

### Module Description, available in: EN

# **Mobile Computing**

#### **General Information**

**Number of ECTS Credits** 

3

Module code

TSM\_MobCom

Valid for academic year

2023-24

Last modification

2019-10-10

Coordinator of the module

Thomas Amberg (FHNW, thomas.amberg@fhnw.ch)

Explanations regarding the language definitions for each location:

- Instruction is given in the language defined below for each location/each time the module is held.
- Documentation is available in the languages defined below. Where documents are in several languages, the percentage distribution is shown (100% = all the documentation).
- The examination is available 100% in the languages shown for each location/each time it is held.

	Lausanne			Lugano	Zurich		
Instruction					<b>X</b> E 100%		
Documentation					<b>X</b> E 100%		
Examination					<b>X</b> E 100%		

### **Module Category**

TSM Technical scientific module

## Lessons

2 lecture periods and 1 tutorial period per week

### **Entry level competences**

Prerequisites, previous knowledge

Students have working knowledge in:

- Programming in Java
- Software development and frameworks
- Desktop or mobile user interface development
- Internet protocols, HTTP and sending Web requests
- Students bring a laptop to class.
- Students bring an Android phone to class (if available).

### Brief course description of module objectives and content

This module enables students to develop advanced, native applications for the Android mobile operating system and get a solid understanding of mobile computing concepts. Building on the basics of mobile application development, this course covers a selection of application programming interfaces for on-board sensors and connectivity options for the integration with backend services, IoT platforms and peripheral devices. Lecturers share their experience and best-practices from recent projects involving mobile computing. Students work with both emulators and real devices.

### Aims, content, methods

#### Learning objectives and competencies to be acquired

Application Development (50%)

- Students know how to design and implement native applications for mobile devices running Android, the most widely used mobile platform.
- Students have basic knowledge of user-interface design guidelines and techniques relevant for mobile application design.
- Students can describe the integration of their application with a cloud backend.

Sensors and Connectivity (50%)

- Students know how to use on-device sensor APIs for motion, position and environment.
- Students have basic knowledge of connectivity options like Near Field Communication (NFC), Bluetooth Low Energy (BLE) and Wi-Fi.
- Students can describe the integration of their application with a peripheral device or IoT platform using request/response or messaging protocols.
- Students have basic knowledge of prototyping a peripheral IoT device with an Arduino-compatible hardware platform, sensors and actuators.

#### Module content with weighting of different components

Application Development

- Development of native mobile applications for Android, including user interfaces
- Specific aspects in mobile application programming such as application lifecycle, data storage, data synchronization with a cloud backend, and security of mobile applications.

Sensors and Connectivity

- Development with on-device sensor APIs for motion, position and environment.
- Specific aspects of connecting to peripheral devices with connectivity options like NFC, BLE and Wi-Fi, prototyping an IoT device and integration with IoT platforms.

### Teaching and learning methods

- Ex-cathedra teaching
- Mini-projects
- Exercises
- Self-Study

# Literature

### Android

- IDE https://developer.android.com/studio
- Docs <a href="https://developer.android.com/docs">https://developer.android.com/docs</a>
- Source Code https://source.android.com/

### Arduino

IDE https://www.arduino.cc/en/Main/Software#download

### **Assessment**

### **Certification requirements**

Module uses certification requirements

# Certification requirements for final examinations (conditions for attestation)

Mini-projects count 30% for final grade.

### Basic principle for exams

As a rule, all standard final exams are conducted in written form. For resit exams, lecturers will communicate the exam format (written/oral) together with the exam schedule.

# Standard final exam for a module and written resit exam

Kind of exam

Written exam

**Duration of exam** 

120 minutes

Permissible aids

Aids permitted as specified below:

Permissible electronic aids

No electronic aids permitted

# Other permissible aids

- Closed book examination
- Personal 1-page summary allowed

# Special case: Resit exam as oral exam

Kind of exam

Oral exam

**Duration of exam** 

30 minutes

Permissible aids

No aids permitted