Shawn David Vella

Shawn.vella.b43358@mcast.edu.mt

Programming for computer games

Notes:

1. This is a guided Home Assignment

2. There will be review days, where your lecturer will guide you on your assignment completion. It is strongly suggested that you adhere to the review days to avoid assignment cramming and avoid losing marks.

3. You may refer to websites, forums, etc. for this assignment but you must include a reference (Harvard style) at the end of the assignment.

4. The final documentation should be uploaded on Turnitin by

**Monday 18 December 2017 12:00pm**

5. The program files must be uploaded (last commit) on Github,

also by **Monday 18 December 2017 12:00pm** and the link

shared with your lecturer.

6. There will be no printed assignment submission, just soft copy.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

***Scenario:***

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

You have been asked to design and develop a 2-player ball game.

The game should have a Play Area, with borders and 2 goal scoring posts. 2 players should guard their own goal and try to score in the opposing player’s goal by reflecting a ball into the goal.

The player who reaches a set amount of score first wins and the game goes to another level, presenting the same play area, but increasing difficulty such as obstacles and increased ball

movement. Scoring in Level 2 should be higher than Level 1 and include static obstacles.

Level 3 should be similar to Level 2 but having moving obstacles.

There should be at least 3 levels in the game.

The players’ scores should be shown in UI Text and at the end of the last level, the player with the highest score wins.

The game should have 1 Menu Screen (Start or Quit), 3 Game

Levels, 1 End Screen showing the scores and who won.

Player 1 should be controlled by the keyboard while player 2 should be controlled using the mouse.

For the game you can make use of any type of assets.

**Question 1 (AA1, AA2): *14 marks***

a. Select 2 game engines *(1 mark)* and list 4 reasons *(1.5 marks each)*

why one particular game engine was selected (giving advantages of

the game engine) and why the other was not selected.

*7 marks*

Game Engines: Unity Game Engine; Unreal Engine.

Engine Chosen: Unity Game Engine.

Advantages:

* Unity is easy to use compared to Unreal Engine and therefore has a simple learning curve for new users.
* Being a popular option among developers, it has a great online community support; more than unreal engine does, meaning new users trying to learn unity can easily find help in others.
* Debugging and tweaking is far easier since unity displays the variables during gameplay, which allows the developer to debug during runtime.
* Having a built-in support system for PhysX Physics engine helps programmers develop and observe on real-time simulations. Having physics ready helps the developer immensely to not have to make their own.

Unreal Engine was not chosen simply because Unity offers too many advantages and new user ease, especially with having new developers like us.

b. Select 2 programming languages *(1 mark)* used in game

development and choose one to support the game engine chosen

and list 4 of its features *(1.5 marks each).*

*7 marks*

Languages: C#; JavaScript.

Language Chosen: C#.

* C# is a modern, simple and object-oriented language coming from C++ and Java, making it easy to pick up and meaning people accustom to statically typed languages are advantaged.
* C# has a better editor support which, in turn, gives the developer loads more control.
* .
* .

**Question 2 (SE1): *10 marks***

Prepare detailed design documents for the game:

*a.* One Game design document (State Diagram) should contain an

overview of the whole game, from the Start Menu till the end showing

correctly the interactions between the scenes.

*5 marks*



b. The second design document should contain a detailed State

Diagram of either Level 2 or Level 3 of the game showing correctly all

events and interactions happening in the scene.

*5 marks*



**Question 3 (KU 4): *5 marks***

In not less than 100 words, explain why compression is needed when using media assets such as images, videos and audio. Provide examples.

Compression is reducing the number of bits needed to transmit or store data. Compression strategically enables unity to reduce storage space, reduce memory utilization or reduce bandwidth, making previously impractical or uneconomic projects (due to storage, hardware or transmission resources) possible.

Unity offers shelf compression for medial assets like textures, sounds, asset bundles and meshes. Without these systems, Unity would require an obscure amount of memory on an impractical number of platforms; mostly mobile devices.

An example of this would be an advanced 2D game containing several assets such as large textures, many background audio files and not to mention a large variation of script files meant to handle events associated with the game.