



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez
Unię Europejską w ramach
Europejskiego Funduszu
Społecznego

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Nazwa przedmiotu		Kod ECTS	
Frontend development		11.3.0792	
Nazwa jednostki prowadzącej przedmiot			
Instytut Informatyki			
Studia			
wydział	kierunek	poziom	drugiego stopnia
Wydział Matematyki, Fizyki i Informatyki	Informatyka	forma	stacjonarne
		moduł	wszystkie
		specjalnościowy	wszystkie
		specjalizacja	wszystkie
Nazwisko osoby prowadzącej (osób prowadzących)			
dr Jakub Neumann; mgr Tadeusz Puźniakowski			
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS	
Formy zajęć		6 Przedmiot w wymiarze 30h wykładu i 30h laboratorium + praca własna studenta.	
Wykład, Ćw. laboratoryjne			
Sposób realizacji zajęć			
zajęcia w sali dydaktycznej			
Liczba godzin			
Ćw. laboratoryjne: 30 godz., Wykład: 30 godz.			
Cykl dydaktyczny			
2016/2017 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"> - Dyskusja - Metoda projektów (projekt badawczy, wdrożeniowy, praktyczny) - Projektowanie doświadczeń - Wykonywanie doświadczeń - Wykład z kodowaniem fragmentów programów na żywo - Wykład z prezentacją multimedialną 		Sposób zaliczenia	
		<ul style="list-style-type: none"> - Zaliczenie na ocenę - Egzamin 	
		Formy zaliczenia	
		<ul style="list-style-type: none"> - egzamin ustny - wykonanie pracy zaliczeniowej - projekt lub prezentacja - ustalenie oceny zaliczeniowej na podstawie ocen częściowych otrzymywanych w trakcie trwania semestru 	
		Podstawowe kryteria oceny	
		Ocenie podlegają zrozumienie i umiejętności praktyczne posługiwania się omawianymi technikami i technologiami. Ocenie podlegają projekty, które student wykona podczas zajęć.	
Sposób weryfikacji założonych efektów kształcenia			

zakładany efekt kształcenia	Dyskusja	Projektowanie doświadczeń	Metoda projektów (projekt badawczy, wdrożeniowy, praktyczny)	Wykonywanie doświadczeń	Wykład z prezentacją multimedialną	Wykonywanie doświadczeń	Wykład z kodowaniem fragmentów programów na żywo
Wiedza							
K_W02	x				x		x
K_W05	x				x		x
K_W09	x				x		x
Umiejętności							
K_U02		x	x	x		x	
K_U06		x	x	x		x	
K_U07		x	x	x		x	
Kompetencje							
K_K01	x				x		
K_K03	x				x		

Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi

A. Wymagania formalne

Bardzo dobra znajomość protokołu HTTP i zasad funkcjonowania aplikacji webowych klient-serwer. Umiejętność projektowania i programowania obiektowego, dobra znajomość HTML, CSS i języka JavaScript.

B. Wymagania wstępne

Brak

Cele kształcenia

Celem przedmiotu jest zapoznanie słuchaczy z nowoczesnymi technologiami, technikami i narzędziami przeznaczonymi do wytwarzania części klienckiej aplikacji webowej, zbudowanej w modelu Single Page Application (SPA) i opartej na języku JavaScript. Obecnie obserwujemy bardzo wiele zmian zarówno w samym języku JavaScript/ECMAScript jak i w rozwoju technologii towarzyszącym wytwarzaniu tego typu aplikacji: rozwój języków takich jak TypeScript, rozwój narzędzi do kompilacji/transpilacji do JavaScriptu, rozwój frameworków takich jak Angular 2+, rozwój bundlerów takich jak webpack, rozwój narzędzi do testowania w różnych odmianach.

Treści programowe

Wykład zapoznaje słuchaczy z nowoczesnymi technologiami, technikami i narzędziami przeznaczonymi do wytwarzania części klienckiej aplikacji webowej, zbudowanej w modelu Single Page Application (SPA) i opartej na języku JavaScript. Omawiać będziemy zmiany zachodzące zarówno w samym języku JavaScript/ECMAScript jak i w rozwoju technologii towarzyszącym wytwarzaniu tego typu aplikacji. W szczególności treści zajęć przedstawiają się następująco:

- nowoczesny JavaScript - "good parts" w ES5 oraz nowości w ES6
- NodeJs jako podstawowe środowisko wytwórcze aplikacji SPA
- język TypeScript
- narzędzia typu transpiler (Babel)
- biblioteka RxJS i zasady programowania reaktywnego (wg ReactiveX)
- bundler na przykładzie webpack
- framework Angular 2
- narzędzi do testowania (Protractor, Karma, Jasmine)

Wykaz literatury

- Dokumentacja techniczna omawianych technologii, liczne tutoriale, artykuły w internecie, darmowe kursy na platformach e-learningowych

Efekty kształcenia

(obszarowe i kierunkowe)

K_W02 ma pogłębioną wiedzę na temat podstawowych paradygmatów programowania; zna również aktualne trendy w językach programowania

K_W05 Zna najważniejsze konstrukcje programistyczne oraz struktury danych

K_W06 zna zaawansowane metody projektowania i analizowania złożoności obliczeniowej algorytmów, zna zasady działania oraz praktycznego zastosowania

Wiedza

- K_W02 ma pogłębioną wiedzę na temat podstawowych paradygmatów programowania; zna również aktualne trendy w językach programowania, zna również aktualne trendy w językach programowania, w towarzyszącym im technologiom i frameworkom, ze szczególnym uwzględnieniem środowiska wytwórczego aplikacji webowych modelu SPA
- K_W05 Zna najważniejsze konstrukcje programistyczne oraz struktury danych, szczególnie związane z językami ES5 i ES6 oraz TypeScript
- K_W09 zna biegle co najmniej dwa języki programowania oraz biblioteki algorytmów i struktur danych oraz ma wiedzę na temat praktycznych

<p>najważniejszych algorytmów różnego typu w sensie ich treści jak i sposobu ich wykonywania</p> <p>K_W09 zna biegle co najmniej dwa języki programowania oraz biblioteki algorytmów i struktur danych oraz ma wiedzę na temat praktycznych uwarunkowań wydajnych implementacji algorytmów</p> <p>K_U02 ma umiejętność projektowania abstrakcyjnych struktur danych i ich wydajnych implementacji</p> <p>K_U06 projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz programuje algorytmy z wykorzystaniem różnych technik programistycznych</p> <p>K_U07 potrafi zastosować znane algorytmy w konkretnych sytuacjach, potrafi efektywnie dobrać rodzaj i sposób wykonania algorytmu w zależności od postawionego problemu</p>	<p>uwarunkowań wydajnych implementacji algorytmów szczególnie w zakresie frameworków aplikacji webowych typu SPA</p>
	<p>Umiejętności</p> <ul style="list-style-type: none"> • K_U02 ma umiejętność projektowania abstrakcyjnych struktur danych i ich wydajnych implementacji przy wykorzystaniu omawianych języków, technik programistycznych, technologii i frameworków środowiska wytwórczego aplikacji webowych w modelu SPA • K_U06 projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz programuje algorytmy z wykorzystaniem różnych technik programistycznych w odniesieniu do aktualnie omawianego języka programowania lub w kontekście omawianej technologii lub frameworka • K_U07 potrafi zastosować znane algorytmy w konkretnych sytuacjach, potrafi efektywnie dobrać rodzaj i sposób wykonania algorytmu w zależności od postawionego problemu w odniesieniu do aktualnie omawianego języka programowania i jego paradygmatu a także w kontekście omawianej technologii lub frameworka uwzględniając ich ograniczenia
	<p>Kompetencje społeczne (postawy)</p> <ul style="list-style-type: none"> • K_K01 rozumie potrzebę dalszego kształcenia, zarówno w doskonaleniu warsztatu programistycznego jak i w odniesieniu do zmieniających się trendów i nowych koncepcji wytwarzania aplikacji webowych • K_K03 potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego rozumowania danego tematu lub odnalezieniu brakujących elementów rozumowania szczególnie w odniesieniu do często zmieniających się technologii i frameworków środowiska wytwórczego aplikacji webowych. Dyskusja i liczne przykłady kodowania na żywo wspierają i zachęcają Studenta do aktywnego uczestnictwa w wykładzie
<p>Kontakt</p> <p>kuba@inf.ug.edu.pl</p>	