



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Programowanie terminali mobilnych

Przedmiot

Kierunek studiów

Elektronika i Telekomunikacja

Studia w zakresie (specjalność)

Technologie Mobilne i Bezprzewodowe

Poziom studiów

pierwszego stopnia

Forma studiów

stacjonarne

Rok/semestr

III/VI

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

Polski

Wymagalność

obieralny

Liczba godzin

Wykład

15

Ćwiczenia

Laboratoria

15

Projekty/seminaria

Inne (np. online)

Liczba punktów ECTS

3

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

dr inż. Marcin Rodziewicz

Odpowiedzialny za przedmiot/wykładowca:

marcin.rodziewicz@put.poznan.pl

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 (głównie w języku angielskim) w celu utworzenia 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 kształcenia 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 (elektroniczne i telekomunikacyjne) 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.

Kolokwiów składa się z 10 pytań (testowych i otwartych), różnie punktowanych. Próg zaliczeniowy: 50% punktów, co oznacza opanowanie wiedzy co najmniej w stopniu dostatecznym.

Przyjęta jest skala ocen:

pkt.	ocena
<= 50%	2.0
51%-60%	3.0
61%-70%	3.5
71%-80%	4.0
81%-90%	4.5
91%-100%	5.0

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.

Zagadnienia zaliczeniowe, na podstawie których opracowywane są pytania zostaną przekazane studentom na stronie internetowej prowadzącego.



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

Wykład:

1. Omówienie środowiska programowego dla platformy Android.
2. Omówienie struktury projektu i wykorzystywanych zasobów.
3. Omówienie zasad tworzenia wyglądu aplikacji i związanych z tym różnych układów graficznych i podstawowych elementów wyglądu aplikacji.
4. Omówienie koncepcji "aktywności" oraz jej cyklu życia.
5. Omówienie koncepcji "fragmentu" i metod zarządzania "fragmentami".
6. Omówienie zasad budowania i debuggowania aplikacji na emulatorze i na urządzeniu, m.in. z wykorzystaniem narzędzia adb.
7. Omówienie koncepcji "Intentu".
8. Omówienie komponentów aplikacji: "Service", "Broadcast".
9. Omówienie manifestu aplikacji.
10. Omówienie zasad pracy z wątkami i procesami.
11. Przedstawienie możliwości wykorzystania różnych rodzajów menu oraz powiadomień.
12. Omówienie interakcji aplikacji z użytkownikiem - zasady wykorzystania klas "Listeners" i "Handlers".
13. Omówienie sposobów dostępu do pamięci urządzenia.
14. Omówienie dostępnych typów czujników i możliwości ich wykorzystania.
15. Omówienie wybranych dodatkowych narzędzi przydatnych przy tworzeniu aplikacji: Usługi Google, Firebase, Volley.

Laboratoria:

1. Zapoznanie się ze środowiskiem programistycznym AndroidStudio
2. Poznanie podstaw tworzenia wyglądu aplikacji
3. Poznanie zasad tworzenia komponentów aplikacji (aktywności, fragmenty, okna dialogowe, listy)
4. Zapoznanie się ze sposobami wykorzystania możliwości sprzętowych terminali mobilnych.



5. Zapoznanie się z usługami Google dostępnymi na platformę Android.

Metody dydaktyczne

1. Wykład: prezentacja multimedialna
2. Ćwiczenia laboratoryjne: Wykonywanie zadań z instrukcji dostarczanych przez prowadzącego

Literatura

Podstawowa

<https://developer.android.com>

Uzupełniająca

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
łączy nakład pracy	75	3,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	31	1,5
Praca własna studenta (przygotowanie do zajęć laboratoryjnych, przygotowanie do kolokwium, realizacja zadań domowych) ¹	44	1,5

¹ niepotrzebne skreślić lub dopisać inne czynności