



IV. Opis programu studiów

3. KARTA PRZEDMIOTU

Kod przedmiotu	E-TD-10-s6
Nazwa przedmiotu	Backend development
Nazwa przedmiotu w języku angielskim	Backend 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
Koordinator 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 VI
Wymagania wstępne	
Egzamin (TAK/NIE)	Tak
Liczba punktów ECTS	6

*pozostawić właściwe

Forma prowadzenia zajęć	wykład	ćwiczenia	laboratorium	projekt	Inne
Liczba godzin w semestrze	30	0	30	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 rozwiązane go 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. Wprowadzenie do technologii backendowych; przedstawienie podstawowych pojęć
	2. Architektury aplikacji WWW,
	3. Implementacja logiki biznesowej aplikacji
	4. Technologia szablonów –implementacja warstwy prezentacji
	5. „Podział obowiązków” – wzorzec MVC
	6. IoC - Inversion of Control; DI - Dependency Injection; - zrzucanie odpowiedzialności
	7. Technika mapowania obiektowo - relacyjnego cz. I
	8. Technika mapowania obiektowo - relacyjnego cz. II
	9. Frameworki wspomagające tworzenie aplikacji w oparciu o model MVC cz.I
	10. Frameworki wspomagające tworzenie aplikacji w oparciu o model MVC cz.II
	11. Usługi sieciowe (webservices) - SOAP, REST
	12. Architektura mikroserwisów
	13. Protokoły i metody autentykacji oraz autoryzacji w systemach informatycznych
	14. Integracja z zewnętrznymi systemami informatycznymi
	15. Metody testowania kodu backendowego - BDD, TDD, Testy jednostkowe oraz integracyjne
laboratorium	1-2. Przygotowanie aplikacji implementującej MVC w oparciu o technologie serwletów
	3-4. Obsługa relacyjnych baz danych z wykorzystaniem technologii ORM
	5-6. Zastosowanie praktyczne komponentów wstrzykiwanych w ramach DI
	7. Implementacja autentykacji z wykorzystaniem filtrów



	8-9. Implementacja usług sieciowych opartych o wzorzec REST
	11-12. Implementacja usług sieciowych opartych o protokół SOAP
	13-15. Implementacja systemu opartego o architekturę mikroservisów
projekt	1-8 Studenci wykonują zadanie w zespole 3-4-osobowym. Wykonanie zadania wymaga przeprowadzenia typowych czynności realizowanych w ramach projektu tworzenia backendowego 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	
1.	Udział w zajęciach zgodnie z planem studiów	W	C	L	P	S	h	
		30	0	30	15	0		
2.	Inne (konsultacje, egzamin)*	2		2	2		h	
3.	Razem przy bezpośrednim udziale nauczyciela akademickiego	81					h	
4.	Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego	3,24					ECTS	



5.	Liczba godzin samodzielnej pracy studenta	69	h
6.	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy	2,76	ECTS
7.	Nakład pracy związany z zajęciami o charakterze praktycznym	45	h
8.	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	1,80	ECTS
9.	Sumaryczne godzinowe obciążenie pracą studenta	150	h
10.	Punkty ECTS za moduł <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	6	

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

LITERATURA

1. Bryan Basham, Kathy Sierra, Bert Bates: Head First Servlets & JSP. Edycja polska. Wydanie II. Helion, 2008
2. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides: Wzorce projektowe. Elementy oprogramowania obiektowego wielokrotnego użytku. Helion, 2010
3. Christian Bauer, Gavin King, Gary Gregory: Java Persistence. Programowanie aplikacji bazodanowych w Hibernate. Wydanie II. Helion, 2016
4. Craig Walls: Spring w akcji. Wydanie V. Helion, 2019
5. Fowler Susan: Mikrouslugi. Wdrażanie i standaryzacja systemów w organizacji inżynierskiej. Helion, 2017
6. Beck Kent: TDD. Sztuka tworzenia dobrego kodu. Helion, 2020

Uwaga: wykaz literatury winien uwzględniać aktualne i dostępne publikacje