potrafi operować pojęciami w zakresie programowania aplikacji SPA w sposób umożliwiający mu porozumienie się z innymi informatykami w tym zakresie. Efekty przedmiotowe P_K1: potrafi formułować wypowiedzi na temat programowania aplikacji typu SPA i rozumie konieczność dalszego kształcenia się (K_K01, K_K03)
Kontakt mateusz.miotk@ug.edu.pl	