

# SYLABUS ZAJ /GRUPY ZAJ

## Dane ogólne:

Jednostka organizacyjna:	Katedra Automatyki i Robotyki				
Kierunek studiów:	Automatyka i robotyka				
Specjalno /Specjalizacja:					
Nazwa zaj / grupy zaj :	Informatyka aplikacji Internetu				
Course / group of courses:	Information Technology				
Forma studiów:	stacjonarne				
Nazwa katalogu:	WP-AR-I-20/21Z				
Nazwa bloku zaj :					
Kod zaj /grupy zaj :	104661	Kod Erasmus:			
Punkty ECTS:	4	Rodzaj zaj :		obowi zkowy	
Rok studiów:	2	Semestr:		3	
Rok	Semestr	Forma zaj	Liczba godzin	Forma zaliczenia	ECTS
2	3	LO	30	Zaliczenie z ocen	3
		W	15	Zaliczenie z ocen	1
Razem			45		4
Koordinator:	magister in ynier Dariusz Piwko				
Prowadz cy zaj cia:					
J zyk wykładowy:	semestr: 3 - j zyk polski				

## Obja nienia:

Rodzaj zaj : obowi zkowe, do wyboru.

Forma prowadzenia zaj : W - wykład, - wiczenia audytoryjne, L - lektorat, S – seminarium/ zaj cia seminaryjne, P - wiczenia praktyczne (w tym zaj cia wf), M - wiczenia specjalistyczne (medyczne/ kliniczne), LO – wiczenia laboratoryjne, LI - laboratorium informatyczne, ZTI - zaj cia z technologii informacyjnych, P – wiczenia projektowe, ZT – zaj cia terenowe, T - wiczenia terenowe na obozach programowych, SK - samokształcenie (i inne), PR - praktyka zawodowa

## Dane merytoryczne

Wymagania wst pne:			
Znajomo podstaw programowania oraz terminologii z nim zwi zanej. Sprawne posługiwanie si elementarnymi instrukcjami programistycznymi.			
Szczegółowe efekty uczenia si			
Lp.	Student, który zaliczył zaj cia zna i rozumie/potrafi/jest gotowy do:	Kod efektu dla kierunku studiów	Sposób weryfikacji efektu uczenia si
1	Posiada wiedz na temat stosowanych technologii w programowaniu aplikacji WWW.	AR1_W10	kolokwium, ocena aktywno ci
2	Potrafi wykorzysta dotychczasow wiedz do łatwego przyswojenia nowych technologii.	AR1_W11	ocena aktywno ci, obserwacja zachowa
3	Potrafi wykorzysta dotychczasow wiedz do łatwego przyswojenia nowych technologii	AR1_U07	ocena aktywno ci, obserwacja zachowa
4	Potrafi korzysta z dokumentacji technicznej standardowych pakietów/bibliotek oraz komponentów.	AR1_U10, AR1_U01	kolokwium, ocena aktywno ci, obserwacja zachowa

5	Potrafi praktycznie zastosować poznane technologie do implementacji aplikacji WWW.	AR1_U10, AR1_U11	kolokwium, ocena aktywności
6	Potrafi zaprojektować aplikację WWW w oparciu o wzorzec projektowy MVC.	AR1_U11	kolokwium, ocena aktywności
7	Potrafi współpracować w rozproszonym zespole nad wspólnym kodem.	AR1_U14, AR1_U07	ocena aktywności
8	Posiada usystematyzowaną podstawową wiedzę, która pozwoli na poznanie nowszych technologii.	AR1_U15	ocena aktywności, obserwacja zachowa
<b>Stosowane metody osiągnięcia zakładanych efektów uczenia się (metody dydaktyczne)</b>			
metody podające (Przedstawianie treści na wykładzie w oparciu o prezentację multimedialną oraz analizę i omawianie praktycznych przykładów.), metody praktyczne (ćwiczenia laboratoryjne prowadzone na podstawie instrukcji (udostępnianych w sposób elektroniczny) z przykładami i zadaniami do samodzielnej realizacji, które podlegają ocenie.)			
<b>Kryteria oceny i weryfikacji efektów uczenia się</b>			
<b>wiedza:</b> ocena kolokwium (ocena kolokwium) obserwacja zachowa (obserwacja zachowa indywidualnych i zespołowych) ocena aktywności (ocena aktywności na zajęciach) <b>umiejętności:</b> ocena kolokwium (ocena kolokwium) obserwacja zachowa (obserwacja zachowa indywidualnych i zespołowych) ocena aktywności (ocena aktywności na zajęciach)			
<b>Warunki zaliczenia</b>			
Wykład: Zaliczenie na podstawie obecności. Laboratorium: Zaliczenie z oceną wystawioną na podstawie pozytywnego zaliczenia kolokwium z zadania, aktywności na zajęciach (w rozwijaniu zadań i problemów). Ocena końcowa jest średnią ocen częściowych uzyskanych z w/w zadania. Zaliczanie zajęć jest oceniane zgodnie ze skalą ocen określoną w Regulaminie Studiów PWSZ. Kolokwium przeprowadzone na laboratorium oceniane jest w poniższej skali (zaokrąglenie na korzyść studenta) Skala ocen: 5.0 (bdb) 100 - 91% 4.5 (+db) 90% - 81% 4.0 (db) 80% - 71% 3.5 (+dst) 70% - 61% 3.0 (dst) 60% - 51% 2.0 (ndst) <50% Ocena końcowa jest to średnia ocen z kolokwium przy zachowaniu w/w skali.			
<b>Treści programowe (opis skrócony)</b>			
1. Podstawy wersjonowania kodu na przykładzie aplikacji internetowej. 2. Tworzenie aplikacji internetowych ? dobre praktyki. 3. Skryptowe języki programowania i ich frameworki. 4. Przegląd technologii internetowych, zasady projektowania aplikacji, modele aplikacji. 5. Zagadnienia dostępności oraz użyteczności aplikacji i stron internetowych. Korzystanie ze specyfikacji i dokumentacji technicznej. 6. Bezpieczeństwo danych aplikacji internetowych. 7. Praca z mockup'ami.			
<b>Content of the study programme (short version)</b>			
1. The basics of versioning the code on the example of an internet application. 2. Creating internet applications - good practices. 3. Script programming languages and their frameworks. 4. Overview of internet technologies, principles of application design, application models. 5. Issues of accessibility and usability of applications and websites. Using specifications and technical documentation. 6. Data security of internet applications. 7. Working with mockups.			
<b>Treści programowe</b>			
			Liczba godzin
Semestr: 3			
Forma zajęć : <b>wykład</b>			
1. Podstawy wersjonowania kodu. 2. Przegląd technologii aplikacji internetowych oraz zasady tworzenia i projektowania aplikacji.			15

3. Wprowadzenie do technologii HTML oraz CSS. 4. Języki skryptowe (np. JavaScript). 5. Przegląd środowisk oraz frameworków do tworzenia aplikacji internetowych (np. Angular, ReactJS, NodeJS). 6. Zasady responsywności, stylizacja aplikacji. Zastosowanie dodatkowych bibliotek i frameworków (np. Bootstrap) 7. Podstawy protokołu HTTP/S. Omówienie kodów zapytań 8. Zasady tworzenia aplikacji użytecznej i dostępnej. Przedstawienie specyfikacji ze wskazówkami tworzenia stron internetowych (np. W3C, WAI-ARIA). 9. Udoskonalanie aplikacji webowej zgodnie z zasadami User Experience. 10. Błąd aplikacji webowych zagrażający bezpieczeństwu danych 11. Praca z mockup'ami. Tworzenie uniwersalnych interfejsów użytkownika. 12. Narzędzia ułatwiające tworzenie aplikacji webowych	15
Forma zajęć: <b>wiczenia laboratoryjne</b>	
Realizacja treści wykładu na wiczeniach laboratoryjnych.	30
<b>Literatura</b>	
Podstawowa	
Chris Aquino, Todd Gandee, Podręcznik frontendowca. The Big Nerd Ranch Guide, Helion 2017	
Christopher Schmitt, Kyle Simpson, HTML5 Cookbook, Helion 2011	
Michał Miszczyszyn, TypeScript na poważnie, Type of Web, Gdańsk 2020	
Peter Gasston, CSS3. Podręcznik nowoczesnego webdevelopera, Helion 2015	
Dokumentacja techniczna dostępna w sieci Internet opisująca użyte frameworki podczas zajęć.	
Uzupełniająca	

#### Dane jako ciowe

Przyporządkowanie zajęć /grup zajęć do dyscypliny naukowej/artystycznej		informatyka techniczna i telekomunikacja	
Sposób określenia liczby punktów ECTS			
Forma nakładu pracy studenta (udział w zajęciach, aktywność, przygotowanie sprawozdania, itp.)		Obciążenie studenta [w godz.]	
Udział w zajęciach		45	
Konsultacje z prowadzącym		2	
Udział w egzaminie		0	
Bezpośredni kontakt z nauczycielem - inne		4	
Przygotowanie do laboratorium, wicze, zajęcia		24	
Przygotowanie do kolokwium i egzaminu		20	
Indywidualna praca własna studenta z literatury, wykładami itp.		20	
Inne		5	
Sumaryczne obciążenie prac studenta		120	
Liczba punktów ECTS			
Liczba punktów ECTS		4	
Zajęcia wymagające bezpośredniego udziału nauczyciela akademickiego		L. godzin	ECTS
		51	1,7

Zajęcia o charakterze praktycznym	L. godzin	ECTS
	97	3,2

1 godz = 45 minut; 1 punkt ECTS = 25-30 godzin

W sekcji 'Liczba punktów ECTS' suma punktów ECTS zajęć wymagających bezpośredniego udziału nauczyciela akademickiego i o charakterze praktycznym może się różnić od łącznej liczby punktów ECTS dla zajęć /grup zajęć.