



## COURSE DESCRIPTION CARD - SYLLABUS

Course name

PO 6.2.1 Programowanie terminali mobilnych (Android) - EC 6.2.1 Programming of Mobile Terminals in Android

### Course

Field of study

Teleinformatics

Year/Semester

3/6

Area of study (specialization)

Profile of study

general academic

Level of study

first-cycle studies

Course offered in

Polish

Form of study

full-time

Requirements

elective

### Number of hours

Lecture

15

Laboratory classes

30

Other (e.g. online)

Tutorials

0

Projects/seminars

0/0

### Number of credit points

3

### Lecturers

Responsible for the course/lecturer:

Responsible for the course/lecturer:

dr inż. Marcin Rodziewicz  
Institute of Radiocommunications  
marcin.rodziewicz@put.poznan.pl  
tel. +48 61 665 3915

### Prerequisites



The student should possess a basic knowledge on programming, architecture of computer systems and operating systems. The student should be able to implement basic algorithms and be able to find necessary information from different resources. The student should also be able to work in a team.

### Course objective

The main goal of the course is to develop student's skills in programming of mobile terminals. After completing the course students will be able to implement their own application, ready for release in Internet markets. Particular attention will be put on the devices using Android system.

### Course-related learning outcomes

#### Knowledge

1. Possesses the grounded knowledge in the area of programming of mobile terminals.
2. Possesses the knowledge about the possibilities of usage of various modules and resources available in nowadays mobile terminals.

#### Skills

1. Possesses the skills of using various resources available in Internet.
2. Is able to prepare the complete application together with the required documentation.

#### Social competences

1. Is aware of his/her knowledge and skills limitations; understand the need of further study.
2. Is aware of the need for professional treatment of the problems to be solved and for taking responsibility for proposed solutions.
3. Is aware of his/her responsibilities for the developed systems.

### Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Theoretic knowledge (based on the lectures) will be checked during a test or oral verification.

Laboratories will be credited based on the tasks assigned during laboratories and homework assignments or project

### Programme content

Topics related to programming applications for the operating system. The basic elements of the application will be discussed, ie Activities, Fragments, Listeners and topics related to the creation of the user interface. In addition, issues related to the use of devices in the form of Internet communication, sensors and information about geographic location will be presented. During the course, we will also get acquainted with the Android Studio environment.

Lecture scope:



1. Android and Android Studio - overview of the basics of Android and the Android Studio development environment
2. Kotlin language - overview of Kotlin language
3. Basics of application development - overview of the basics of creating applications for Android. Discussion of the basic components of the application (in particular the concept of Activity), ways of creating the user interface, application resources
4. User interface and navigation in the application - Overview of the basic elements for creating a user interface and the handling of interaction with them. Introducing the concept of the Fragments. Overview of navigation in applications both with the use of Fragments (navigation component) and Activities (concept of intention).
5. Application manifest, Application components, application memory - Discuss the meaning and content of the application manifest. Discussion of advanced application components in the form of services and broadcast messages. Overview of data saving methods in Android applications.
6. Background Tasks, Network Operations, Device Capabilities, Google Play Services - Discussing the different ways to perform tasks in separate threads. Overview of selected libraries for communication with remote services via the http protocol. Overview of the use of device sensors and obtaining information about its location. Overview of Google Play services on the example of Google Maps and the Firebase platform.

#### Laboratory scope:

1. Basics of application development - creating the first simple application and learning about the concept of Activity and its life cycle. Get to know Android Studio, Android project structure and Kotlin language
2. User interface - - Create and modify application user interface. Adding new views to an existing activity and interacting with them. Views. Using Intentions to move between activities.
3. Fragments, Navigation Component, RecyclerView - creating an application using fragments and a list view (RecyclerView). Using the navigation component to define transitions between different parts in the application. Create and interact with simple dialog boxes.
4. Network operations, databases - Creation of an application that allows displaying data from an external API using the Retrofit library. The user can save selected data in a local database operated by the Room library.
5. Google Play, Firebase, Sensors services - Creation of an application demonstrating the use of Google Play services in the application in the form of Firebase authentication modules and real-time databases.

### Teaching methods

1. Lecture: Multimedia presentation
2. Laboratories: Performing tasks given in instructions provided by the supervisor

### Bibliography

#### Basic

1. Materials from the lecturer
2. <https://developer.android.com>



Additional

1. <https://cloud.google.com/>
2. <https://firebase.google.com>

### Breakdown of average student's workload

	Hours	ECTS
Total workload	86	3.0
Classes requiring direct contact with the teacher	45	2.0
Student's own work (preparation for tests, preparation for laboratory classes, literature studies)	41	1.0