



### IV. Opis programu studiów

#### 3. KARTA PRZEDMIOTU

Kod przedmiotu	E-TD-06-s5
Nazwa przedmiotu	Frontend development
Nazwa przedmiotu w języku angielskim	Frontend development
Obowiązuje od roku akademickiego	2020/21

#### USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

Kierunek studiów	Teleinformatyka
Poziom kształcenia	I stopień
Profil studiów	Praktyczny
Forma i tryb prowadzenia studiów	Studia stacjonarne
Zakres	
Jednostka prowadząca przedmiot	Katedra Systemów Informatycznych
Koordynator przedmiotu	Dr inż. Karol Wieczorek
Zatwierdził	Dziekan Wydziału Elektrotechniki Automatyki i Informatyki Dr hab. inż. Antoni Różowicz, prof. PŚk

#### OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów*	Przedmiot kierunkowy
Status przedmiotu*	Obieralny
Język prowadzenia zajęć	Polski
Usytuowanie modułu w planie studiów - semestr	Semestr V
Wymagania wstępne	
Egzamin (TAK/NIE)	Nie
Liczba punktów ECTS	3

\*pozostawić właściwe

Forma prowadzenia zajęć	wykład	ćwiczenia	laboratorium	projekt	Inne
Liczba godzin w semestrze	15	0	15	15	0



### EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty kształcenia	Odniesienie do efektów kierunkowych
Wiedza	W01	Wiedza z języków programowania, baz danych, inżynierii oprogramowania.	TI1_W06
	W02	Zna zagadnienia związane z tworzeniem aplikacji internetowych i mobilnych.	TI1_W09
	W03	Zna ogólne zasady i metody zarządzania projektami.	TI1_W17
Umiejętności	U01	Umiejętność tworzenia programów w językach programowania z wykorzystaniem różnych typów danych, instrukcji sterujących i funkcji oraz zdolność implementowania wybranych algorytmów.	TI1_U01 TI1_U02
	U02	Umiejętność planowania i zarządzania przykładowym projektem zgodnie z wybraną metodyką.	TI1_U01 TI1_U10 TI1_U19
	U03	Umiejętność opracowania dokumentacji i sprawozdania dotyczących rozwiązanego problemu programistycznego.	TI1_U01 TI1_U02 TI1_U20
Kompetencje społeczne	K01	Kompetencja do pracy w zespole podczas rozwiązywania problemów programistycznych i umiejętność ich podziału na mniejsze zadania.	TI1_K01 TI1_K02 TI1_K03

### TREŚCI PROGRAMOWE

Forma zajęć*	Treści programowe
wykład	1. Podstawy HTML5 (tagi HTML, tagi semantyczne, metatagi, SEO)
	2. Podstawy CSS3 (3 sposoby definiowania stylu – inline, style wewnątrz dokumentu, style zewnętrzne)
	3. Preprocesory CSS (SCSS oraz LESS)
	4. JavaScript (ES5, ES6, animacje, wykonywanie akcji przypisanych do przycisków)
	5. jQuery – jak pisać bardziej zwęży i czytelny kod
	6. AJAX – zapytania asynchroniczne w JavaScript i jQuery
	7. NodeJS oraz Webpack (optymalizacja aplikacji internetowych)
	8. Implementacja modelu MVC w najpopularniejszych frameworkach frontendowych
laboratorium	1. Szablon strony internetowej z wykorzystaniem tagów semantycznych
	2. Stylowanie strony HTML wykorzystując 3 dostępne sposoby
	3. Napisanie stylu z wykorzystaniem jednego z omawianych preprocesorów CSS oraz kompilacja za pomocą babel
	4. Implementacja wybranego algorytmu z wykorzystaniem standardu ES6
	5. Przygotowanie aplikacji pobierającej asynchronicznie dane z zewnętrznego serwera oraz wyświetlenie ich
	6-8. Implementacja aplikacji opartej o wzorzec MVC z wykorzystaniem dowolnego Frameworka JavaScript
projekt	1-8 Studenci wykonują zadanie w zespole 2-3-osobowym. Wykonanie zadania wymaga przeprowadzenia typowych czynności realizowanych w ramach projektu tworzenia frontendowego systemu informatycznego: specyfikacji wymagań, projektowania, implementacji, opracowania testów. Projekt wymaga użycia wybranego frameworka implementującego wzorzec MVC



### METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ

Symbol efektu	Metody sprawdzania efektów uczenia się					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01			X	X	X	
W02			X	X	X	
W03			X	X	X	
U01			X	X	X	
U02			X	X	X	
U03			X	X	X	
K01			X	X	X	

### FORMA I WARUNKI ZALICZENIA

Forma zajęć*	Forma zaliczenia*	Warunki zaliczenia
Wykład	Egzamin/zaliczenie z oceną/zaliczenie	Uzyskanie min. 50% punktów z testu końcowego
Laboratorium	zaliczenie z oceną/zaliczenie	Wykonanie wszystkich ćwiczeń i sprawozdań ocenionych na ocenę pozytywną
Projekt	zaliczenie z oceną/zaliczenie	Wykonanie projektu ocenionego na ocenę pozytywną

\*) zostawić tylko realizowane formy zajęć oraz wybrać formę zaliczenia

### NAKŁAD PRACY STUDENTA

Bilans punktów ECTS							
Lp.	Rodzaj aktywności	Obciążenie studenta					Jednostka
		W	C	L	P	S	
1.	Udział w zajęciach zgodnie z planem studiów	15	0	15	15	0	h
2.	Inne (konsultacje, egzamin)*	2		2	2		h
3.	<b>Razem przy bezpośrednim udziale nauczyciela akademickiego</b>	<b>51</b>					h
4.	<b>Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego</b>	<b>2,05</b>					ECTS
5.	<b>Liczba godzin samodzielnej pracy studenta</b>	<b>24</b>					h
6.	<b>Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy</b>	<b>0,95</b>					ECTS
7.	<b>Nakład pracy związany z zajęciami o charakterze praktycznym</b>	<b>30</b>					h
8.	<b>Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym</b>	<b>1,20</b>					ECTS



9.	<b>Sumaryczne godzinowe obciążenie pracą studenta</b>	<b>75</b>	h
10.	<b>Punkty ECTS za moduł</b> <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	<b>3</b>	

*\* wszelkie formy weryfikacji efektów, w tym egzaminy oraz nie więcej niż 2 godziny konsultacji dla każdej formy zajęć*

### LITERATURA

1. Jon Duckett: HTML i CSS. Zaprojektuj i zbuduj witrynę WWW. Podręcznik Front-End Developera. Helion, 2014.
2. Jon Duckett: JavaScript i jQuery. Interaktywne strony WWW dla każdego. Podręcznik Front-End Developera. Helion, 2015.
3. Jake Rutter: Podręcznik jQuery. Interaktywne interfejsy internetowe. Smashing Magazine, Helion, 2012.
4. Chris Ullman, Lucinda Dykes: Ajax. Od podstaw. Helion, 2008.
5. Mike Cantelon, Marc Harter, TJ Holowaychuk, Nathan Rajlich: Node.js w akcji. Helion, 2014.
6. Adam Freeman: Vue.js 2. Wprowadzenie dla profesjonalistów. Helion, 2019

*Uwaga: wykaz literatury winien uwzględniać aktualne i dostępne publikacje*