



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Programowanie terminali mobilnych [S2EiT1E-TIT>PTM]

Przedmiot

Kierunek studiów

Elektronika i telekomunikacja/Electronics and Telecommunications

Rok/Semestr

1/2

Studia w zakresie (specjalność)

Technologie informacyjno-telekomunikacyjne

Profil studiów

ogólnoakademicki

Poziom studiów

drugiego stopnia

Język oferowanego przedmiotu

angielski

Forma studiów

stacjonarne

Wymagalność

obieralny

Liczba godzin

Wykład

30

Laboratorium

15

Inne

0

Ćwiczenia

15

Projekty/seminaria

0

Liczba punktów ECTS

4,00

Koordynatorzy

dr inż. Marcin Rodziewicz

marcin.rodziewicz@put.poznan.pl

Wykładowcy

Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z podstaw programowania, architektury systemów komputerowych i systemów operacyjnych, a także z zakresu współczesnych systemów radiokomunikacji ruchomej. Powinien również posiadać umiejętność programowego zrealizowania podstawowych algorytmów obliczeniowych oraz umiejętność pozyskiwania informacji ze wskazanych źródeł. Student powinien także mieć gotowość do podjęcia współpracy w ramach zespołu

Cel przedmiotu

Celem przedmiotu jest zapoznanie studentów z zasadami programowania terminali mobilnych oraz wykształcenie umiejętności przygotowania własnej, kompletnej aplikacji gotowej do umieszczenia w dedykowanych serwisach internetowych (sklepach aplikacji dla terminali mobilnych). Szczególny nacisk jest położony na programowanie terminali wykorzystujących system operacyjny Android.

Przedmiotowe efekty uczenia się

Wiedza:

1. Posiada ugruntowaną wiedzę z zakresu programowania terminali mobilnych.

2. Posiada wiedzę na temat możliwości wykorzystania różnorodnych modułów i zasobów dostępnych we współczesnych terminalach mobilnych.

Umiejętności:

1. Posiada umiejętność korzystania z różnorodnych zasobów dostępnych w Internecie (zazwyczaj w języku angielskim).
2. Potrafi przygotować kompletny wniosek wraz z wymaganą dokumentacją.

Kompetencje społeczne:

1. Jest świadomy/a ograniczeń swojej wiedzy i umiejętności; rozumie potrzebę dalszego kształcenia.
2. Jest świadomy/a potrzeby profesjonalnego podejścia do rozwiązywania problemów i brania odpowiedzialności za proponowane rozwiązania.
3. Jest świadomy/a swojej odpowiedzialności za rozwijane systemy.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wiedza nabyta w ramach wykładu jest weryfikowana przez kolokwium lub zaliczenie ustne realizowane na ostatnim wykładzie.

Kolokwium składa się z pytań testowych i otwartych, różnie punktowanych. Próg zaliczeniowy: 50% punktów, co oznacza opanowanie wiedzy co najmniej w stopniu dostatecznym.

Zaliczenie ustne składa się z odpowiedzi na 3-5 pytań. Każda odpowiedź na zadane pytanie oceniana jest w skali od 2 do 5. Ocena końcowa z zaliczenia ustnego stanowi średnią z ocen za poszczególne odpowiedzi. Zaliczenie jest zdane, gdy średnia ocena jest wyższa niż 3.

Umiejętności nabyte w ramach zajęć laboratoryjnych weryfikowane są podstawie zadań przydzielanych podczas zajęć lub projekt. Każde zadanie oceniana jest w skali od 2 do 5. Ocena końcowa zajęć laboratoryjnych stanowi średnią z ocen za poszczególne zadania lub ocena za projekt.

Treści programowe

1. Środowisko programistyczne
2. Język Kotlin
3. Tworzenie interfejsu użytkownika
4. Podstawowe komponenty aplikacji
5. Zapis danych w aplikacjach
6. Operacje sieciowe
7. Popularne biblioteki wykorzystywane w programowaniu urządzeń z systemem Android

Tematyka zajęć

Program wykładu obejmuje:

1. Omówienie środowiska programistycznego dla platformy Android.
2. Wprowadzenie do języka Kotlin
3. Omówienie struktury projektu aplikacji i wykorzystywanych zasobów.
4. Omówienie zasad tworzenia wyglądu aplikacji i związanych z tym różnych układów graficznych i podstawowych elementów wyglądu aplikacji.
5. Omówienie koncepcji "aktywności" oraz jej cyklu życia.
9. Omówienie koncepcji intencji oraz nawigacji z wykorzystaniem grafu nawigacji.
10. Omówienie manifestu aplikacji.
11. Omówienie zasad pracy z wątkami i procesami.
12. Omówienie sposobów zapisu danych m.in. biblioteka Room
13. Omówienie sposobów komunikacji z wykorzystaniem HTTP
14. Omówienie wybranych dodatkowych narzędzi przydatnych przy tworzeniu aplikacji: Usługi Google, Firebase,

Program laboratorium obejmuje:

1. Zapoznanie się ze środowiskiem programistycznym Android Studio
2. Poznanie podstaw tworzenia wyglądu aplikacji

3. Poznanie zasad tworzenia komponentów aplikacji i wykorzystanie bazy danych
4. Poznanie bibliotek umożliwiających komunikację z REST API

Metody dydaktyczne

1. Wykład: Prezentacja multimedialna i dyskusja
2. Laboratoria: Wykonywanie zadań podanych w instrukcjach wydanych przez opiekuna
3. Samouczki: Prezentacja z przykładami kodu i aplikacji

Literatura

<https://developer.android.com>

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	100	4,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	70	3,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	30	1,00