MEĐIMURSKO VELEUČILIŠTE U ČAKOVCU
MEĐIMURJE UNIVERSITY OF APPLIED SCIENCES IN ČAKOVEC

COURSE SYLLABUS

ACADEMIC YEAR:	2025/2026

1. GENERAL COURSE INFORMATION

1.1 Course name	Development of Computer Games			
1.2 Study program/s	UNDERGRADUATE PROFESSIONAL STUDY PROGRAMME COMPUTER ENGINEERING			
1.3 Course status (O,E)	E	1.6 Mode of instruction (number of hours)	Lectures	30
1.4 Course code			Exercises	30
1.5 Course abbreviation			Seminars	
1.6 Semester	VI		E-learning	Merlin
1.7 ECTS	4	1.7 Place and time of instruction	The premises of the Međimurje University of Applied Sciences in Čakovec, according to the schedule published on the website	

2. TEACHING STAF.F

2.1 Course leader/s-title	Nenad Breslauer, senior lecturer	contact	nbreslauer1@mev.hr
		contact	
2.2 Assistant/s- title		contact	
		contact	
2.3 Instruction held by-title	Nenad Breslauer, senior lecturer	contact	nbreslauer1@mev.hr
2.4 Course leader/s-title		contact	

3. COURSE DESCRIPTION

3.1. Course goals	After completing the course, the student will learn to use a platform for developing computer games, during which they will acquire the knowledge needed to develop simple 3D and 2D computer games. Students will master the use of the game development platform by integrating concepts related to using pre-made 2D and 3D content with knowledge of software development. The acquired skills and knowledge of game development will be further enhanced with knowledge that enables the creation of virtual and augmented reality systems. Special attention will be given to creating educational 3D games within virtual and augmented reality, as well as designing user interfaces and interactions within them. Students will learn to use modern game development platforms (Unity game engine, C# programming language) and equipment for virtual and augmented reality systems.
3.2 Prerequisites	No conditions.
3.3 Course outcomes	After successfully completing the course, students will be able to: I1 - Explain what a game development platform is and what its basic advantages are. I2 - Build an environment where interactions among objects (participants) take place. I3 - Create mechanisms for simulating physical laws. I4 - Design and develop scripts within the game development platform. I5 - Assemble a simple virtual and/or augmented reality system.

3.4 Contribution of the course to the study program	Apply communication and professional ethics Identify trends in ICT technologies in the domestic and international markets Analyze user needs (research and detect data sources, currently present business systems, technological limitations, specific business environment characteristics) Develop program code in multiple programming languages using modern methods and tools Apply relevant mathematical and statistical methods in software engineering Identify the basic specifics of operating systems Differentiate types and communication protocols of computer networks Apply database basics through creating, modelling, and administering databases Choose data structuring methods in program code, as well as techniques for writing complex program forms, and utilize standard algorithms Develop applications using the object-oriented paradigm in solving programming tasks Analyze basic elements of electrical engineering and digital circuits and identify computer architecture Use Cloud computing as a concept for accessing data and applications Develop web and mobile projects, applying advanced technologies, and connecting with databases using modern methods and tools Choose the appropriate programming language and technology in solving programming tasks
3.5 Course content	Thematic units will cover various areas of game development and creation, including creating and using graphic elements and implementing their behaviors.

3.6 Types of coursework

x	Lectures	X	Exercises		Blended e-learning	X	Individual activities		Laboratory
x	Seminars and workshops		Distant learning		Field classes		Multimedia and network		Mentorship
	Other								

3.7 Language of instruction	Croatian

3.8 Monitoring students' work (enter the number of ECTS credits for each activity so that the total number of ECTS credits is equal to the total ECTS value of the course, 1 ECTS = 30 hours)

	Class attendance			Seminars			Research	
	Midterm exams/written exam	2		Project			Essay	
	Oral exam	2		Practical task				

Activity specification	Percent %	Points
Assessment during instruction		
Class Assignments	30%	30
Seminar paper/projec	70%	70
Exam assessment for the students who failed to fullfil all the obligatory requirements during the semester		
Written Exam	70%	70
Oral Exam	30%	30
Total:	**100%**	**100**

Points   Grade
89 – 100 excellent (5)
76 – 88   very good (4)
63 – 75   good (3)
50 – 62   pass (2)
* 0 – 49   fail (1)*

3.9 Specific features related to taking the course	*If a student collects 50% of the points of each outcome he/she directly accesses the oral exam. If a student does not achieve a sufficient number of points on the midterm exam, he/she cannot take the next midterm exam, and has to take the written exam.* *If he/she passes the written exam, he/she will proceed to the oral part of the exam.*

		The seminar paper is submitted within the agreed deadline, and certainly before the exam term. *The final grade is given on the oral part of the exam.*
3.10 Students obligations		• a full-time student has the right to sit for the exam if he attends classes for a minimum of 70% of the total prescribed number of hours • a full-time student who attends classes from 50 to 70% of the total prescribed number of hours can exercise the right to take the exam by completing additional teaching activities in agreement with the course teacher • a full-time student who attends a certain course for less than 50% of the prescribed number of hours enrolls in the course the following academic year • a part-time student has the right to sit for the exam if he/she attends classes for a minimum of 30% of the total prescribed number of hours • a part-time student who attends classes for 20 to 30% of the total prescribed number of hours can exercise the right to sit for the exam by completing additional teaching activities in agreement with the course teacher • a part-time student who attends the classes of a certain course for less than 20% of the prescribed number of hours re-enrolls in the course the following academic year
3.11 Written assignments		
3.12 Required reading	1.	Thorn, A.: Unity 5.x By Example, Packt publishing, 2016
	2.	Nenad Breslauer: Skripta Razvoj računalnih igara,2022
3.13 Additional reading	1.	Felicia P.: Unity 5 From zero to Proficiency (Foundations), 2015. 3.
	2.	Norton, T.; Learning C# by Developing Games with Unity 3D, Packt publishing, 2013.

4. ADDITIONAL INFORMATION ABOUT THE COURSE

4.1 Quality control	The quality of the program, teaching process, teaching skills and level of mastery of the material will be established by conducting a written evaluation based on questionnaires, and in other standardised ways and in accordance with the by-laws of the Međimurje University of Applied Sciences in Čakovec.
4.2 Contact the teacher	Students can contact the teacher during the office hours and during classes. All other methods of communication are arranged with the teacher. It is also possible to ask questions by e-mail, which will be answered in 48 hours at the latest. It is desirable for students to come as often as possible for any possible questions during the teacher's office hours.
4.3 Information about the course	It is the obligation of each student to be regularly informed about the course. All notifications about the classes or possible postponement of classes will be posted on the bulletin board and on the website of the University at least 24 hours in advance.

5. ELABORATION OF THEMATIC UNITS

Week	Topic	Course outcome
1.	Introduction to the course content.	I1

2.	General concepts of game development using specially adapted platforms for this purpose.	I1
3.	Game development platform editor, interface parts, and their purposes (Hierarchy, Project, Console, Inspector).	I1
4.	Different views (scene view, game view), repository of pre-made objects, scripts, and other assets (asset store). Main menu.	I1
5.	Coordinate system, scene, camera, and light sources. Changing views, adding elements to the scene.	I1
6.	Measurement and scaling, measurement units.	I1
7.	History of games.	I1
8.	Types of video games.	I1
9.	Game development team.	I1
10.	Elements of video game design.	I2
11.	Rules and mechanics of a video game.	I3
12.	User interface, elements for creating a user interface.	I4
13.	Stereoscopic 3D devices for virtual and augmented reality.	I5
14.	Virtual and augmented reality systems.	I1
15.	Presentations and evaluation of final student projects.	I1

Week	Topic	Course outcome
1.	Introduction to the course content. General concepts of game development using specially adapted platforms for this purpose.	I1
2.	Game development platform editor, interface components, and their purposes (Hierarchy, Project, Console, Inspector). Different views (scene view, game view), repository of pre-made objects, scripts, and other assets (asset store). Main menu.	I1
3.	Coordinate system, scene, camera, and light sources. Changing views, adding elements to the scene.	I1
4.	Creating a scene and adding basic objects, as well as adding textures and colors. Transforming (translation, rotation, scaling). Object management.	I1
5.	Creating terrain, peaks, valleys, trees, and water simulation. Landscape creation. Using the so-called "First Person Controller."	I2
6.	Simulating physical laws, simulating forces, gravity, and collisions. Mechanisms for introducing physical laws into the virtual 3D world (Rigidbodies, Colliders, Joints, Character Controllers). Applying forces to objects. Character creation.	I2
7.	Development of scripts using the C# programming language.	I3
8.	Platform libraries, the application's entry point, the most important platform classes (Unity: MonoBehaviour, GameObject). Debugging.	I4
9.	Accessing and locating components and other objects using code. Working with events, managing time, and frames per second (FPS). Unity: Using GameObject, MonoBehaviour, Transform, and RigidBody classes.	I4
10.	Mechanisms for creating game flow. Moving objects, detecting collisions, and tracking scores. Implementing a simple game.	I4
11.	Layout and arrangement, buttons, switches, input fields, and other user interface elements.	I4
12.	Stereoscopic 3D view, devices for virtual and augmented reality.	I5
13.	Position of the player's head, specific hardware and software components for individual virtual and augmented reality systems.	I5

14.	Preparing a project for execution on devices for virtual and augmented reality systems.	I5
15.	Presentations and evaluation of final student projects.	