



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Programowanie terminali mobilnych (Windows i iOS) [S1Teleinf1>PTMW]

Przedmiot

Kierunek studiów
Teleinformatyka

Rok/Semestr
3/6

Studia w zakresie (specjalność)
–

Profil studiów
ogólnoakademicki

Poziom studiów
pierwszego stopnia

Język oferowanego przedmiotu
polski

Forma studiów
stacjonarne

Wymagalność
obieralny

Liczba godzin

Wykład
15

Laboratorium
30

Inne
0

Ćwiczenia
0

Projekty/seminaria
0

Liczba punktów ECTS

3,00

Koordynatorzy

dr inż. Paweł Sroka
pawel.sroka@put.poznan.pl

Wykładowcy

Wymagania wstępne

Student rozpoczynający moduł powinien posiadać podstawową znajomość zasad obiektowego tworzenia programów komputerowych, wiedzę z zakresu informatyki oraz znajomość składni języków programowania takich jak: C++, C# czy Java. Dodatkowo student powinien mieć podstawową wiedzę na temat funkcjonowania systemów operacyjnych, w szczególności z rodziny Windows i Unix, a także baz danych. Student powinien również znać podstawy architektury i sposobu działania mikroprocesorowych układów programowalnych i cyfrowych systemów telekomunikacyjnych w zakresie możliwości ich praktycznego wykorzystania. Dodatkowo student powinien umieć zaimplementować zaawansowane algorytmy z wykorzystaniem wybranych języków programowania, posiadać umiejętność wyszukiwania rozwiązań problemów programistycznych korzystając z różnych źródeł, a także być gotowym do podjęcia współpracy w ramach zespołu. Student powinien być świadomy swoich umiejętności, ograniczeń, a także konieczności ciągłego kształcenia się. Powinien też rozumieć znaczenie profesjonalnego podejścia do realizowanego zadania i odpowiedzialności za opracowane rozwiązania.

Cel przedmiotu

Celem przedmiotu jest przekazanie studentom wiedzy na temat zasad tworzenia aplikacji dla urządzeń mobilnych, a także wykształcenie umiejętności opracowania własnej autorskiej aplikacji z wykorzystaniem istniejących specyfikacji podprogramów, struktur danych i protokołów komunikacyjnych (tzw. API). Po ukończeniu przedmiotu student będzie potrafił zaprojektować i zaimplementować aplikację o określonej funkcjonalności korzystając z dostępnych zasobów i narzędzi, przeprowadzić testy aplikacji, a także przygotować ją do publikacji w sklepie internetowym. Student będzie też rozumiał dynamikę rozwoju zagadnień związanych programowaniem urządzeń mobilnych i konieczność dalszego pozyskiwania wiedzy na ten temat.

Przedmiotowe efekty uczenia się

Wiedza:

1. Ma ugruntowaną wiedzę w zakresie programowania urządzeń mobilnych.
2. Ma wiedzę dotyczącą projektowania i programowania obiektowego aplikacji dla urządzeń mobilnych, architektury i cyklu działania aplikacji.
3. Posiada wiedzę na temat sposobów użycia różnych układów i zasobów dostępnych w urządzeniach mobilnych.
3. Wie jak wykorzystać różne narzędzia oraz biblioteki programistyczne w celu poszerzenia funkcjonalności aplikacji.

Umiejętności:

1. Umie korzystać z różnych zasobów i narzędzi dostępnych w Internecie.
2. Potrafi zaprojektować i zaimplementować w pełni funkcjonalną aplikację uwzględniając specyficzne cechy systemu operacyjnego wybranego urządzenia mobilnego oraz przygotować opisującą ją dokumentację.
3. Potrafi sprawdzić poprawność implementacji aplikacji korzystając z mechanizmów testowania

Kompetencje społeczne:

1. Jest świadomy swojej wiedzy i umiejętności, a także związanych z tym ograniczeń. Rozumie konieczność dalszego kształcenia się związaną z szybkim starzeniem się wiedzy i umiejętności z zakresu tworzenia aplikacji dla iOS.
2. Jest świadomy konieczności profesjonalnego podejścia do realizowanych zadań.
3. Jest świadomy odpowiedzialności za stworzone rozwiązania i aplikacje

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wiedza nabyta w trakcie wykładów jest weryfikowana podczas egzaminu, który może mieć formę pisemną lub ustną. Egzamin pisemny składa się 6-10 pytań (testowych i/lub otwartych), które mogą być różnie punktowane. Próg zaliczeniowy dla egzaminu pisemnego to 45% możliwych do zdobycia punktów. Egzamin ustny składa się z 3 pytań związanych z tematyką wykładów, które oceniane są w skali 2-5 uwzględniając rozumienie zagadnienia przez studenta, a także szczegółowość odpowiedzi. Próg zaliczeniowy dla egzaminu ustnego to 50% pozytywnie ocenionych odpowiedzi na pytania. Umiejętności nabyte podczas laboratorium są weryfikowane podczas realizacji 3-7 ćwiczeń praktycznych polegających na implementacji aplikacji o określonej funkcjonalności, zgodnie z przygotowaną instrukcją do zadania. Każde z ćwiczeń jest oceniane w postaci punktów na podstawie przygotowanego sprawozdania, przy czym liczba punktów odpowiadająca zadaniu jest zależna od jego stopnia skomplikowania. Wpływ na ocenę końcową ma też ocena pracy i zaangażowania studenta na zajęciach. Ocena końcowa zależna jest od liczby zdobytych punktów, przy czym próg zaliczeniowy wynosi 50% maksymalnej liczby punktów za wszystkie ćwiczenia.

Treści programowe

W ramach przedmiotu omawiane są następujące tematy

- Wprowadzenie do programowania urządzeń z iOS.
- Tworzenie prostych aplikacji i podstawy budowania interfejsu użytkownika.
- Tworzenie adaptacyjnego i responsywnego interfejsu użytkownika. Obsługa gestów.
- Trwały zapis danych w urządzeniach z iOS.
- Komunikacja sieciowa urządzeń iOS.

- Inne wybrane funkcjonalności aplikacji dla iOS.

Tematyka zajęć

Na wykładach omawiane są następujące zagadnienia:

- Wprowadzenie do programowania urządzeń mobilnych - omówienie wybranych języków i zasad programowania aplikacji mobilnych.
- Architektury programistyczne aplikacji mobilnych i związana z nimi struktura projektu aplikacji.
- Tworzenie interfejsu użytkownika aplikacji mobilnej. Obsługa gestów.
- Trwały zapis danych w aplikacjach dla urządzeń mobilnych.
- Użycie powiadomień.
- Testowanie aplikacji mobilnych.
- Wybrane funkcjonalności aplikacji, takie jak: lokalizacja urządzeń, wykorzystanie map, łączność sieciowa.
- Publikowanie aplikacji.

W ramach laboratorium realizowane są następujące tematy:

- Podstawy tworzenia aplikacji dla urządzeń mobilnych. Implementacja podstawowych elementów graficznych.
- Implementacja aplikacji wielostronicowej (dynamiczna zmiana zawartości na ekranie).
- Tworzenie adaptacyjnego interfejsu użytkownika i obsługa gestów.
- Trwały zapis danych aplikacji.
- Testowanie aplikacji mobilnej.
- Wykorzystanie wybranych bibliotek i funkcjonalności dostępnych w API do realizacji zaawansowanych funkcjonalności.

Na wykładach omawiane są następujące zagadnienia:

- Wprowadzenie do programowania urządzeń iOS i języka programowania Swift.
- Rodzaje, rola i działanie kontrolerów widoków w aplikacji iOS.
- Tworzenie interfejsu użytkownika aplikacji iOS. Dynamiczna adaptacja interfejsu aplikacji oraz obsługa gestów.
- Trwały zapis danych w urządzeniach iOS.
- Powiadomienia i ich rola w aplikacji iOS.
- Testowanie aplikacji dla iOS.
- Dodatkowe funkcje aplikacji iOS: pobieranie informacji o lokalizacji, użycie map, usługi sieciowe.
- Publikowanie aplikacji w sklepie internetowym.

Tematyka laboratorium przedstawia się następująco:

- Tworzenie prostej aplikacji z podstawowym interfejsem użytkownika.
- Wprowadzenie do implementacji elementów graficznych.
- Implementacja aplikacji wielostronicowej.
- Trwały zapis danych.
- Implementacja adaptacyjnego interfejsu użytkownika oraz obsługa gestów.
- Testowanie aplikacji iOS: testy jednostkowe i integracji.
- Wykorzystanie map oraz informacji o lokalizacji w aplikacji.
- Tworzenie i obsługa powiadomień w iOS.

Metody dydaktyczne

Wykłady: prezentacja multimedialna wspierana prezentacją wybranych zagadnień w postaci aplikacji uruchamianej na komputerze lub udostępnionego filmu.

Laboratorium: ćwiczenia laboratoryjne - implementacja aplikacji o określonej funkcjonalności w sposób opisany w instrukcji z wykorzystaniem komputerów zawierających odpowiednie oprogramowanie i symulator wybranych urządzeń mobilnych.

Literatura

Podstawowa:

Mark A. Lassoﬀ, Tom Stachowitz (tł. Robert Górczyński), "Podstawy języka Swift : programowanie aplikacji dla platformy iOS", Helion, 2016.

Matt Neuburg (tł. Robert Górczyński), "iOS 12 : wprowadzenie do programowania w Swiftie", Helion, 2019.

Uzupełniająca:

N.Smyth, "iOS 10 App Development Essentials", CreateSpace Independent Publishing Platform, 2016
<https://developer.apple.com>
<https://www.appcoda.com/>
<https://www.raywenderlich.com/>
Godzin ECTS
Łączny nakład pracy 86 3.0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem 45 2.0
Praca własna studenta (przygotowanie do zaliczenia, przygotowanie do 41 1.0
laboratorium, studia literaturowe)

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	86	3,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	45	2,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwίων/egzaminu, wykonanie projektu)	41	1,00