

## Specification for the book of courses

<b>Study program</b>		Electrical Engineering and Computer Science		
<b>Module</b>		Electronics - Multimedia technologies		
<b>Type and level of studies</b>		Undergraduate Academic Studies		
<b>The name of the course</b>		Gaming		
<b>Lecturer (for lectures)</b>		Jovanović S. Goran		
<b>Lecturer/associate (for exercises)</b>		Cvetković S. Stevica		
<b>Lecturer/associate (for OFE)</b>		Cvetković S. Stevica		
<b>Number of ECTS</b>	5	<b>Course status (obligatory/elective)</b>	Obligatory	
<b>Prerequisites</b>				
<b>Course objectives</b>				
Students learn how to create 2D and 3D multimedia applications for all platforms. Connecting the knowledge with image, audio and video processing, 3D modeling and animation, and connecting this knowledge with programming.				
<b>Course outcomes</b>				
Students will connect the acquired knowledge in the field of image processing, audio and video, as well as knowledge of 3D modeling and animation with programming and creating applications. Such multimedia applications can be adapted to most hardware platforms and to all important operating systems.				
<b>Course outline</b>				
<b>Theoretical teaching</b>				
Basics of C # programming language. Creating, drawing a scene, terrain, "sky", creating an environment. Loading or creating 3D objects and models, textures, materials. Lights and cameras. User input (controller), mouse, keyboard, joystick, touch screen and more. Built-in commands, scripts for objects. C # script to control the game. Programming collider, collision, raycasting, object physics. Graphical user interface. Buttons, menus, drop menus, textures. Particle systems. Navigation. Animation and control of objects. Sound programming. Upgrade with add-ons.				
<b>Practical teaching (exercises, OFE, study and research work)</b>				
Example of sketch a terrain, skybox and environment. Loading 3D objects, models, textures and materials. Colliders, collision and rigid body for objects. Lights and cameras. #C script. User input (controller) with a mouse and keyboard. Built-in commands. Example of writing scripts for objects. Graphical user interface. Buttons, menus, drop down menus, textures. Particle systems (smoke and fire). Basic script and class. Object control. Example of sound programming. Pathfinding. Examples of complete projects of 2D and 3D games.				
<b>Textbooks/references</b>				
1	John Shaper, Microsoft Visual C# 2012, CET, 2013.			
2	Sue Blackman, Beginning 3D Game Development with Unity, Apress, 2011.			
3				
4				
5				
<b>Number of classes of active education per week during semester/trimester/year</b>				
<b>Lectures</b>	<b>Exercises</b>	<b>OFE</b>	<b>Study and research work</b>	<b>Other classes</b>
2	1	1	0	0
<b>Teaching methods</b>				
Lectures, exercises, consultations.				
<b>Grade (maximum number of points 100)</b>				
<b>Pre-exam duties</b>		<b>Points</b>	<b>Final exam</b>	<b>Points</b>
<b>Activity during lectures</b>		10	<b>Written exam</b>	
<b>Exercises</b>		10	<b>Oral exam</b>	40
<b>Colloquia</b>				
<b>Projects</b>		40		