

| KARTA OPISU MODUŁU KSZTAŁCENIA | | |
|---|--|--|
| Nazwa modułu/przedmiotu Technologie programistyczne | | Kod 1010512321010510611 |
| Kierunek studiów Informatyka | Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki | Rok / Semestr 1 / 2 |
| Ścieżka obieralności/specjalność Inteligentne systemy wspomagania decyzji | Przedmiot oferowany w języku: polski | Kurs (obligatoryjny/obieralny) obligatoryjny |
| Stopień studiów: II stopień | Forma studiów (stacjonarna/niestacjonarna) stacjonarna | |
| Godziny Wykłady: 30 Ćwiczenia: - Laboratoria: - Projekty/seminaria: 45 | | Liczba punktów 5 |
| Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) kierunkowy | | (ogólnouczelniany, z innego kierunku) z danego kierunku |
| Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne | | Podział ECTS (liczba i %) 5 100% |
| Odpowiedzialny za przedmiot / wykładowca: | | |
| dr hab. inż. Szymon Wilk email: Szymon.Wilk@cs.put.poznan.pl tel. 61 6652930 Instytut Informatyki ul. Piotrowo 2, 60-965 Poznań | | dr inż. Bartłomiej Prędko email: Bartlomiej.Predki@cs.put.poznan.pl tel. 61 6652932 Instytut Informatyki ul. Piotrowo 2, 60-965 Poznań |
| Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych: | | |
| 1 | Wiedza: | 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 Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z zakresu obiektowych języków programowania, technologii mobilnych oraz internetowych. |
| 2 | Umiejętności: | Powinien posiadać umiejętność rozwiązywania podstawowych problemów związanych z projektowaniem i implementacją prostych systemów informatycznych oraz umiejętność pozyskiwania informacji ze wskazanych źródeł. |
| 3 | Kompetencje społeczne | Powinien również rozumieć konieczność poszerzania swoich kompetencji / mieć gotowość do podjęcia współpracy w ramach zespołu. Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi. |
| Cel przedmiotu: | | |
| 1. Przekazanie studentom podstawowej wiedzy z informatyki, w zakresie nowoczesnych technologii programistycznych służących do budowy systemów mobilnych i internetowych oraz w zakresie projektowania i implementacji tego typu systemów 2. Rozwijanie u studentów umiejętności rozwiązywania problemów związanych z oceną i wyborem odpowiednich technologii programistycznych dostosowanych do specyfiki, z projektowaniem i implementacją mobilnych i internetowych systemów informatycznych oraz z przygotowaniem prezentacji i sprawozdań opisujących wykonane prace 3. Kształtowanie u studentów umiejętności pracy zespołowej związanej z realizacją projektów programistycznych w grupach wieloosobowych (2-4 osoby) | | |
| Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia | | |
| Wiedza: | | |
| 1. ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie architektury systemów komputerowych, systemów operacyjnych, technologii sieciowych, języków programowania, grafiki i komunikacji człowiek-komputer, - [K_W4] 2. ma podbudowaną teoretycznie szczegółową wiedzę związaną z wybranymi zagadnieniami z zakresu informatyki, takimi jak: projektowanie aplikacji dla urządzeń mobilnych, implementacja aplikacji mobilnych w obiektowych językach programowania i nowoczesne technologie dla aplikacji internetowych, - [K_W5] 3. ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach (technologiach) w informatyce, - [K_W6] 4. ma podstawową wiedzę o cyklu życia systemów informatycznych, - [K_W7] 5. zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich z wybranego obszaru informatyki (systemy mobilne i internetowe), - [K_W8] | | |

Umiejętności:

1. potrafi pozyskiwać informacje z literatury, baz danych oraz innych źródeł (w języku ojczystym i angielskim), integrować je, dokonywać ich interpretacji i krytycznej oceny - [K_U1]
2. potrafi przygotować i przedstawić, w języku ojczystym i angielskim, prezentację ustną, dotyczącą szczegółowych zagadnień z zakresu informatyki - [K_U4]
3. potrafi formułować i testować hipotezy związane z problemami inżynierskimi i prostymi problemami badawczymi - [K_U12]
4. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych - [K_U13]
5. potrafi wybrać język programowania odpowiedni do danego zadania programistycznego - [K_U26]
6. potrafi - zgodnie z zadaną specyfikacją, uwzględniając aspekty pozatechniczne - zaprojektować złożone urządzenie, system informatyczny lub proces oraz zrealizować ten projekt ? co najmniej w części ? używając właściwych metod, technik i narzędzi, w tym przystosowując do tego celu istniejące lub opracowując nowe narzędzia - [K_U27]

Kompetencje społeczne:

1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K_K1]
2. potrafi współdziałać i pracować w grupie, przyjmując w niej różne role - [K_K5]
3. potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania - [K_K6]

Sposoby sprawdzenia efektów kształcenia

Efekty kształcenia przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

- a) w zakresie wykładów:
 - na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach,
- b) w zakresie projektów:
 - na podstawie oceny bieżącego postępu realizacji zadań,

Ocena podsumowująca:

- a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:
 - ocenę wiedzy i umiejętności wykazanych na sprawdzianie pisemnym obejmującym zarówno pytania otwarte, jak i test wielokrotnego wyboru. Sprawdzian obejmuje 10 pytań, punktacja wyrażona jest procentowo, przy czym do uzyskania oceny 3.0 konieczne jest uzyskanie 51% punktów. Podczas sprawdzianu nie można korzystać z żadnych materiałów dydaktycznych,
 - omówienie wyników sprawdzianu,
- b) w zakresie projektów weryfikowanie założonych efektów kształcenia realizowane jest przez:
 - ocenę prezentacji multimedialnych oraz sprawozdań pisemnych przedstawiających postępy w realizacji projektów. Każda grupa projektowa zobowiązana jest przygotować dwie prezentacje i sprawozdania w ciągu semestru,
 - ocenę i obronę przez grupę projektową końcowej prezentacji multimedialnej przedstawiającej zakończony projekt oraz demonstrację projektu,
 - ocenę końcowego sprawozdania pisemnego przygotowywanego częściowo w trakcie zajęć, a częściowo po ich zakończeniu; ocena ta obejmuje także umiejętność pracy w zespole,
 - ocenę systemu informatycznego zrealizowanego przez grupę projektową ze szczególnym uwzględnieniem jakości kodu źródłowego (czytelność, dokumentacja)

Treści programowe

| | | |
|--|---------------|---------------------|
| <p>Program wykładu obejmuje następujące zagadnienia:</p> <ol style="list-style-type: none"> 1. Historia rozwoju platform mobilnych 2. Wprowadzenie do języka Objective C (historia, główne cechy, podstawowe elementy programowania obiektowego) 3. Projektowanie i tworzenie aplikacji dla platformy iOS (charakterystyka platformy, cykl życia oprogramowania, narzędzia programistyczne ? Xcode, konfiguracja profilu aplikacji i przygotowanie do cyklu produkcyjnego, zgłoszenie aplikacji do AppStore) 4. Projektowanie i tworzenie aplikacji dla platformy Windows Phone (charakterystyka platformy, narzędzia programistyczne, podstawowe i zaawansowane elementy aplikacji, zgłoszenie aplikacji do Windows Phone Store) 5. Projektowanie i tworzenie aplikacji dla platformy Windows 8 (charakterystyka platformy, narzędzia programistyczne, podstawowe i zaawansowane elementy aplikacji, zgłoszenie aplikacji do Windows Store) 6. Platforma Windows Azure (charakterystyka platformy, narzędzia programistyczne i podstawy programowania, integracja z Windows Phone) 7. Platforma Microsoft SharePoint (charakterystyka platformy, narzędzia programistyczne i podstawy programowania) 8. Metody zdalnych wywołań procedur, w szczególności REST/JSON. <p>Zajęcia projektowe prowadzone są w formie piętnastu 3-godzinnych zajęć seminaryjnych odbywających się w laboratorium, podczas których studenci w grupach 2-4 osobowych realizują wybrane przez siebie projekty. Celem projektów jest zaprojektowanie i implementacja mobilnego lub internetowego systemu informatycznego przy wykorzystaniu technologii prezentowanych na wykładach. Tematy projektów związane są z charakterem specjalizacji i dotyczą szeroko pojętego podejmowania decyzji i przetwarzania danych. Program laboratorium obejmuje:</p> <ol style="list-style-type: none"> 1. zapoznanie się z propozycjami projektów i wybór projektu przez każdą z grup, 2. zapoznanie się z tematem projektu wraz z niezbędną wiedzą dziedzinową, 3. opracowanie specyfikacji wymagań dla projektu, wybór technologii oraz przygotowanie harmonogramu prac 4. przygotowanie formalnego projektu realizowanego systemu, jego implementację oraz testy 5. przygotowanie prezentacji multimedialnych oraz sprawozdań pisemnych ? cząstkowych i końcowych (łącznie 3 prezentacje i sprawozdania) 6. przedstawienie prezentacji łącznie z ?obroną? prezentacji końcowej oraz demonstracją projektu. <p>Metody dydaktyczne:</p> <ol style="list-style-type: none"> 1. wykład: prezentacja multimedialna, pokaz multimedialny, demonstracja. 2. projekt / seminarium: dyskusja, praca w zespole, prezentacja multimedialna | | |
| <p>Literatura podstawowa:</p> <ol style="list-style-type: none"> 1. Tworzenie aplikacji na platformę iOS 5 : z wykorzystaniem Xcode, Interface Builder, Instruments, GDB oraz innych kluczowych narzędzi, B. Alexander, J.B. Dillon, K.Y. Kim, Helion, 2012 2. Objective-C : praktyczny podręcznik tworzenia aplikacji na systemy iOS i Mac OS X!, S.G. Kochan, Helion 2012 3. Podstawy Windows Phone 7.5: projektowanie aplikacji przy użyciu Silverlight, S. Widemuth, Helion 2012 | | |
| <p>Literatura uzupełniająca:</p> <ol style="list-style-type: none"> 1. Windows Azure Platform, T. Redkar, T. Guidici, Apress 2011 2. Windows 8 Apps with XAML and C#, A. Nathan, SAMS, 2012 | | |
| <p>Bilans nakładu pracy przeciętnego studenta</p> | | |
| Czynność | | Czas (godz.) |
| 1. udział w zajęciach projektowych | | 45 |
| 2. przygotowanie prezentacji i sprawozdań na zajęcia projektowe (czas poza zajęciami) | | 15 |
| 3. dokończenie realizowanych projektów (czas poza zajęciami) | | 15 |
| 4. udział w konsultacjach związanych z realizacją procesu kształcenia, w szczególności projektu (częściowo mogą być realizowane drogą elektroniczną) | | 6 30 |
| 5. udział w wykładach | | 5 |
| 6. zapoznanie się ze wskazaną literaturą / dokumentacją / materiałami dydaktycznymi (10 stron tekstu naukowego = 1 godz.), 50 stron | | 8 |
| 7. przygotowanie do zaliczenia wykładów 8. udział w sprawdzenie zaliczeniowym | | 2 |
| 8. udział w sprawdzenie zaliczeniowym | | |
| <p>Obciążenie pracą studenta</p> | | |
| forma aktywności | godzin | ECTS |
| Łączny nakład pracy | 126 | 5 |
| Zajęcia wymagające bezpośredniego kontaktu z nauczycielem | 83 | 3 |
| Zajęcia o charakterze praktycznym | 75 | 3 |

