Syllabus

Course Description

Course Title	Mobile and Physical Systems
Course Code	76262
Course Title Additional	
Scientific-Disciplinary Sector	INFO-01/A
Language	English; German
Degree Course	Bachelor in Computer Science
Other Degree Courses (Loaned)	
Lecturers	Prof. Dr. Michael Haller, Michael.Haller@unibz.it https://www.unibz.it/en/faculties/engineering/academic-staff/person/11217 dr. Niccolò Pretto, Niccolo.Pretto@unibz.it https://www.unibz.it/en/faculties/engineering/academic-staff/person/47860
Teaching Assistant	
Semester	First semester
Course Year/s	3
CP	12
Teaching Hours	60
Lab Hours	60
Individual Study Hours	180
Planned Office Hours	
Contents Summary	• Functional and declarative programming • Design of mobile applications • Frameworks and platforms for mobile development • Data and resource management in a mobile context • Mobile device sensors • Internet of Things

	• Motivation & Short overview about prototyping (using ProtoPie) • Sensor Fundamentals (using Arduino/ESP32) • 3D Environment Development (using Unity) • Design & sketch a next-generation controller • Presentation of the prototype • Evaluation of the prototype
Course Topics	In Engineering of Mobile Systems, students will learn the key concepts of mobile application development and the internet of things. Practical experience will be gained by using state of the art technologies for the development of mobile applications. Upon completion of the course, students shall have acquired expertise in writing mobile applications that leverage advanced mobile APIs and connect to outside web services, and shall be aware of the various trade-offs in the development of mobile applications. In Physical Computing Project, by building an idea, designers are challenged to "build to think" and thus gain deeper insights. This course will go beyond early physical prototyping and show how to implement smart sensing devices that can be used to control an interactive environment (e.g., game). Participants will learn basic electronics, microcontroller programming, and physical prototyping using the Arduino/ESP32 platform, then use digital and analog sensors which results in a next-generation controller, e.g., in combination with Unity. Therefore, students will gain a profound understanding of sensor technologies as well as a broad overview of how to design and implement a 3d environment.
Keywords	Physical Prototyping, Hardware, Electronics, Ubiquitous Computing, Front-End Development, Mobile Computing
Recommended Prerequisites	The course requires knowledge from the "Software Engineering" course.
Propaedeutic Courses	
Teaching Format	The course includes frontal lectures, lab exercises, and projects.
Mandatory Attendance	Attendance is not compulsory, but highly recommended as many labs require an adequate software and hardware infrastructure; non-attending students may contact the lecturer at the start of the course to get support on the modalities of the independent study.
Specific Educational	Knowledge and Understanding

Objectives and Learning Outcomes	– D1.15 Possess knowledge and methodologies of software design and development in the mobile environment – D1.17 Know the main methods for the design of interactive smart objects for IoT Applying knowledge and understanding – D2.2 Be able to develop small and medium size programs using different programming languages and paradigms. – D2.3 Be able to solve problems using programming methodologies. – D2.14 Be able to develop Mobile applications. – D2.19 Be able to apply the own knowledge in different working contexts. – D2.20 Be able to select and apply innovative technologies and methods that are appropriate for a given context and problem. – D2.23 Be able to coordinate small project teams and to interact with members of the group. – D2.25 Be able to apply interactive design principles and patterns for IoT solutions and smart objects. Ability to make judgments – D3.1 Be able to collect and interpret useful data and to judge information systems and their applicability. – D3.2 Be able to work autonomously according to the own level of knowledge and understanding. – D3.3 Be able to take the responsibility for development of projects or IT consulting. Communication skills – D4.1 Be able to use one of the three languages English, Italian and German, and be able to use technical terms and communication appropriately. – D4.2 Be able to use modern communication systems, even at a distance. – D4.4 Be able to structure and write technical documentation. – D4.5 Be able to work in teams for the realization of IT systems. Learning skills – D5.1 Have developed learning capabilities to pursue further studies with a high degree of autonomy. – D5.2 Have acquired learning capabilities that enable to carry out project activities in companies, public institutions or in distributed development communities.

	– D5.3 Be able to follow the fast technological evolution and to learn cutting edge IT technologies and innovative aspects of last generation information systems.
Specific Educational Objectives and Learning Outcomes (additional info.)	-
Assessment	The assessment is based on a final exam, which covers the topics addressed in both modules and consists of two parts. Part 1 covers the module Engineering of Mobile Systems (50% of the final exam) and is based on an oral exam and a project work in which students will design and develop a mobile application for addressing a practical problem. The project will be submitted in a git repository provided by the lecturer, described in a report, and presented during the oral exam. In case of a positive mark, the project will count for all 3 regular exam sessions. Projects must be submitted at least a week before the exam, otherwise the student cannot access to the oral exam. Part 2 (50% of the final exam) covers the module Physical Computing Project. The major activity of the class is centered around a group project (in pairs of two), but there will be individual assignments early in the semester. The goal of these assignments is to ensure everyone in the class gains experience and understanding of the design and implementation of a novel input controller, without which creating an interesting and sophisticated project (next-generation controller) will be difficult.
Evaluation Criteria	The exam is evaluated based on correctness of projects, answers, clarity of answers, ability to summarize, evaluate, and establish relationships between topics, skills in critical thinking, quality of argumentation, problem solving ability. To pass the exam, the students must get at least 18/30 in each module. The mark related to each part contributes to the final grade as follows: Part 1 covers the module Engineering of Mobile Systems (50% of the final exam). Of this part, 70% of the mark is determined by the project work and 30% by the oral exam. Students need to pass project work to access the oral exam. The project work is assessed evaluating if it contains working solutions and if it adheres to good programming practices and styles. The oral exam is evaluated

	based on the clarity of answers and the project presentation, being able to master the terminology of the course, and being able to solve problems related to mobile applications or summarize theoretical concepts. Part 2 (50% of the final exam) covers the module Physical Computing Project. Each student group is provided with a physical computing kit including an Arduino/ESP32 compatible board as well as everything needed to learn how to use sensors and actuators and how to combine it with 3rd party tools (e.g. Unity). Through hands-on experiences during class periods, students acquire basic skills and learn to build a range of typical circuits that will communicate to Unity. Along with basic skill acquisition, students are involved in a semester-long group assignment in which they develop a complex project from start to finish. Students are encouraged to quickly arrive at a working prototype at which point they can fine-tune their project through testing. To evaluate this part, at the end of the semester, the projects are presented to the rest of the class.
Required Readings	Lecture slides and lab materials provided by instructors
Supplementary Readings	– Dawn Griffiths and David Griffiths. Head First Android Development: A Learner's Guide to Building Android Apps with Kotlin. O'Reilly Media, Sebastopol, 3rd edition edition, December 2021. ISBN 978-1-4920-7652-0. – Bryan Sills, Brian Gardner, Kristin Marsicano, and Chris Stewart. Android Programming: The Big Nerd Ranch Guide. Addison-Wesley Professional, Atlanta, GA, 5th edition edition, July 2022. ISBN 978-0-13-764554-1. Peter Marwedel. Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems, and the Internet of Things. Springer, 4th edition edition, January 2021.
Further Information	Software used: – Android Studio (https://developer.android.com) – ProtoPie (https://www.protopie.io) – Unity (https://unity.com) – Arduino IDE (https://www.arduino.cc/en/software) – Different microcontrollers and microelectronics kits are used. Only participant students, who attend classes, can use them during

	class time. Further information is provided on the course page.
Sustainable Development Goals (SDGs)	Industry, innovation and infrastructure

Course Module

Course Constituent Title	Engineering of Mobile Systems
Course Code	76262A
Scientific-Disciplinary Sector	INFO-01/A
Language	English
Lecturers	dr. Niccolò Pretto, Niccolo.Pretto@unibz.it https://www.unibz.it/en/faculties/engineering/academic-staff/person/47860
Teaching Assistant	
Semester	First semester
CP	6
Responsible Lecturer	
Teaching Hours	30
Lab Hours	30
Individual Study Hours	90
Planned Office Hours	
Contents Summary	– Functional and declarative programming – Design of mobile applications – Frameworks and platforms for mobile development – Data and resource management in a mobile context – Mobile device sensors – Internet of Things
Course Topics	In Engineering of Mobile Systems, students will learn the key concepts of mobile application development and the internet of things. Practical experience will be gained by using state of the art technologies for the development of mobile applications. Upon completion of the course, students shall have acquired expertise in writing mobile applications that leverage advanced mobile APIs and connect to outside web services, and shall be aware of the various trade-offs in the development of mobile applications.

Teaching Format	The course includes frontal lectures, lab exercises, and projects.
Required Readings	Lecture slides and lab materials provided by instructors
Supplementary Readings	– Dawn Griffiths and David Griffiths. Head First Android Development: A Learner's Guide to Building Android Apps with Kotlin. O'Reilly Media, Sebastopol, 3rd edition edition, December 2021. ISBN 978-1-4920-7652-0. – Bryan Sills, Brian Gardner, Kristin Marsicano, and Chris Stewart. Android Programming: The Big Nerd Ranch Guide. Addison-Wesley Professional, Atlanta, GA, 5th edition edition, July 2022. ISBN 978-0-13-764554-1.

Course Module

Course Constituent Title	Prototyping Physical Interactive Experiences
Course Code	76262B
Scientific-Disciplinary Sector	INFO-01/A
Language	German
Lecturers	Prof. Dr. Michael Haller, Michael.Haller@unibz.it https://www.unibz.it/en/faculties/engineering/academic-staff/person/11217
Teaching Assistant	
Semester	First semester
CP	6
Responsible Lecturer	
Teaching Hours	30
Lab Hours	30
Individual Study Hours	90
Planned Office Hours	
Contents Summary	– Motivation & Short overview about prototyping (using ProtoPie) – Sensor Fundamentals (using Arduino/ESP32) – 3D Environment Development (using Unity) – Design & sketch a next-generation controller – Presentation of the prototype

	– Evaluation of the prototype
Course Topics	In Physical Computing Project, by building an idea, designers are challenged to "build to think" and thus gain deeper insights. This course will go beyond early physical prototyping and show how to implement smart sensing devices that can be used to control an interactive environment (e.g., game). Participants will learn basic electronics, microcontroller programming, and physical prototyping using the Arduino/ESP32 platform, then use digital and analog sensors which results in a next-generation controller, e.g., in combination with Unity. Therefore, students will gain a profound understanding of sensor technologies as well as a broad overview of how to design and implement a 3d environment.
Teaching Format	Physical Prototyping, Hardware, Electronics, Ubiquitous Computing, Front-End Development
Required Readings	Lecture slides and lab materials provided by instructors
Supplementary Readings	Peter Marwedel. Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems, and the Internet of Things. Springer, 4th edition edition, January 2021.