

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Frontend Development		Kod 1010512331010510135
Kierunek studiów Informatyka	Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki	Rok / Semestr 2 / 3
Ścieżka obieralności/specjalność Gry i technologie internetowe	Przedmiot oferowany w języku: polski	Kurs (obligatoryjny/obieralny) obieralny
Stopień studiów: II stopień	Forma studiów (stacjonarna/niestacjonarna) stacjonarna	
Godziny Wykłady: 30 Ćwiczenia: - Laboratoria: 30 Projekty/seminaria: -		Liczba punktów 3
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) (ogólnouczelniany, z innego kierunku) kierunkowy z danego kierunku		
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne		Podział ECTS (liczba i %) 3 100%
Odpowiedzialny za przedmiot / wykładowca:		
<p>dr inż. Marcin Borowski email: Marcin.Borowski@put.poznan.pl tel. 61 6653032 Instytut Informatyki ul. Piotrowo 2, 60-965 Poznań</p>		
Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:		
1	Wiedza:	<p>Efekty kształcenia ze studiów I stopnia zdefiniowane w Uchwale Senatu PP, a szczególnie efekty K_W1-2, K_W4, K_W6-15, K_U1-2, K_U4, K_U7-8, K_U14-20, K_U22-23, K_U26, K_K1-9, weryfikowane w procesie rekrutacji na studia 2 stopnia ? efekty te prezentowane są w serwisie internetowym wydziału www.fc.put.poznan.pl.</p> <p>W zakresie wiedzy, student powinien posiadać podstawową wiedzę z zakresu programowania, technik oraz technologii wykorzystywanych w budowie aplikacji internetowych, sieci komputerowych oraz baz danych.</p>
2	Umiejętności:	<p>Student również powinien posiadać umiejętności rozwiązywania podstawowych problemów z zakresu algorytmiki, optymalizacji kodu programu, pracy z bibliotekami zewnętrznym, budowy prostych stron internetowych.</p>
3	Kompetencje społeczne	<p>Wskazany byłaby chęć i dążenie do poszerzania swojej wiedzy i kompetencji.</p> <p>Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.</p>
Cel przedmiotu:		
<p>1. Przekazanie studentom podstawowej wiedzy dotyczącej technologii wykorzystywanych przy budowie aplikacji webowych działających po stronie klienta, w zakresie podejść do projektowania, doboru technologii oraz implementacji (w tym również na rozwiązań przeznaczonych dla urządzeń mobilnych).</p> <p>2. Rozwijanie u studentów umiejętności rozwiązywania problemów związanych z projektowaniem aplikacji internetowych również działających w czasie rzeczywistym, wykorzystywanie frameworków, bibliotek oraz innych narzędzi wspierających budowę serwisów związanych z działalnością komercyjną w internecie.</p> <p>3. Kształtowanie u studentów umiejętności pracy zespołowej oraz samodzielności w rozwiązywaniu problemów.</p>		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza:		
<p>1. ma podbudowaną teoretycznie szczegółową wiedzę związaną z wybranymi zagadnieniami z zakresu informatyki, takimi jak: programowanie obiektowe, MVC, projektowanie aplikacji internetowych w oparciu o technologię node.js, projektowanie baz danych dla aplikacji internetowych, budowa interfejsów typu REST dla aplikacji współdziałających - [K_W5]</p> <p>2. ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach w informatyce i w wybranych pokrewnych dyscyplinach naukowych ? technologie wykorzystywane do budowy aplikacji internetowych oraz mobilnych (technologie i narzędzia oparte o node.js). - [K_W6]</p> <p>3. zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich z zakresu budowy i rozwoju aplikacji internetowych. - [K_W8]</p>		
Umiejętności:		

<p>1. potrafi pozyskiwać informacje z literatury, baz danych oraz innych źródeł (w języku ojczystym i angielskim), integrować je, dokonywać ich interpretacji i krytycznej oceny, wyciągać wnioski oraz formułować i wyczerpująco uzasadniać opinie, w szczególności korzystać różnego rodzaju dokumentacji technicznych oraz API narzędzi wykorzystywanych w trakcie laboratoriów; - [K_U1]</p> <p>2. potrafi określić kierunki dalszego uczenia się i zrealizować proces samokształcenia. - [K_U5]</p> <p>3. potrafi wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych, skupiających się wokół systemów e-commerce, metody analityczne, symulacyjne oraz eksperymentalne (takie jak: oszacowanie liczby odwiedzin serwisu, oszacowanie obciążenia serwera zapytaniami sql, ocenę optymalizacji serwisu internetowego itp.); - [K_U9]</p> <p>4. potrafi wykorzystać wspomniane metody do poprawnego zaprojektowania i zaimplementowania wydajnych mechanizmów wykorzystywanych w budowanych aplikacjach internetowych; - [K_U9]</p> <p>5. potrafi - przy formułowaniu i rozwiązywaniu zadań inżynierskich - integrować wiedzę z różnych obszarów informatyki (a w razie potrzeby także wiedzę z innych dyscyplin naukowych) oraz zastosować podejście systemowe, uwzględniające także aspekty pozatechniczne. - [K_U10]</p> <p>6. potrafi formułować i testować hipotezy związane z problemami inżynierskimi i prostymi problemami badawczymi dotyczącymi poruszanych na zajęciach zagadnień - [K_U12]</p> <p>7. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych. - [K_U13]</p>
Kompetencje społeczne:
<p>1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe. - [K_K1]</p> <p>2. potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania. - [K_K2]</p>

Sposoby sprawdzenia efektów kształcenia
<p>Ocena formująca:</p> <p>a) w zakresie wykładów:</p> <ul style="list-style-type: none">- na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach, <p>b) w zakresie laboratoriów / ćwiczeń:</p> <ul style="list-style-type: none">- na podstawie oceny bieżącego postępu realizacji zadań, <p>Ocena podsumowująca:</p> <p>a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:</p> <ul style="list-style-type: none">- ocenę wiedzy i umiejętności wykazanych w kolokwium o charakterze problemowym (student może korzystać z dowolnych materiałów dydaktycznych) / w formie testu wielokrotnego wyboru, (30 pytań za łączną liczbę 60 punktów (2 punkty za prawidłową odpowiedź), zaliczenie od 60%+1)- omówienie wyników kolokwium, <p>b) w zakresie laboratoriów / ćwiczeń weryfikowanie założonych efektów kształcenia realizowane jest przez:</p> <ul style="list-style-type: none">- ocenę i obronę przez studenta sprawozdania z realizacji projektu, <p>Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:</p> <ul style="list-style-type: none">- omówienia dodatkowych aspektów zagadnienia,- efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadane go problemu,- uwagi związane z udoskonaleniem materiałów dydaktycznych,- wskazywanie trudności percepcyjnych studentów umożliwiające bieżące doskonalenia procesu dydaktycznego.
Treści programowe
<p>Program wykładu obejmuje następujące zagadnienia:</p> <p>Różnice w protokołach komunikacji HTTP 1.0 a 1.1. Wprowadzenie do technologii node.js. Budowa prostych serwerów popularnych usług sieciowych (echo, chat, http). Wprowadzenie do frameworka Express. Wprowadzenie do frameworka MVC działającego w czasie rzeczywistym Sails.js. Omówienie narzędzi wspomagających takich jak Grunt, Bower, Less, Compass. Języki definiowania szablonów aplikacji EJS, Jade.</p> <p>Zajęcia laboratoryjne prowadzone są w formie piętnastu 2-godzinnych ćwiczeń, odbywających się w laboratorium. Ćwiczenia realizowane są samodzielnie przez studentów lub przez 2-osobowe zespoły studentów. Program laboratorium obejmuje następujące zagadnienia:</p> <p>Instalacja i konfiguracja środowiska node.js. Uruchamianie aplikacji napisanych w node.js. Proste serwer usług. Prosta aplikacja TODO zrealizowana w frameworku Express. Realizacja aplikacji w Sails.js. Korzystanie z narzędzi wspomagających oraz modułów dla node.js: Grunt, Forever, Bower itp.</p> <p>Metody dydaktyczne:</p> <ol style="list-style-type: none">1. wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy, rozwiązywanie zadań, pokaz multimedialny, demonstracja.2. ćwiczenia laboratoryjne: ćwiczenia praktyczne, dyskusja, praca w zespole, pokaz multimedialny, studium przypadków, de-monstracja.

Literatura podstawowa:		
1. jQuery Documentation - http://api.jquery.com 2. Bootstrap Framework ? http://getbootstrap.com		
Literatura uzupełniająca:		
Bilans nakładu pracy przeciętnego studenta		
Czynność		Czas (godz.)
1. udział w wykładach		30
2. udział w zajęciach laboratoryjnych / ćwiczeniach		30
3. przygotowanie do ćwiczeń laboratoryjnych		5
4. udział w konsultacjach związanych z realizacją procesu kształcenia, w szczególności ćwiczeń laboratoryjnych / projektu		2
5. napisanie programu / programów, uruchomienie i weryfikacja (czas poza zajęciami laboratoryjnymi)		5
6. przygotowanie do zaliczenia wykładów i udział w kolokwium zaliczeniowym (2 godz.)		4
7. omówienie wyników kolokwium		1
Obciążenie pracą studenta		
forma aktywności	godzin	ECTS
Łączny nakład pracy	77	3
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	65	3
Zajęcia o charakterze praktycznym	40	2