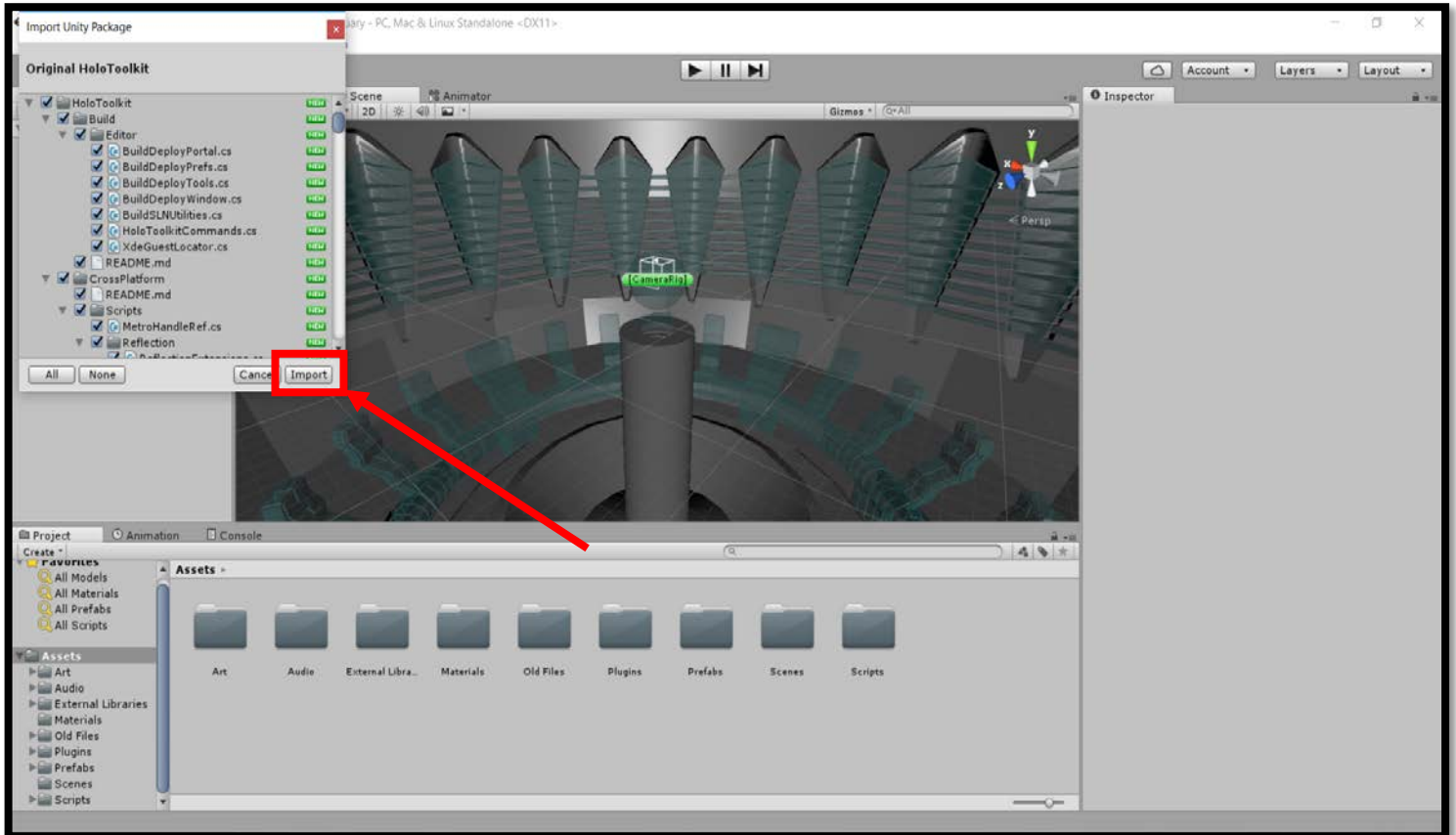


Importing Project Files and Naming Conventions Guide:

This is a guide for how to structure and name file types when adding them to the build in Unity. Please stick to this as much as possible and let me know of any required changes to the process.

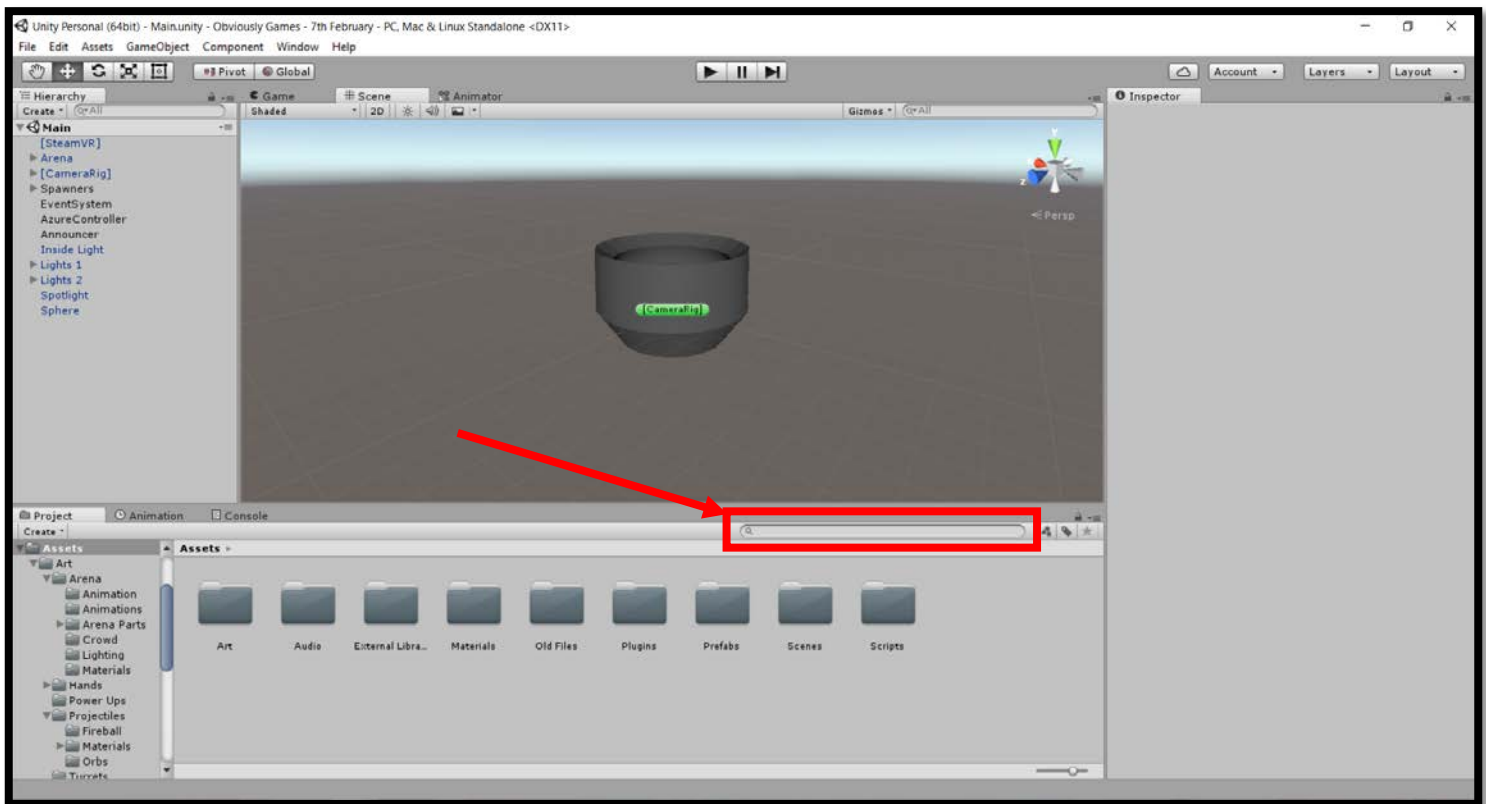
1. Import the file(s) into Unity

Import the files required by dragging them into the project window in the Unity editor. If importing a Unity package a window will be displayed to select the correct files as shown below. Ensure all required assets are selected.



2. Locate the imported files

When importing files, Unity may automatically place them in another folder than desired. To locate them you can find them manually or search using the bar in the project window.

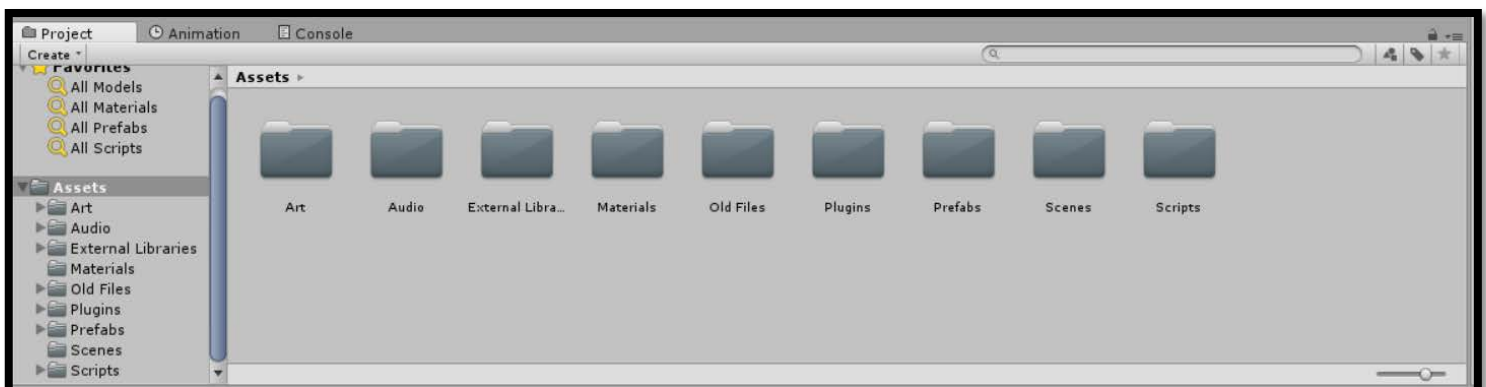


3. Move the files to the correct location

After importing the files move the files to be structured correctly in the build folders. How to save files of each type can be seen below:

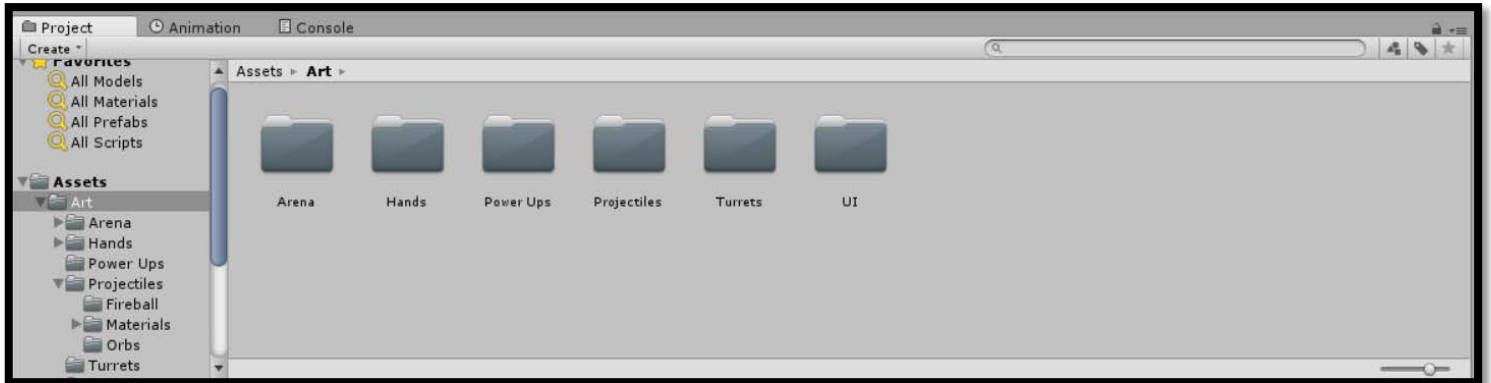
General Overview:

Here is an overview of everything the assets folder contains in the project, any unused files should be placed in the "Old Files" folder until the creator of those assets confirms they can be removed.

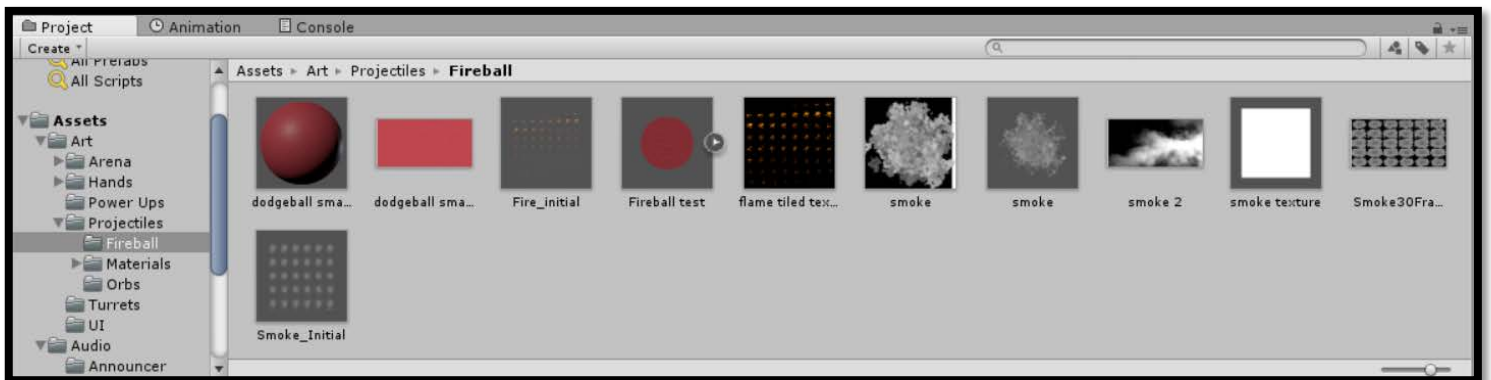
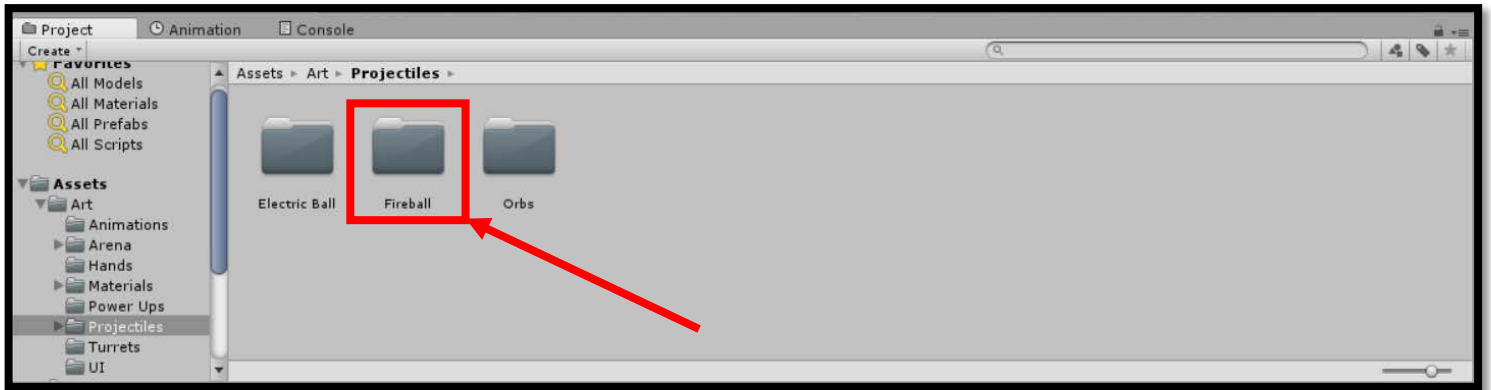


Art:

Art assets should be stored in the group they relate to e.g. UI items should be in the “UI” folder, projectiles should be in the “projectiles” folder, etc.

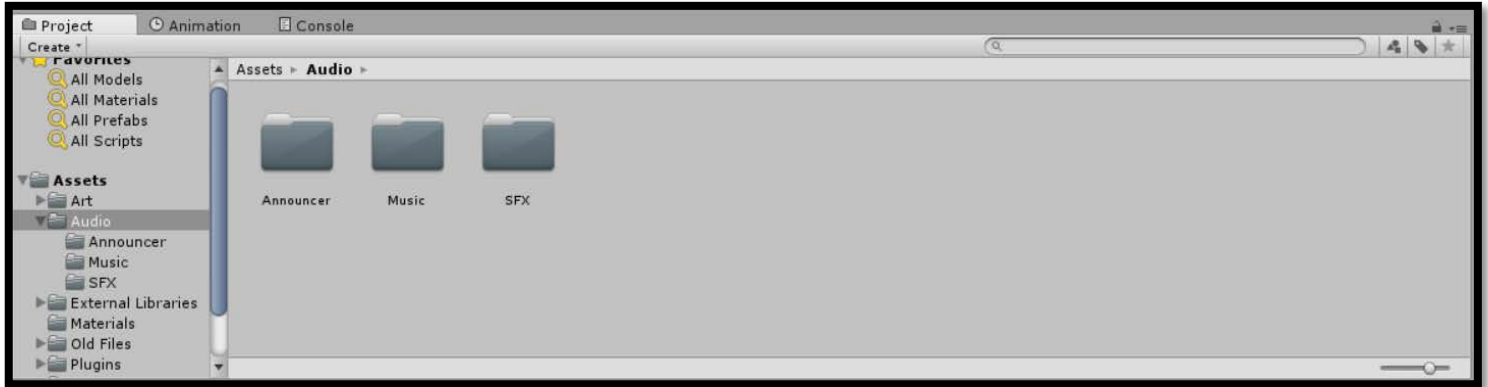


Once in the correct group a folder should be created that contains all the files related for the specific asset (materials, normal maps, animations, etc.) An example is shown below:



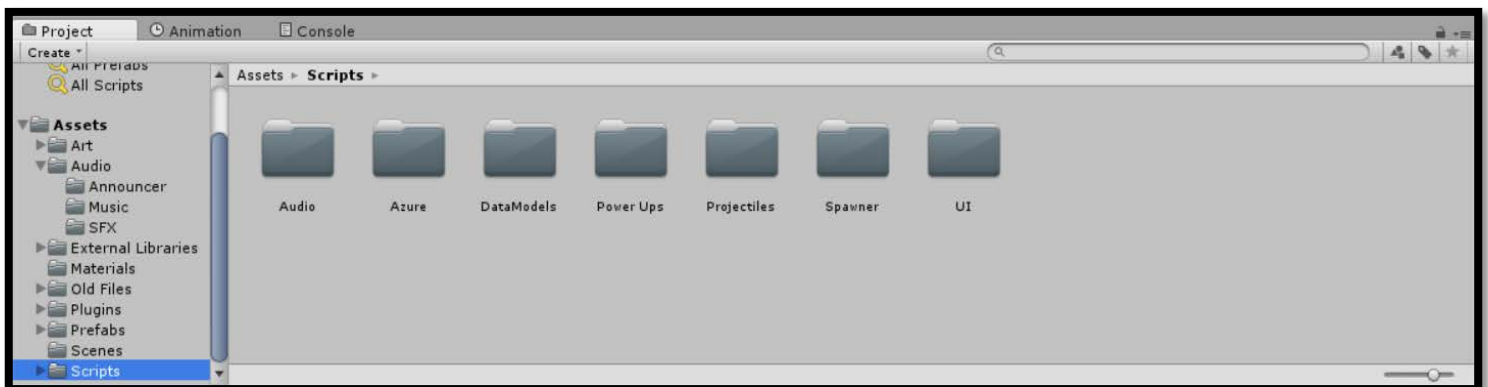
Audio:

Audio files should be saved in the .wav file format wherever possible. They can be converted within Audacity if they are a different format. Unity occasionally adds blank data to audio tracks saved as mp3 files, preventing them from looping and this avoids this. Once imported, files should be stored as music, sound effects (SFX) or in the announcer folder for narrator lines.



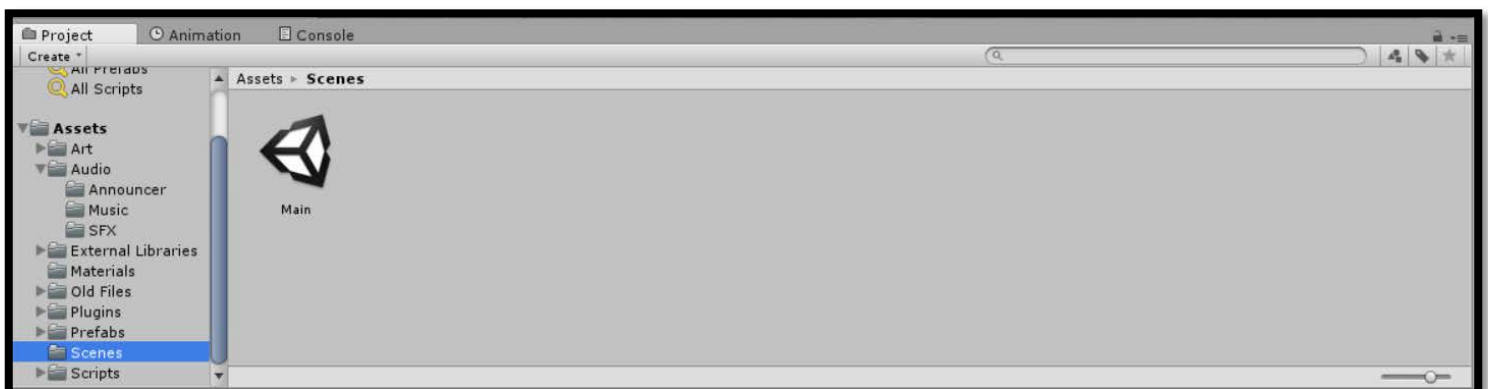
Scripts:

In the "Scripts" folder, scripts are stored based on the area they are associated with. Only C# files should be present here.



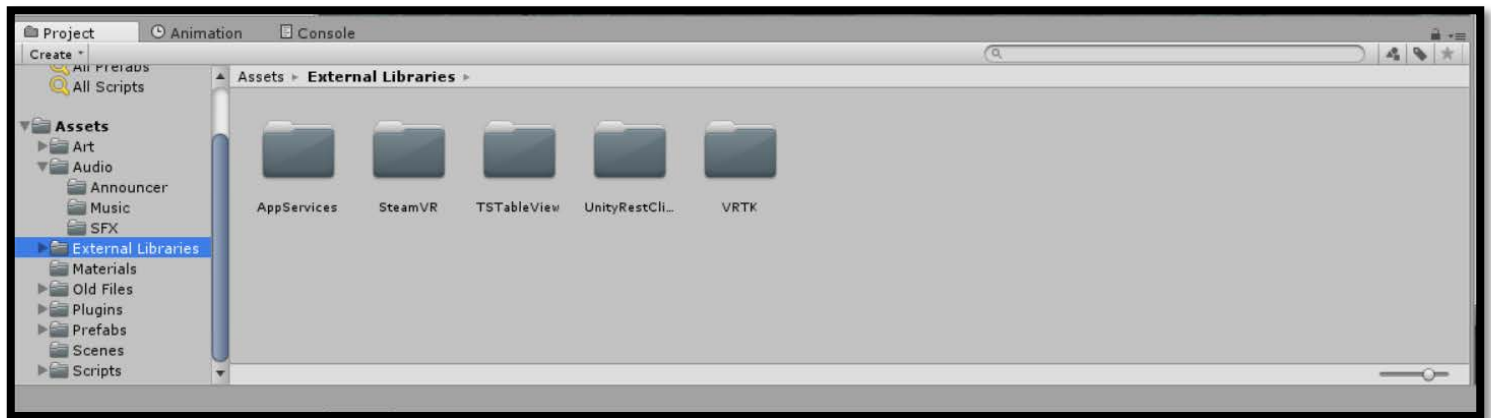
Scenes:

Any additional levels can be stored in the "Scenes" folder, currently only one scene is needed to run the game but others will be added such as Menu screens at a later date.



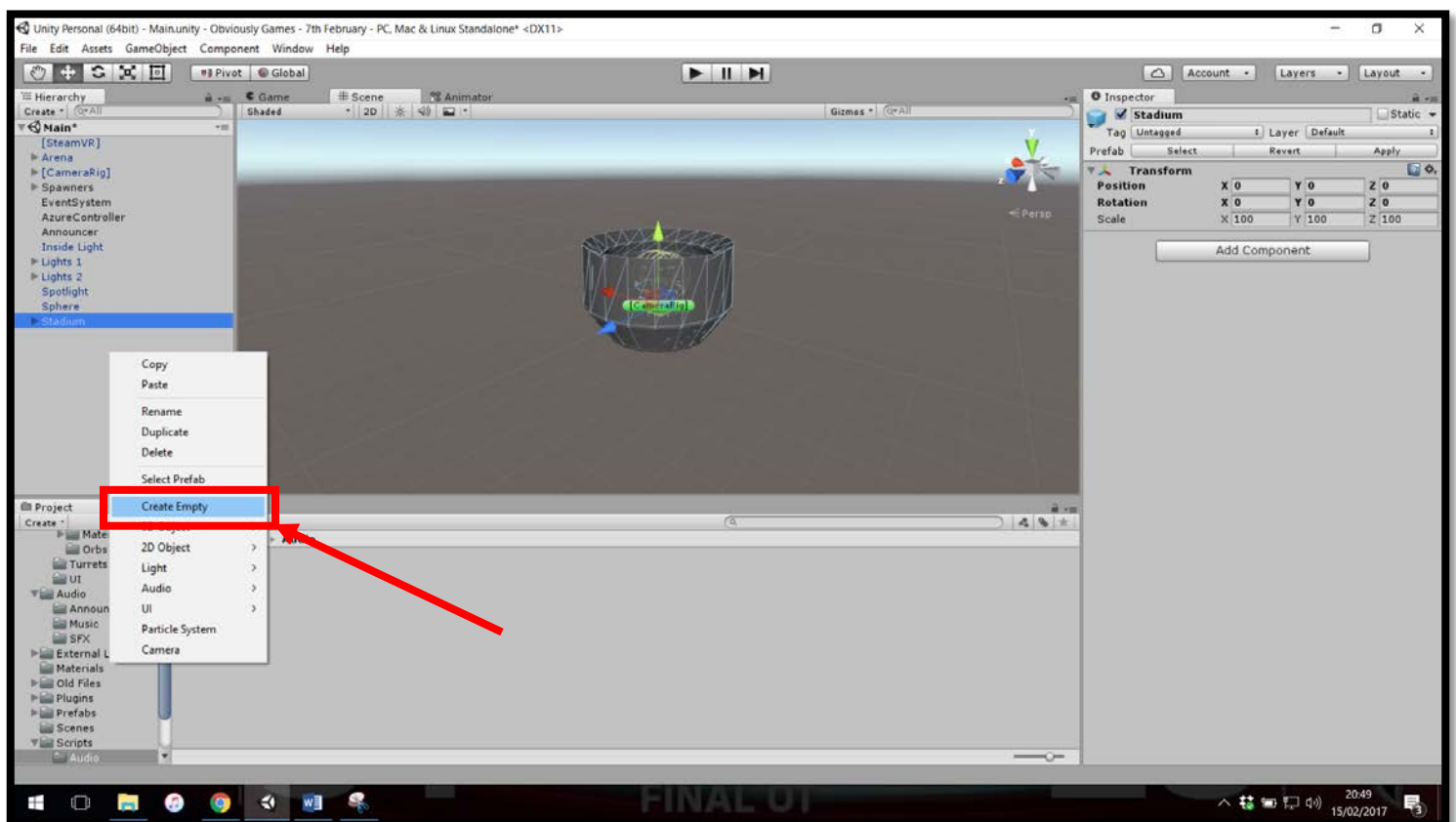
External Libraries:

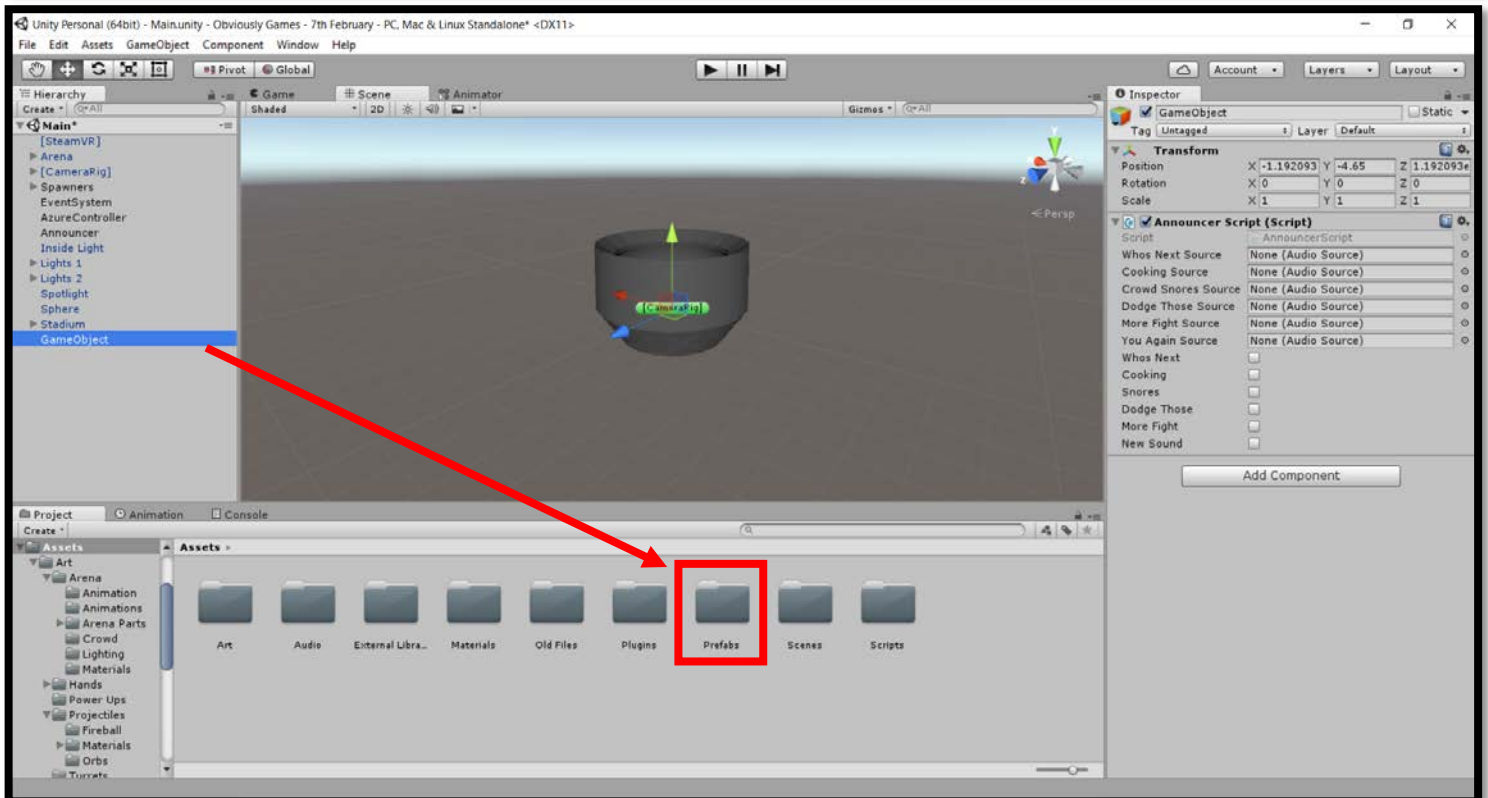
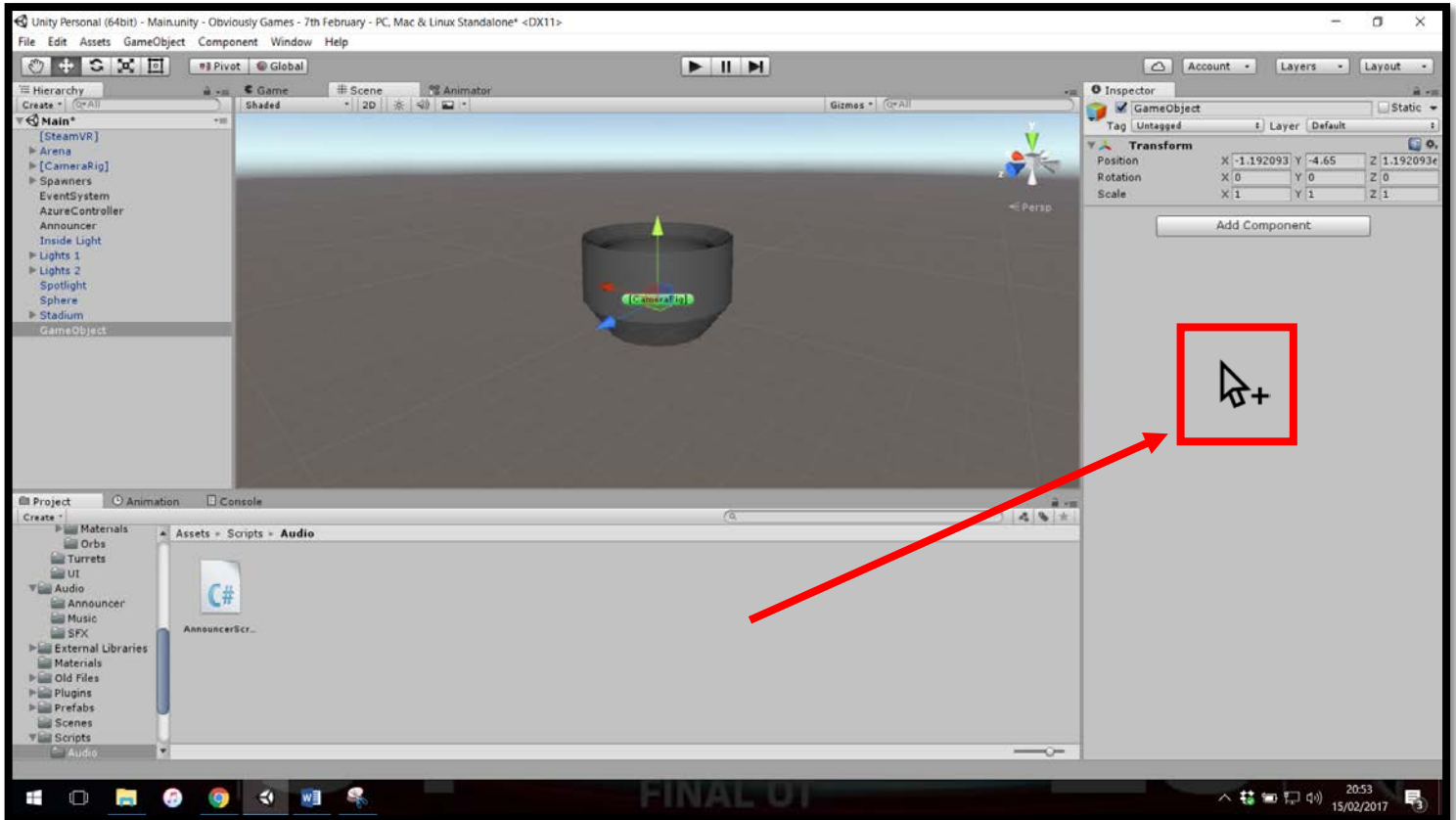
As we are using multiple open source libraries to assist with the technology requirements of the brief, all external libraries should be kept in this folder with license agreements also imported alongside them.



4. Prefab any created Game Objects in the prefab folder

Once the required assets are imported, create a new game object by right-clicking in the Hierarchy and selecting "create empty". Then, construct the game object by connecting all the components required by dragging them into the inspector on the right. After this is done, ensure the objects position is set to 0 on each axis and drag the game object into the "Prefabs" folder. This allows the project to save the object with all required assets attached.





5. Save the project and upload to source control

Once complete save the scene and project and upload all changes to the correct branch in source control.