



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Programowanie terminali mobilnych (Android) [S1Teleinf1>PTMA]

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. 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 uporządkowaną wiedzę w zakresie tworzenia aplikacji przeznaczonych na terminale mobilne.

2. Posiada wiedzę w zakresie możliwości wykorzystania zasobów dostępnych w terminalach mobilnych w rozwiązywaniu rzeczywistych problemów inżynierskich.

Umiejętności

1. Potrafi korzystać z bogatych zasobów dostępnych w Internecie (W tym w języku angielskim) w celu stworzenia aplikacji.

2. Potrafi przygotować pełną aplikację wraz z wymaganą dokumentacją (również w języku angielskim).

Kompetencje społeczne

1. Zna ograniczenia własnej wiedzy i umiejętności, rozumie konieczność dalszego doskonalenia się.

2. Posiada świadomość konieczności profesjonalnego podejścia do rozwiązywanych problemów technicznych i podejmowania odpowiedzialności za proponowane przez siebie rozwiązania techniczne.

3. Ma poczucie odpowiedzialności za zaprojektowane systemy i zdaje sobie sprawę z zagrożeń dla ludzi i dla społeczeństwa w wypadku ich nieodpowiedniego zaprojektowania lub wykonania

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. Każde zadanie oceniana jest w skali od 2 do 5. Ocena końcowa zajęć laboratoryjnych stanowi średnią z ocen za poszczególne zadania.

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.
6. Omówienie interakcji aplikacji z użytkownikiem
7. Omówienie koncepcji "fragmentu" i metod zarządzania "fragmentami".
8. Omówienie zasad budowania i debuggowania aplikacji
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 (aktywności, fragmenty, okna dialogowe, listy)
4. Poznanie bibliotek umożliwiających komunikację z REST API
5. Zapoznanie się z usługami Google dostępnymi na platformę Android.

Metody dydaktyczne

1. Wykład: Instruktaż wraz z prezentacją multimedialną
2. Ćwiczenia laboratoryjne: Wykonywanie zadań z instrukcji dostarczanych przez prowadzącego

Literatura

<https://developer.android.com>

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 kolokwium/egzaminu, wykonanie projektu)	41	1,00