



Introduction to Game Programming with Unity

The programming language for this course is JavaScript.

This course is aimed at anyone willing to learn about game programming with a game engine.

The course starts with an introduction to game programming industry; who are the game development actors, how the market looks worldwide, what are the potential legal issues you could face. Development phases, legal issues, financing and monetization are covered.

Unity3d game engine is then introduced. The course includes 8 chapters of theory material, which is first giving an introduction to different topics and then followed by a laboratory-like approach; you will be creating a fully playable Arkanoid -style game during the course. The game is built alongside Viope; you need to download and install Unity3D game engine on your computer (available for Windows and Mac OS X).

Requirements – Prior Knowledge

The student should have at least elementary skills on computer programming. Also any experiences from object-oriented programming is a plus. Calculus is constantly involved, high school or university basic level mathematics skills are recommended.

Table of contents:

1 Introduction to Game Development 1.1 Definition of a video game 1.2 Game types 1.3 Game development actors 1.4 Designing a game 1.5 Development phases 1.6 Legal issues with game development 1.7 Finance and monetization in games 1.8 Localization	2 Unity Basics 2.1 First contact with Unity 2.2 Interface 2.3 Scripting 2.4 Introduction to UnityScript	3 Basic Components 3.1 Creating a project 3.2 Light 3.3 Adding our player 3.4 Input Class 3.5 Parenting 3.6 Prefabs 3.7 Importing assets 3.8 Rigidbody 3.9 Instantiate and Destroy 3.10 Collision 3.11 Audio
4 Terrain Editor 4.1 Terrain editor 4.2 Relief 4.3 Adding colors 4.4 Trees 4.5 Grass 4.6 Settings 4.7 Water 4.8 Skybox 4.9 Fog	5 Starting the game 5.1 Starting to play 5.2 Adding the level 5.3 Frame Miss 5.4 Normal map or Bump map 5.5 Getting the sphere to act 5.6 About Input 5.7 About collision 5.8 Collision that does not	6 Continuing the game 6.1 Instantiating our chests 6.2 Adding some logic 6.3 Adding power-ups 6.4 Adding more power-ups 6.5 Animating our gun 6.6 Some mathematics 6.7 Particles 6.8 Background music 6.9 Completing the level

4.10 Terrain scripting	collide	
7 Adding GUI	8 Optimization and Build	
7.1 Score and lives	8.1 Avoiding code when possible	
7.2 Introduction of level and finishing the level	8.2 Lookups	
7.3 Pause menu	8.3 Variable types	
7.4 Introduction screen	8.4 Stats	
7.5 Main Menu	8.5 Sleepy AI	
7.6 Preparing the build	8.6 Built-in arrays	
	8.7 Object pooling	
	8.8 Textures and optimized 3D models	
	8.9 Benchmark	
	8.10 Level of Design (LOD) and model optimization	
	8.11 Building our game	