



## KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Aplikacje internetowe [S1S1E>APINT]

### Przedmiot

Kierunek studiów

Sztuczna inteligencja/Artificial Intelligence

Rok/Semestr

3/5

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

angielski

Forma studiów

stacjonarne

Wymagalność

obligatoryjny

### Liczba godzin

Wykład

30

Laboratorium

30

Inne

0

Ćwiczenia

0

Projekty/seminaria

0

### Liczba punktów ECTS

5,00

### Koordynatorzy

dr inż. Maciej Piernik

maciej.piernik@put.poznan.pl

### Wykładowcy

### Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z zakresu protokołów sieciowych, systemów baz danych i programowania zorientowanego obiektowo. Powinien posiadać umiejętność programowania prostych aplikacji z wykorzystaniem zintegrowanych środowisk programistycznych. Powinien również rozumieć konieczność poszerzania swoich kompetencji / mieć gotowość do podjęcia współpracy w ramach zespołu.

### Cel przedmiotu

1. Uzyskanie wiedzy dotyczącej podstawowych pojęć i koncepcji z zakresu implementacji dokumentów WWW i aplikacji WWW, niezbędnej do rozróżniania podstawowych architektur aplikacji internetowych oraz metod implementacji ich modułów. 2. Uporządkowanie wiedzy z zakresu architektur sieciowych, protokołów sieciowych, bezpieczeństwa systemów rozproszonych. 3. Nabycie umiejętności implementacji aplikacji WWW z wykorzystaniem podstawowych technologii implementacji interfejsu użytkownika, m.in. HTML, CSS, JavaScript, XML+XSL, technologii implementacji logiki prezentacji, frameworków Single Page Application, technologii implementacji logiki biznesowej. 4. Pozyskanie kompetencji społecznych dotyczących zespołowego przygotowania projektu, w tym organizacji pracy zespołowej, a zwłaszcza przywództwa oraz komunikatywności w procesie grupowego rozwiązywania problemu.

## Przedmiotowe efekty uczenia się

### Wiedza:

ma uporządkowaną wiedzę dotyczącą tworzenia aplikacji WWW (K1st\_W2)  
zna podstawowe techniki, metody, algorytmy oraz narzędzia wykorzystywane w procesie projektowania, tworzenia oraz wdrażania aplikacji WWW (K1st\_W4)  
ma podstawową wiedzę o cyklu życia oraz procesach zachodzących w programowych i sprzętowych systemach informatycznych z zakresu projektowania, implementacji i wdrażania aplikacji WWW (K1st\_W7)  
ma wiedzę nt. bezpieczeństwa, zagrożeń oraz zagadnień etycznych związanych z tworzeniem oraz wykorzystaniem aplikacji WWW (K1st\_W9)

### Umiejętności:

posiada umiejętności informatyczne w zakresie programowania aplikacji WWW z użyciem popularnych języków (K1st\_U2)  
potrafi zaprojektować - zgodnie z zadaną specyfikacją - oraz zaimplementować algorytmy aplikacji WWW, dobierając i stosując dostępne metody, techniki i narzędzia informatyczne (K1st\_U8)  
ma umiejętność prostej adaptacji istniejących oraz formułowania i implementacji nowych mechanizmów w aplikacjach WWW (K1st\_U9)  
potrafi zaprojektować aplikację WWW przetwarzającą w bezpieczny sposób dane różnego typu w sposób umożliwiający przeprowadzanie analiz umożliwiających wyciąganie wniosków przydatnych do rozwiązywania problemów biznesowych (K1st\_U10)

### Kompetencje społeczne:

rozumie, że w świecie aplikacji WWW wiedza i umiejętności bardzo szybko stają się przestarzałe, dostrzegając przy tym potrzebę ciągłego doskonalenia oraz podnoszenia własnych kompetencji (K1st\_K1)  
potrafi myśleć i działać w sposób przedsiębiorczy, m.in. znajdując komercyjne zastosowania dla stworzonych aplikacji WWW, mając na uwadze nie tylko korzyści ekonomiczne, ale również aspekty prawne i społeczne (K1st\_K5)

## Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

### Ocena formująca:

- w zakresie wykładów: - na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach;  
- w zakresie laboratorium: - na podstawie oceny bieżącego postępu realizacji zadań.

### Ocena podsumowująca:

Sprawdzanie założonych efektów kształcenia realizowane jest przez:

- w zakresie laboratorium - ocenę i obronę zadań wykonanych przez studenta w trakcie semestru oraz projektu końcowego  
- w zakresie wykładu - ocenę wiedzy i umiejętności wykazanych na zaliczeniu pisemnym o formie testu.

## Treści programowe

1. Różne podejścia do zagadnienia logiki prezentacji na różnych platformach
  2. Sposoby implementacji logiki biznesowej umożliwiającej obsłużenie wielu typów aplikacji i interfejsów na najpopularniejszych platformach
  3. Projektowanie aplikacji opartych o dane; organizacja warstwy dostępu do danych w sposób umożliwiający jej ponowne wykorzystanie, skalowalność, wydajność i bezpieczeństwo; przykłady na różnych platformach
  4. Infrastruktura aplikacji www; najpopularniejsze architektury
  5. Servery HTTP
  6. Zaawansowane zagadnienia interfejsu użytkownika: preprocesory CSS, responsive web design, frameworki front-end"owe
  7. Tworzenia aplikacji typu Single Page Application, zaawansowane zagadnienia JavaScript, przetwarzanie asynchroniczne
  8. Uwierzytelnianie i autoryzacja w aplikacjach webowych; najważniejsze ataki oraz jak się przed nimi bronić
- Cześć wymienionych wyżej treści programowych realizowana jest w ramach pracy własnej studenta.

## Tematyka zajęć

1. HTML, CSS, responsive design, preprocesory CSS
2. JavaScript
3. React
4. Node.js
5. Architektura MVC
6. Aplikacje internetowe czasu rzeczywistego

## Metody dydaktyczne

1. wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy, rozwiązywanie zadań, demonstracja.
2. ćwiczenia laboratoryjne: rozwiązywanie zadań, ćwiczenia praktyczne, warsztaty

## Literatura

### Podstawowa

1. J. Duckett, Web Design with HTML, CSS, JavaScript and jQuery, Wiley, 2014.
2. B. Sholtz, A. Tijms, The Definitive Guide to JSF in Java EE 8: Building Web Applications with JavaServer Faces, Apress, 2018.
3. K. Hadlock, Ajax for Web Application Developers, Sams Publishing, 2006.
4. J. Liberty, D. Hurwitz, B. MacDonald, Learning ASP.NET 2.0 with AJAX: A Practical Hands-on Guide, O'Reilly, 2007.

### Uzupełniająca

1. E. Jendrock, I. Evans, D. Gollapudi, K. Haase, C. Srivathsa, The Java EE 6 Tutorial, Oracle, 2010.

## Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	125	5,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	60	2,50
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	65	2,50