Subject programme

1. Subject name / subject module: **Elective Subject: 3D Cross-platform applications**
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: **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:

Mode of study	Teaching activities with the tutor												Total
	Form of classes												
	...	SOW	ECTS	Laboratory work	SOW	ECTS	...	SOW	ECTS	...	SOW	ECTS	ECTS
Full-time studies				45	55	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 lectures	-
Participation in laboratory classes	45
Preparation to laboratory classes	30
Independent study of the subject	13
Preparation to a final test	10
Participation in an exam / graded assignment	2
Total student workload (TSW)	100
ECTS credits	4
* Student's workload related to practical forms	100
Student's workload in classes requiring direct participation of academic teachers	45

7. Implementation notes: recommended duration (semesters), recommended admission requirements, relations between the forms of classes:
   - Recommended admission requirements – none.
   - 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, at an advanced level, the Unity environment, programming and designing 3D applications for different platforms, such as Windows, Android, iOS, macOS, and is able to use this knowledge to apply it in mechatronics industry.**	**Laboratory work**	**Inquiry methods**	**Final test, Student learning activities**
Skills				
K_U02	**Student is able to use and create documentation related to the programming environment as well as different platforms for use with Unity 3D environment for implementation in projects and tasks related to mechatronics.**	**Laboratory work**	**Inquiry methods**	**Final test, Student learning activities**
Social competence				
K_K02	**Student is ready to recognize and solve problems related to developing 3D cross-platform applications and knows where to find information related to particular platform or programming environment in case he needs assistance.**	**Laboratory work**	**Inquiry methods**	**Final test, Student learning activities**

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

Activity	Grades	Calculation	To final
Final test/project	bdb (5)	5*30%	1,5
Attendance	on 70% of all classes	0,70*5 -> 3,5*20%	0,7
Tasks done during laboratories	dst, db, bdb (3,4,5)	arithmetic mean (3,4,5) * 50%	2,5
Final result			4,70

0 – 3.00	ndst	**4.01 – 4.50**	db
3.01 – 3.50	dst	**4.51 – 4.7**	db+
3.51 – 4.00	dst+	**4.71 – 5.0**	bdb

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

a. Laboratory work:

1. Overview of Unity interface;
2. Creating and importing assets to Unity Project: Simple object modelling, Using Asset Store, Importing assets;
3. In-game world design, Moving and rotating elements, Applying textures to models, Scenes, Cameras and characters;
4. Designing multi-platform applications: Overview of available systems in Unity, 2D and 3D applications, Developing scripts in C#, Applying scripts to objects;
5. Targeting and testing of applications.

11. Required teaching aids:

a. Laboratory classes - specialist laboratory.

12. Literature:

a. Basic literature:

1. Fowler A., Chu P.: Learn Unity 2017 for iOS Game Development, Apress, Berkeley, 2017.
2. Blackman S.: Unity for Absolute Beginners, Apress, Berkeley, 2014.
3. Blackman S.: Beginning 3D Game Development with Unity 4, Apress, Berkeley, 2013.

b. Supplementary literature:

1. Sharp J.: Microsoft Visual C#. Step by step., Microsoft Press, Redmond, 2015.
2. Sinicki A.: Learn Unity for Android Game Development, Apress, Berkeley, 2017.
3. Thorn A.: Learn Unity for 2D Game Development, Apress, Berkeley, 2013.

b. Internet sources:

1. https://unity.com/learn, Unity learning library
2. https://docs.unity3d.com/Manual/index.html, Unity user manual and documentation
3. https://brackeys.com/, Game development tutorials database

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:

Subject programme

Form of education	Name and surname
1. Lecture	
2. Laboratory classes	Skiba Grzegorz, mgr inż.
3. Training	
4. Project classes	
5. Workshop classes	
6. Simulation game	
7. Language classes	