



**KAPITAŁ LUDZKI**  
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez  
Unię Europejską w ramach  
Europejskiego Funduszu  
Społecznego

**UNIA EUROPEJSKA**  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



<b>Nazwa przedmiotu</b>		<b>Kod ECTS</b>	
Frontend development		11.3.1057	
<b>Nazwa jednostki prowadzącej przedmiot</b>			
Instytut Informatyki			
<b>Studia</b>			
<b>wydział</b>	<b>kierunek</b>	<b>poziom</b>	<b>pierwszego stopnia</b>
Wydział Matematyki, Fizyki i Informatyki	Informatyka	<b>forma</b>	stacjonarne
		<b>moduł</b>	wszystkie
		<b>specjalnościowy</b>	wszystkie
		<b>specjalizacja</b>	wszystkie
<b>Nazwisko osoby prowadzącej (osób prowadzących)</b>			
dr Jakub Neumann; dr Andrzej Borzyszkowski			
<b>Formy zajęć, sposób ich realizacji i przypisana im liczba godzin</b>		<b>Liczba punktów ECTS</b>	
<b>Formy zajęć</b>		4	
Wykład, Ćw. laboratoryjne			
<b>Sposób realizacji zajęć</b>			
zajęcia w sali dydaktycznej			
<b>Liczba godzin</b>			
Ćw. laboratoryjne: 30 godz., Wykład: 15 godz.			
<b>Termin realizacji przedmiotu</b>			
2020/2021 zimowy			
<b>Status przedmiotu</b>		<b>Język wykładowy</b>	
obowiązkowy		polski	
<b>Metody dydaktyczne</b>		<b>Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne</b>	
<ul style="list-style-type: none"> <li>- wykład z prezentacją multimedialną</li> <li>- ćwiczenia laboratoryjne - projektowanie doświadczeń</li> <li>- ćwiczenia laboratoryjne -- sporządzanie i uruchamianie programów komputerowych</li> </ul>		<b>Sposób zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- Zaliczenie na ocenę</li> <li>- Zaliczenie (zał)</li> </ul>	
		<b>Formy zaliczenia</b>	
		ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru	
		<b>Podstawowe kryteria oceny</b>	
		<p>Przedmiot kończy się zaliczeniem.</p> <p>Warunkiem uzyskania zaliczenia przedmiotu jest uzyskanie pozytywnych ocen z ćwiczeń laboratoryjnych.</p> <p>Ocenie podlegają zrozumienie i umiejętności praktyczne postępowania się omawianymi technikami i technologiami oraz cenie podlegają projekty, które student wykona podczas zajęć.</p>	
<b>Sposób weryfikacji założonych efektów kształcenia</b>			

zakładany efekt kształcenia	egzamin	kolokwium	projekt	referat	raport	aktywność	obserwacja postawy i umiejętności
	Wiedza						
K_W07			X				
	Umiejętności						
K_U04			X				
K_U09			X				
	Kompetencje						
K_K02							X

### Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi

#### A. Wymagania formalne

Brak wymagań formalnych

#### B. Wymagania wstępne

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

### 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 poświęcony będzie na 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. 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

A. Literatura wymagana do ostatecznego zaliczenia zajęć (zdania egzaminu):

A.1. wykorzystywana podczas zajęć

A.2. studiowana samodzielnie przez studenta

B. Literatura uzupełniająca

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

### Kierunkowe efekty kształcenia

P6S\_WG K\_W07 P6S\_UW P6S\_UK P6S\_UU K\_U04  
K\_U09 P6S\_KK K\_K02

#### Wiedza

ma wiedzę w zakresie projektowania, wytwarzania, testowania, wdrażania i utrzymania aplikacji webowych oraz ich bezpieczeństwa

#### Umiejętności

potrafi tworzyć, uruchamiać i testować programy przy wykorzystaniu dedykowanych narzędzi oraz wzorców projektowych potrafi zgodnie z zadaną specyfikacją zaprojektować oraz zrealizować system informatyczny

#### Kompetencje społeczne (postawy)

potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania

### Kontakt

[j.neumann@inf.ug.edu.pl](mailto:j.neumann@inf.ug.edu.pl)