

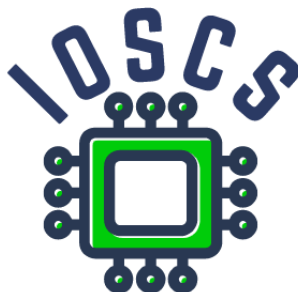
**Project: Innovative Open Source Courses for Computer Science**

# **Mobile Application Development Syllabus**

**dr inż. Radosław Maciaszczyk**  
**West Pomeranian University of Technology in Szczecin**

**29. 1. 2020**

## Innovative Open Source Courses for Computer Science



This syllabus was written as one of the outputs of the project “Innovative Open Source Courses for Computer Science”, funded by the Erasmus+ grant no. 2019-1-PL01-KA203-065564. The project is coordinated by West Pomeranian University of Technology in Szczecin (Poland) and is implemented in partnership with Mendel University in Brno (Czech Republic) and University of Žilina (Slovak Republic). The project implementation timeline is September 2019 to December 2022.

### Project information

Project was implemented under the Erasmus+.

Project name: **“Innovative Open Source courses for Computer Science curriculum”**

Project nr: **2019-1-PL01-KA203-065564**

Key Action: **KA2 – Cooperation for innovation and the exchange of good practices**

Action Type: **KA203 – Strategic Partnerships for higher education**

#### Consortium

ZACHODNIOPOMORSKI UNIWERSYTET TECHNOLOGICZNY W SZCZECINIE

MENDELOVA UNIVERZITA V BRNE

ZILINSKA UNIVERZITA V ZILINE

#### Erasmus+ Disclaimer

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

#### Copyright Notice

This content was created by the IOSCS consortium: 2019–2022. The content is Copyrighted and distributed under Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) free for Non-Commercial use.



Co-funded by the  
Erasmus+ Programme  
of the European Union

## COURSE DESCRIPTION

Field of study: **Computer Science**

Level: **First cycle**

Course name: **Mobile Application Development**

ECTS credits: **4**

Instruction forms: **Lecture, project, laboratory**

Instruction hours: **15, 15,15**

Type, extent and method of teaching activities: **1 - 1 - 1 (lectures-exercises-labs) hours weekly or 2-2-2 hours every 2 weeks; class attendance required.**

Prerequisites: **Knowledge of at least one object programming language, Preferred Java language.**

Module/course unit objective: **The main objective of the course is to introduction to programming mobile devices. After the course students will know how to create applications for mobile devices with Android OS.**

Course content is divided into various forms of instruction (with a number of hours):

<i>Course content - Lectures</i>		<i>Hours</i>
<i>W-1</i>	Introducing mobile device and mobile systems. Development tools	2
<i>W-2</i>	Application Fundamentals, components - activities, services, broadcast receivers, content providers	2
<i>W-3</i>	Component lifecycle - activity, fragments, services	2
<i>W-4</i>	User Interface, Introduction to Material Design, typography, main component	2
<i>W-5</i>	Sensors, GNNS, - use case of sensors, type of sensors, sensors lifecycle	2
<i>W-6</i>	Data persistence - Room Database, App preferences	2
<i>W-7</i>	Design pattern MVVM	2
<i>W-8</i>	Networking, HTTP connections	1
	<b>Laboratory</b>	
<i>L-1</i>	Configure the Development Environment; create the first program. Debug programs	2
<i>L-2</i>	Create a user interface. Introduction to widgets	2
<i>L-3</i>	Activities and Intents	2
<i>L-4</i>	Database and <i>RecyclerView</i> to display data	2
<i>L-5</i>	Locations, permissions	2
<i>L-6</i>	Sensors	2
<i>L-7</i>	MVVM, LiveData	1
<i>L-8</i>	Networking	2
	<b>Project</b>	
<i>P-1</i>	Introduction to project, project's functions	2
<i>P-2</i>	Working with own Project	10
<i>P-3</i>	Documentation	2
<i>P-4</i>	Presentation project	1

**Student workload – forms of activity:** individual work with computer in Android Studio, realise short applications, working with own project.

**Teaching methods/tools:** Lectures- informative, problem solving, conversational, laboratories and project, computer laboratory with installed Android Studio SDK and IDE, and internet connection.

**Evaluation methods:** Evaluation is based on two components –continuous assessment during the semester and final exam. They are appreciated as follows.

**Continuous examination:**

- Semester - 80 points:
  - project completeness - max. 50 points, min 25 points,
  - laboratory tasks - max. 30 points, min 15 points,
- Exam - 20 points: theoretical questions/tasks - min. 10 points

Final examination: Successful completion presumes obtaining at least 61 points, including at least 10 points for theoretical problems. Evaluation of the subject:

- A 93 – 100,
- B 85 – 92,
- C 77 – 84,
- D 69 – 76,
- E 61 – 68.

The student must have at least 30.0 points to enrol for an exam.

Planned learning outcomes: After completing the course, the student:

- implement well-structured Android applications,
- use of sensors and databases in Android applications,
- integrate internet services in the application,
- use the "Material Design" pattern to build the user interface.

**Bibliography:**

1. Ian F. Darwin, Android Cookbook, Problems and Solutions for Android Developers, O'Reilly, 2012
2. Zigurd Mednieks, Laird Dornin, G. Blake Meike, Masumi Nakamura, Programming Android, 2nd Edition-Java Programming for the New Generation of Mobile Devices, O'Reilly, 2012
3. Mark L. Murphy, The Busy Coder's Guide to Android Development, CommonsWare – Digital version - <http://commonsware.com/Android/>,
4. <http://d.android.com>, 2019
5. Android Developer Fundamentals, <https://developer.android.com/courses/fundamentals-training/overview-v2> 2019
6. Material Design [material.io](http://material.io)