SYLLABUS

Android Things

University year 2025-2026

1. **Information regarding the programme**

1.1. Higher education institution	Babeș-Bolyai University
1.2. Faculty	Faculty of Mathematics and Computer Science
1.3. Department	Department of Computer Science
1.4. Field of study	Computer Science
1.5. Study cycle	Bachelor
1.6. Study programme/Qualification	Computer Science
1.7. Form of education	Full time

2. **Information regarding the discipline**

2.1. Name of the discipline	**Android Things**					Discipline code	**MLE5117**
2.2. Course coordinator			Dan Cojocar, PhD				
2.2. Seminar coordinator			Dan Cojocar, PhD				
2.4. Year of study	**3**	2.5. Semester	**6**	2.6. Type of evaluation	**C**	2.7. Discipline regime	**Optional**

3. **Total estimated time** (hours/semester of didactic activities)

3.1. Hours per week	**5**	Of which: 3.2 course	**2**	3.3. Seminar/laboratory/project	**1 lab +2 pr**
3.4. Total hours in the curriculum	**60**	Of which: 3.5 course	**24**	3.6. Seminar/laboratory/project	**36**
Time allotment for individual study (ID) and self-study activities(SA)					**hours**
Learning using manual, course support, bibliography, course notes (SA)					10
Additional documentation (in libraries, on electronic platforms, field documentation)					20
Preparation for seminars/labs, homework, papers, portfolios, and essays					20
Tutorship					10
Evaluations					5
Other activities					0
3.7. Total individual study hours				**65**	
3.8. Total hours per semester				**125**	
3.9. Number of ECTS credits				**5**	

4. **Prerequisites** (if necessary)

4.1. Curriculum	Mobile Applications
4.2. Competencies	Average Java/Kotlin programming skills

5. **Conditions** (if necessary)

4.1. For the course	Projector
4.2. For the seminar/lab activities	• Internet access and ability to use personal laptops • Interactive whiteboard. • IoT development boards.

6.1. **Specific competencies acquired**[1]

Professional/ essential competencies	• C1.3. Elaboration of adequate source codes and unitary testing of some components in a known programming language, based on given design specifications. • C1.5. Development of program units and elaboration of the corresponding documentation. • C2.4. Collaborating through digital technologies. • C2.5. Development of specific software systems. • C3.4. Programming. • C5.1. Appropriate use of the operating principles of electronic devices and circuits and methods of measuring electrical quantities. • C6.3. Techniques for installation, configuration, and administration of systems and computer networks.

[1]One can choose either competences or learning outcomes, or both. If only one option is chosen, the row related to the other option will be deleted, and the kept one will be numbered 6.

Transversal Competencies	• CT1. Application of efficient and organized work rules, of responsible attitudes towards the didactic-scientific domain, to creatively value one's own potential, with respect towards the principles and norms of professional ethics. • CT3. Use of efficient methods and techniques to learn, inform, research, and develop the abilities to value the knowledge, to adapt to the requirements of a dynamic society, and to communicate in the Romanian language and in a language of international circulation.

6.2. **Learning outcomes**

Knowledge	The graduate can design/develop/debug basic IoT applications.
Skills	The graduate can apply architectural styles, design patterns, and best practices in the field to design IoT applications.
Responsibility and autonomy	• The graduate is familiar with the tools used for testing, debugging, and validating IoT applications. • The graduate is familiar with project management tools, version control systems, and continuous integration/continuous delivery (CI/CD) concepts, methods, tools.

7. **Objectives of the discipline** (outcome of the acquired competencies)

7.1. **General objective of the discipline**	• Be able to use the IoT Developer Platforms. • Improved development skills. • Average IoT programming abilities.
7.2. **Specific objective of the discipline**	• To understand the key concepts of IoT. • Develop software using the Android Things Developer Platform. • Develop applications using the Android Things Developer Kit.

8. **Content**

8.1. **Course**	Teaching methods	Remarks
1. Introduction to IoT/Android Things	Exposure: description, examples, discussion of case studies, live demo.	
2. Basic Electronics		
3. Arduino		
4. Protocol&Interfaces		
5. Raspberry Pi		
6. User-Space Drivers		
7. AI in IoT		
8. Cloud IoT		
9. CoaP		
10. MQTT		
11. Thread		
12. Wrap-up	Discussion of case studies, exam discussions.	

Bibliography

- Android Things website: https://developer.android.com/things/index.html
- Android Things reference: https://developer.android.com/things/reference/index.html
- Francesco Azzola - Android Things Projects: Efficiently build IoT projects with Android Things, Packt Publishing, 2017

8.2. Seminar/Laboratory	Teaching methods	Remarks
1. Hand out developer kits. • Create a project plan. • Discuss the development kit features. • IoT Laboratory Platform Component Presentation.	Exposure: description, discussion. Evaluation.	The lab is structured as 2 hours classes every second week.
2. Present the current ideas to the first-course students.		

• Build the teams. • Discuss the ideas.		
3. Discuss/Evaluate progress.		
4. Discuss/Evaluate progress.		
5. Discuss/Evaluate progress.		
6. Discuss/Evaluate progress.		
7. Paper/Project Demos/Presentations		

Bibliography

- Android Things website: https://developer.android.com/things/index.html
- Android Things reference: https://developer.android.com/things/reference/index.html
- Francesco Azzola - Android Things Projects: Efficiently build IoT projects with Android Things, Packt Publishing, 2017

9. **Corroborating the content of the discipline with the expectations of the epistemic community, professional associations, and representative employers within the field of the program**

- The course aligns with the IEEE and ACM curriculum recommendations for Computer Science programs.
- It is included in the study programs of major universities both in Romania and internationally.
- The course content is considered essential by software companies for developing solid, industry-relevant programming skills.

10. **Evaluation**

Activity type	10.1. Evaluation criteria	10.2. Evaluation methods	10.3. Percentage of final grade
10.4 Course	• The basic principle of the domain. • Apply the course concepts. • Problem-solving.	Individual project grading.	60%
10.5 Seminar/ laboratory	• Be able to implement course concepts and algorithms. • Apply techniques for different classes of programming languages.	Team project grading.	40%
10.6. Minimum standard of performance			
• Attend 90% of lab activities during the semester. • At least grade 5 (1 to 10 scale) in all activities, seminar/lab, and written exam. • The final grade must be at least 5.			

11. **Labels ODD (Sustainable Development Goals)**[2]

Not applicable.

Date: Signature of course coordinator, Dan Cojocar, PhD

Signature of seminar coordinator, Dan Cojocar, PhD

Date of approval: Signature of the head of department, Adrian Sterca, PhD

[2]Keep only the labels that, according to the Procedure for applying ODD labels in the academic process, suit the discipline and delete the others, including the general one for Sustainable Development – if not applicable. If no label describes the discipline, delete them all and write "Not applicable.".