

Application development for Android platform - course description

General information	
Course name	Application development for Android platform
Course ID	11.3-WE-INF-D-ADfAP-Er
Faculty	Faculty of Computer Science, Electrical Engineering and Automatics
Field of study	Computer Science
Education profile	academic
Level of studies	Second-cycle Erasmus programme
Beginning semester	winter term 2021/2022

Course information	
Semester	3
ECTS credits to win	5
Course type	optional
Teaching language	english
Author of syllabus	<ul style="list-style-type: none">dr inż. Piotr Powroźnik

Classes forms					
The class form	Hours per semester (full-time)	Hours per week (full-time)	Hours per semester (part-time)	Hours per week (part-time)	Form of assignment
Lecture	15	1	-	-	Credit with grade
Laboratory	15	1	-	-	Credit with grade
Project	15	1	-	-	Credit with grade

Aim of the course

To provide knowledge about advanced programming technics for mobile devices working on Android platform.

To extend skills in programming of touch interfaces.

To provide knowledge about limitation in programming of mobile devices.

To give skills in designing, testing and publishing portable mobile application.

Prerequisites

Fundamentals of Android system programming, Mobile device programming

Scope

Designing mobile applications for Android OS. Determining the ability to satisfy requirements of the application. Prepare test plans for quality control. Selection of source code management system. Using the numbering system version. Designing for expansion and patching applications. Designing for application interoperability. Testing of mobile applications for the Android platform. Designing a system for recording errors for programming mobile devices.

Methods of application publishing. Preparation of code to create an installation package. Generating and signing an application package. Testing the published version of the application package. Support for end-user applications. Tracking and verification of faults. Testing firmware update target devices.

Teaching methods

Lecture: conventional lecture, discussion, consultation.

Laboratory: laboratory exercises, discussion, consultation.

Project: project exercises, group work, discussion, consultation.

Learning outcomes and methods of theirs verification

Outcome description	Outcome symbols	Methods of verification	The class form
Is able to design a mobile application for Android		<ul style="list-style-type: none">a projectan evaluation testan observation and evaluation of the student's practical skillsan ongoing monitoring during classes	<ul style="list-style-type: none">LectureLaboratoryProject
Can work individually and in a team		<ul style="list-style-type: none">an observation and evaluation of the student's practical skills	<ul style="list-style-type: none">LaboratoryProject

Outcome description	Outcome symbols	Methods of verification	The class form
Can use a version control system for tracking changes in source code		<ul style="list-style-type: none"> • a project • an evaluation test • an observation and evaluation of the student's practical skills • an ongoing monitoring during classes 	<ul style="list-style-type: none"> • Lecture • Laboratory • Project
Is able to publish mobile apps for Android		<ul style="list-style-type: none"> • a project • an evaluation test • an observation and evaluation of the student's practical skills • an ongoing monitoring during classes 	<ul style="list-style-type: none"> • Lecture • Laboratory • Project
Is able to create and test mobile apps for Android		<ul style="list-style-type: none"> • a project • an observation and evaluation of the student's practical skills • an ongoing monitoring during classes 	<ul style="list-style-type: none"> • Lecture • Laboratory • Project

Assignment conditions

Lecture – the passing condition is to obtain a positive mark from the final test.

Laboratory – the passing condition is to obtain positive marks from all laboratory exercises to be planned during the semester.

Project – the passing condition is to obtain positive marks for all project tasks as scheduled.

Calculation of the final Grade: lecture 40% + laboratory 30% +project 30%

Recommended reading

1. Phillips B., Stewart C., Marsicano K.: Android Programming: The Big Nerd Ranch Guide (3rd Edition) (Big Nerd Ranch Guides) 3rd Edition, Big Nerd Ranch Guides; 2017
2. Griffiths D., Griffiths D.: Head First Android Development: A Brain-Friendly Guide 2nd Edition, O'Reilly Media; 2 edition, 2017
3. MacLean D., Komatineni S., Allen G.: Pro Android 5 5th ed. Edition, Apress; 5th ed. edition, 2015
4. Yener M., Dundar O.: Expert Android Studio 1st Edition, Wrox; 1 edition, 2016
5. Hogbin Westby E. M.: Git for Teams: A User-Centered Approach to Creating Efficient Workflows in Git 1st Edition, O'Reilly Media; 1 edition, 2015
6. Laster B.: Professional Git 1st Edition, Wrox; 1 edition, 2016
7. Chacon S., Straub B.: Pro Git 2nd ed. Edition, Apress; 2nd ed. edition, 2014

Further reading

1. Gerber A., Craig C.: Android Studio. Wygodne i efektywne tworzenie aplikacji. Helion, Gliwice, 2016
2. DiMarzio J. F.: Tworzenie gier na platformę Android 4. Helion, Gliwice, 2013
3. Guihot H.: Optymalizacja wydajności aplikacji na Android, Helion, Gliwice, 2013
4. Gerber A., Craig C.: Android Studio. Wygodne i efektywne tworzenie aplikacji, Helion, Gliwice 2016
5. Taskos G.: Xamarin. Tworzenie aplikacji cross-platform. Receptury, Helion, Gliwice, 2017
6. Eckel B.: Thinking in Java. Edycja polska. Wydanie IV, Helion, Gliwice, 2006
7. Gajda W.: Git: Rozproszony system kontroli wersji, Helion, Gliwice, 2013
8. Silverman R. E.: Git: Leksykon kieszonkowy, Helion, Gliwice, 2014

Notes

Modified by dr inż. Piotr Powroźnik (last modification: 14-07-2021 13:10)

Generated automatically from SylabUZ computer system