Subject programme

1. Subject name / subject module: **Mobile Devices Programming**
2. Lecture language: **English**
3. The location of the subject in study plans:
    - Area or areas of the studies: **Computer Control Systems Engineering**
    - Degree of the studies: **2nd degree studies**
    - Field or fields (implementation of effects standard): **Mechatronics**
4. Supervision of subject implementation:
    - The Institute / Another unit: **The Institute of Informatics and Mechatronics**
    - The person responsible for the subject: **Skiba Grzegorz, mgr inż.**
    - People cooperating in the development of the programme of the subject:
5. The number of hours and forms of teaching for individual study system and the evaluation method

	Teaching activities with the tutor																						
																							Total
Form of classes / Mode of study		SOW	ECTS	Laboratory work	SOW	ECTS	...	SOW	ECTS	...	SOW	ECTS	...	SOW	ECTS	...	SOW	ECTS	...	SOW	ECTS		ECTS
Full-time studies				47	53	4																	4
Part-time studies																							
Credit rigor	...			Graded assigment																			

6. Student workload – ECTS credits balance

1 ECTS credit corresponds to 25-30 hours of student work needed to achieve the expected learning outcomes including the student's own work

Activity (please specify relevant work for the subject)	**Hourly student workload (full-time studies/part-time studies)**
Participation in laboratory classes	**47**
Preparation of final project	**30**
Independent study of the subject	**21**
Participation in an exam / graded assignment / final grading	**2**
Total student workload	**100**
ECTS credits	**4**
* Student's workload related to practical forms	**100**
Student's workload in classes requiring direct participation of academic teachers	**47**

7. Implementation notes: recommended duration (semesters), recommended admission requirements, relations between the forms of classes:

Basics of programming

Recommended duration of the subject is taken from the course plan.

8. Specific learning outcomes – knowledge, skills and social competence

Specific learning outcomes for the subject		Form	Teaching method	Methods for testing of (checking, assessing) learning outcomes
Outcome symbol	Outcome description			
Knowledge				
K_W07	**Student knows how to apply in practice the programming skills in Java for Android systems to be used with appropriate programming environment and methods to solve problems that may occur in real-life mechatronics environment.**	**Laboratory work**	**Inquiry methods**	**Student learning activities, evaluation of the implementation of a programming tasks**
Skills				
K_U02	**Student is able to use and create documentation related to the Android Studio programming environment as well as software designed and created by the studentt for implementation in projects and tasks related to mechatronics.**	**Laboratory work**	**Inquiry methods**	**Student learning activities, evaluation of the implementation of a programming tasks**
Social competence				
K_K01	**Student is ready to gather information and improve workshop related to mobile devices programming for Android platform,**	**Laboratory work**	**Inquiry methods**	**Student learning activities, evaluation of the implementation of a programming tasks**

	especially those related to third-party libraries, components and devices.			

9. Assessment rules / criteria for each form of education and individual grades

0% - 50%	ndst	**81% - 90%**	db
51% - 70%	dst	**91% - 93%**	db+
71% - 80%	dst+	**94% - 100%**	bdb

Activity	**Grades**	**Calculation**	**To Final**
Tasks done during laboratories	**bdb (5)**	**5*20%**	**1**
Attendance	**on 70% of all classes**	**0,70*5 -> 3,5*10%**	**0,35**
Final project	**dst, db, bdb (3,4,5)**	**3/4/5 * 70%**	**3,5**
Final result			**4,85**

10. The learning contents with the form of the class activities on which they are carried out

(Laboratory work)

1. Using Android UI objects: Buttons; EditTexts; TextViews; Layouts; Views; Events;

2. Communication inside Android application: Saving and reading data; Shared Preferences, Intents;

3. Creating synchronous and asynchrous methods in Android;

4. Using HTTP protocol for communication with remote Application Programming Interface (API): RESTful Web services; JSON data format; GET and POST methods;

5. Long-running background operations: Service; AlarmManager;

6. Google Maps SDK for Android: Getting API Key; Configuration; Using markers with popups;

7. Configuring and developing notifications under certain conditions;

11. Required teaching aids

Laboratory classes - specialist laboratory

12. Literature:

a. Basic literature:

Hagos T.: Learn Android Studio 3, Springer, Apress, Berkeley, 2018.

Wickham M.: Practical Android, Apress, Berkeley, 2018.

Smith D., Hellman E.: Android Recipes, Apress, Berkeley, 2016.

a. Supplementary literature:

Jackson W.: Android Apps for Absolute Beginners, Springer, Lompoc, 2017.

Friesen J.: Learn Java for Android Development, Apress, Berkeley, 2014.

b. Internet sources:

https://loopj.com/android-async-http/, Documentation for asynchronous HTTP client in Android

https://appinventor.mit.edu/, Tool for fast prototyping mobile applications

https://developer.android.com/docs, Android documentation

13. Available educational materials divided into forms of class activities (Author's compilation of didactic materials, e-learning materials, etc.)

14. Teachers implementing particular forms of education

Form of education	Name and surname
1. Laboratory classes	Skiba Grzegorz, mgr inż.