A Simple World in Unity

**Name:** Rhys Streefland

**Student Number:** 22011021

# Screenshot



Figure 1 Exterior of building

# Sources used

A number of online sources were used to assist in the creation of the world. These are documented below in bullet point form for brevity.

* The YouTube videos by Jimmy Vegas [1] were used to learn the basics of Unity.
* The grass and bricks textures used in the first video [2] were also used in this world.
* The ReCalcCubeTexture script [5] was used to ensure that the textures did not stretch when cubes were scaled during editing
* The Unity documentation [4] was used to learn the basics of C# scripting with Unity
* The Sweet Home 3D [5] and Archive3D [6] websites were used to obtain external 3D models for items such as doors and furniture
* The Door Manager script [7] was used to enable the opening and closing of doors
* Floor [8] [9] and roof [10] textures were used to texture the floors and roofs
* A roof model [11] was used for the roof because Unity lacks the features required to create a roof
* A tree pack was used to enhance the scenery [12]
* A fence model was used to enhance the scenery [13]

[1] J. Vegas. (2015, June). Unity 5 Tutorial For Beginners - How To Make A Game - Part 001. [Online]. Available: <https://www.youtube.com/watch?v=G9BdFZ2MCXc>

[2] J. Vegas. (2015, June). Unity 5 Tutorial For Beginners. [Online]. Available <http://jvunity.weebly.com/teal-series.html>

[3] Dsphar. (2016, March). Cube\_Texture\_Auto\_Repeat\_Unity. [Online]. Available: <https://github.com/Dsphar/Cube_Texture_Auto_Repeat_Unity>

[4] Unity Technologies. (2016). Creating and Using Scripts. [Online]. Available: <https://docs.unity3d.com/Manual/CreatingAndUsingScripts.html>

[5] eTeks. (2017). Search free 3D models. [Online]. Available: <http://www.sweethome3d.com/searchModels.jsp?model=window>

[6] Archive3D. (2016). Archive3D. [Online]. Available: <http://archive3d.net/>

[7] A. Ameye. (2017, January). Door Manager Script Free. [Online]. Available: <https://www.assetstore.unity3d.com/en/#!/content/38694>

[8] J. Guilherme. (2015, May). PBR Tile Texture Floor. [Online]. Available: <https://www.assetstore.unity3d.com/en/#!/content/36243>

[9] Mikelarg. (2015, July). Wooden Floor Pack. [Online]. Available: <https://www.assetstore.unity3d.com/en/#!/content/31492>

[10] V. Lang. (2012, December). Roof textures. [Online]. Available: <https://www.assetstore.unity3d.com/en/#!/content/5844>

[11]

<https://3dwarehouse.sketchup.com/model.html?id=f490aae1ca02ab73e921cf4ce7c80e0>

[12] Laxer. (2016, February). Mobile Tree Package. [Online]. Available: <https://www.assetstore.unity3d.com/en/#!/content/18866>

[13] D. Nenenko. (2016, September). Stylized Fence. [Online]. Available: <https://www.assetstore.unity3d.com/en/#!/content/71535>

All web addresses were verified on 30 January 2017.

# Code written

The door opening script did not function correctly when initially implemented. The virtual hinge around which the door would rotate was in the wrong position. This meant that the door would intersect with the nearby wall when it was opened. This issue was fixed by changing the code of the Door.cs file. The change is documented below.

## Before Modification

// RIGHT

if (HingePosition == PositionOfHinge.Right)

{

// CALCULATE

if(transform.localScale.x > transform.localScale.z)

{

HingePosCopy.x = (PosDoorX + (ScaleDoorX / 2 \* CosDeg));

HingePosCopy.z = (PosDoorZ - (ScaleDoorX / 2 \* SinDeg));

HingePosCopy.y = PosDoorY;

HingeRotCopy.x = RotDoorX;

HingeRotCopy.y = -InitialAngle;

HingeRotCopy.z = RotDoorZ;

}

else

{

HingePosCopy.x = (PosDoorX - (ScaleDoorZ / 2 \* SinDeg));

HingePosCopy.z = (PosDoorZ - (ScaleDoorZ / 2 \* CosDeg));

HingePosCopy.y = PosDoorY;

HingeRotCopy.x = RotDoorX;

HingeRotCopy.y = -InitialAngle;

HingeRotCopy.z = RotDoorZ;

}

}

## After modification

// RIGHT

if (HingePosition == PositionOfHinge.Right)

{

HingePosCopy.x = (PosDoorX - ScaleDoorX/10);

HingePosCopy.z = (PosDoorZ);

HingePosCopy.y = PosDoorY;

HingeRotCopy.x = RotDoorX;

HingeRotCopy.y = -InitialAngle;

HingeRotCopy.z = RotDoorZ;

}

# Navigation commands

The world uses a simple FirstPersonCharacter for navigation hence the control scheme is similar to that of most computer games. The keyboard is used to control the direction of walking and the mouse is used to control the ‘looking’ direction. The basic controls are listed in the table below.

|  |  |
| --- | --- |
| **Action** | **Control** |
| Look around | Move **mouse** in desired direction |
| Walk forwards | **W** key or **UP** arrow key |
| Walk backwards | **S** key or **DOWN** arrow key |
| Walk left | **A** key or **LEFT** arrow key |
| Walk right | **D** key or **RIGHT** arrow key |
| Jump | **Spacebar** key |
| Open/close door when near | **E** key |
| Exit | **Alt+F4** keys |