

Faculty of Engineering and Technology

Computer Science Department

**Comp438 Mobile Software Development**

1. **Course information:**
   1. **Course Code:** COMP 438.
   2. **Course Name:** Mobile App Development.
   3. **Prerequisite:** Comp333
2. **Instructors:**

**Name: Dr. Samer Zein**

**Email:** [**szain@birzeit.edu**](mailto:szain@birzeit.edu)

**Room: Masri417  
Name: Mr. Ahmad Sabah  
Email:** [**asabah@birzeit.edu**](mailto:asabah@birzeit.edu) **Room: Masri 418**

1. **Course Description**

A comprehensive hands-on programming course targeting latest technologies and frameworks in Android native mobile app development. Concise explanations and hand-on case studies will guide students through the different frameworks, tools, and patterns for developing scalable, compelling, and robust mobile apps that can reach customers and clients on a variety of devices. The course focuses on development for Android platform as well as providing platform agnostic knowledge that can help students to understand how other platforms work too. Taught topics include: UX best practices, Android main building components (Activities, Intents, Fluid UI, shared preferences), version control using Github, life cycle conformance, consuming Rest Web Services using Volley and GLIDE, building backend web REST services using Python and Flask, developing in restricted environments, asynchronous programming, Location Services and Maps, Android Jetpack ROOM, GSON, Cloud data storage using Firebase, Real-time databases, and unit tests.

1. **Course Goals**
   1. Understand the peculiarities of mobile computing in general and native mobile development in particular
   2. Best UX practices for mobile apps.
   3. Understand the main building blocks of Android platform development.
   4. Creating robust mobile apps with respect to lifecycle states.
   5. Understand how to consume scalable RESTful Web Services.
   6. Understand fluid interfaces development strategies.
   7. Understand and apply asynchronous programming.
   8. Understand version control frameworks.
   9. Understand how to build REST services using Python Flask
   10. Understand cloud data persistence and Real-time databases.
   11. illustrate test driven development and test automation.
2. **Course Outcomes**

Upon successful completion of this course, the student will be able to:

* 1. **Knowledge and understanding**
     1. Comprehend the building blocks of native mobile applications using Android platform.
     2. Understand asynchronous programming and fluid interfaces.
     3. Understand cloud data storage and real-time databases.
     4. Apply test driven development and test automation.
  2. **Cognitive skills (thinking and analysis).**

1. Establishing a mobile strategy to understand how mobile apps are different than traditional web and desktop applications
2. Analyse and investigate different development frameworks for mobile applications.
3. Build mobile application for the enterprise.
4. Develop back-end Real-time databases.
   1. **Communication skills (personal and academic**).

. Be able to communicate with other member of the testing team in order to evaluate and develop mobile apps.

* 1. **Practical skills (Transferable Skills)**

. Use available Android integrated development environments such as Android Studio and Firebase toolkit.

1. **Course Content**

|  |  |
| --- | --- |
| **# of Lectures** | **Topics** |
| 3 | Native apps and their peculiarities, Android SDK Features, Android Structure |
| 2 | Introduction to Activities |
| 3 | More about Activities and Intents. |
| 3 | Data persistence using shared preferences |
| 2 | User interfaces and layouts, Asynchronous programming |
| 3 | Connecting with Restful Web services with Volley and GLID |
| 2 | Building and consuming REST Services using Python Flask |
| 2 | Firebase, Cloud Data storage, and Realtime databases |
| 2 | Data persistence using Jetpack ROOM |
| 2 | Location based-services |
| 1 | Android Maps |
| 1 | best UX design practices |
| 2 | Test driven development and test automation |
|  |  |

1. **Teaching and learning methods**
   1. **Lectures**
   2. **Assignments**
   3. **Presentations and Exams.**
2. **assessment methods based on outcomes**
3. ......Exams ......*To Assess*........ knowledge and understanding.......
4. ....Practical projects .........*To Assess*........practical skills & cognitive skills.......
5. **Weighting of assessments**

|  |  |
| --- | --- |
| **Project** | **20%** |
| **Assignments & quizzez** | **20%** |
| **Midterm** | **25%** |
| **Final Exam** | **35%** |
| **Total** | **100%** |

1. **References**
   1. Essential books /text books
      1. Beginning Android Programming with Android Studio (2016)
      2. Headfirst Android Development, second Edition.
      3. Professional Android, Wrox, 2018. 11 – Additional Readings

[www.developer.android.com](http://www.developer.android.com/)

12- Important course Regulations:

A) You will have one week to review your assignment/exam mark. After that, we cannot review your mark.

B) Work submission on Ritaj is only accepted if it is as a reply to same assignment message.

C) Late submission will not be accepted.

D) Code Submission is only accepted as a link to Github Repository.