

ThunderBlossom Style Guide

Table of Contents

[Folder Names](#)

[Always Use PascalCase](#)

[Never Use Spaces](#)

[Never Use Unicode Characters And Other Symbols](#)

[Asset Names](#)

[Always Use PascalCase](#)

[Never Use Names With 'Script' or 'Model'](#)

[Developers Folders](#)

[Purpose](#)

[Use Your Name And Initials](#)

[Hierarchy](#)

[Use Empty GameObjects As Folders](#)

[Project Files](#)

[Never Put Files In Assets/](#)

[Always Use Appropriate Folders](#)

All structure, assets, and code in any project should look like a single person created it, no matter how many people contributed.

Folder Names

These are common rules for naming any folder in the content structure.

Always Use PascalCase

PascalCase refers to starting a name with a capital letter and then instead of using spaces, every following word also starts with a capital letter. For example, DesertEagle, RocketPistol, and ASeriesOfWords.

Never Use Spaces

Never Use Unicode Characters And Other Symbols

Do not use other characters outside `a-z`, `A-Z`, and `0-9` such as `@`, `-`, `_`, `,`, `*`, and `#`.

Asset Names

Always Use PascalCase

See [Folder Names](#).

Never Use Names With ‘Script’ or ‘Model’

If you make a script for player movement never name it PlayerMovementScript.

Mentioning ‘Script’ at the end is pointless, we can see it is a script by the extension or Unity search filters.

_Developers Folders

Purpose

The _Developer folder in Assets/ is for individual testing. If you are using a debugging asset (example would be a blank character model, testing a script, unused material, etc) you would put in it _Developers/YourName.

You can reference other assets from people’s folder and there is no need to duplicate assets.

Use Your Name And Initials

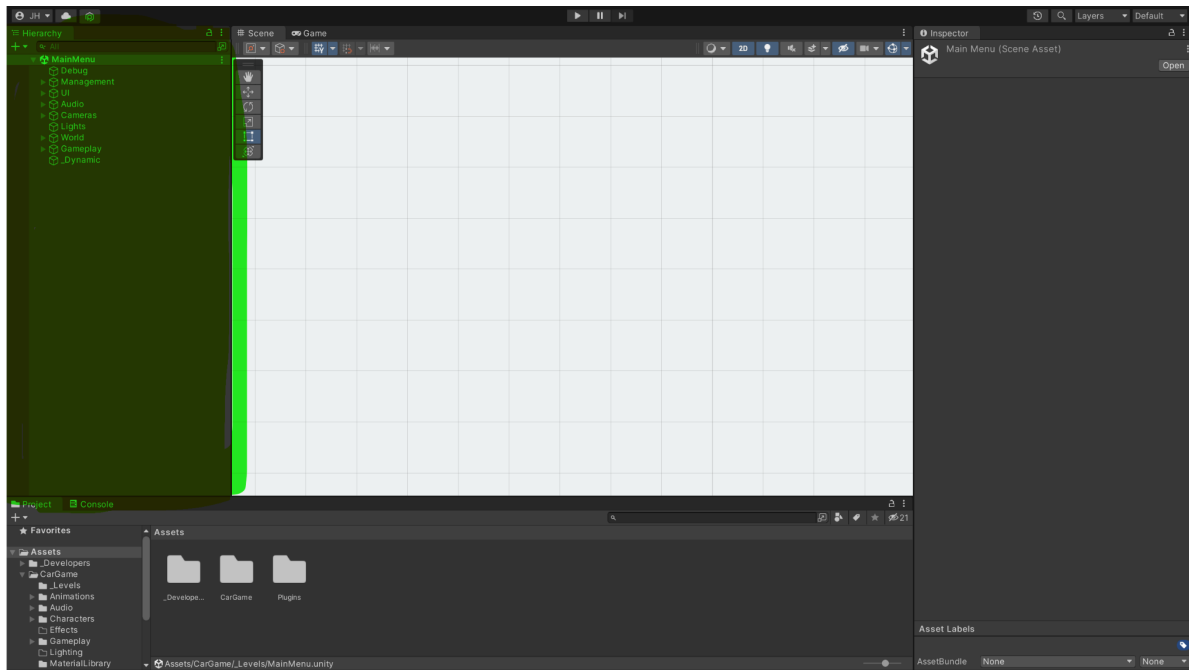
You will have your own folder, no matter your role. Your folder will be named with your first name and initials after.

Example

For Jack Hughes-King, it would be _Developers/**jackhk**.

Hierachy

This is where the “**Hierachy**” section is. Also known as **World Outliner** in Unreal Engine.

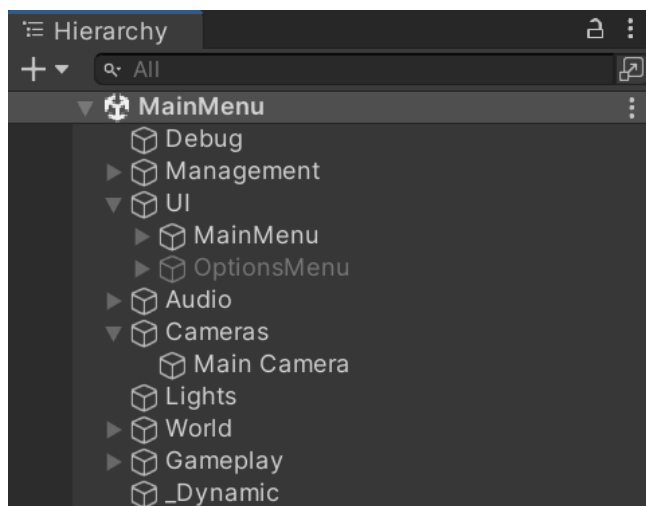


Use Empty GameObjects As Folders

Example

LevelName

- Debug
- Management
- UI
- Audio
- Cameras
- Lights
- World
- Gameplay
- _Dynamic



Management is for scripts that do not attach to in-game objects such as GameManager, AudioManager, SceneManager, etc

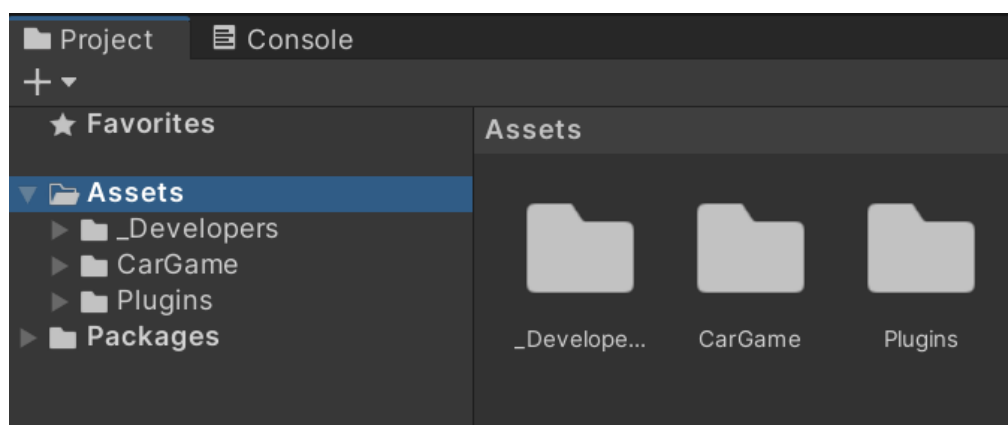
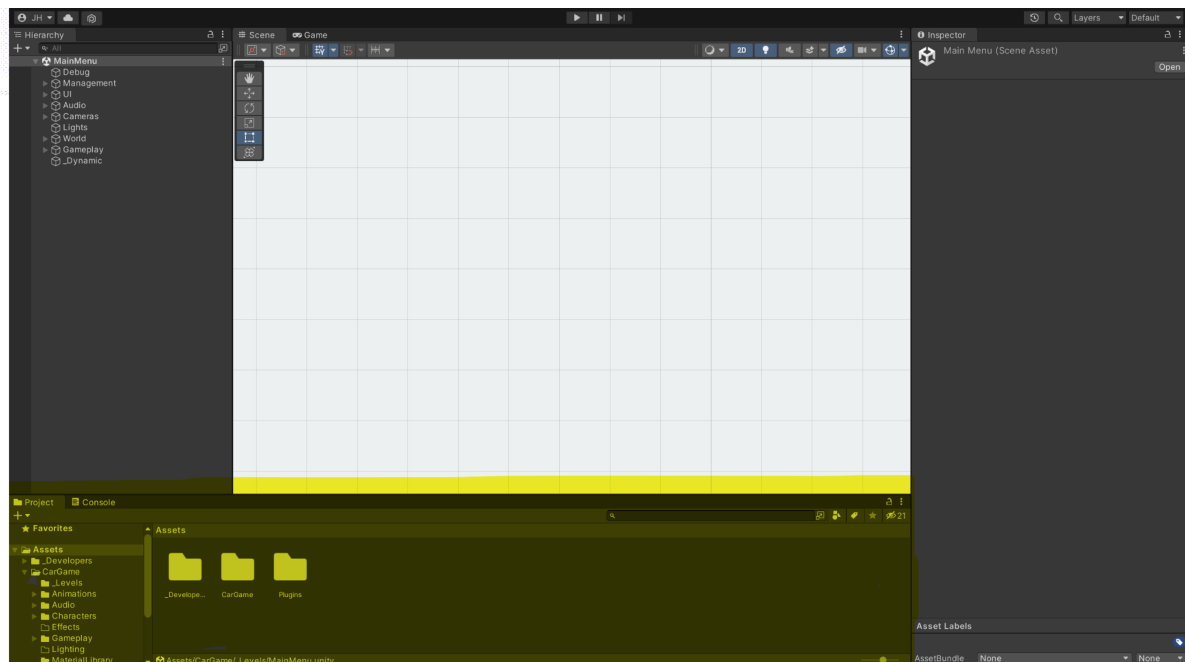
World is for static objects such as walls, floors, platforms, trees, etc.

Gameplay is for characters, enemies, pick-ups, etc.

_Dynamic is for temporary objects such as bullets, ragdolls, some effects, etc.

Project Files

This is where the “**Project**” section is. Also known as **Content Browser** in Unreal Engine.



Never Put Files In Assets/

Only **_Developers**, **ProjectName** and **Plugins** should exist in Assets.

All **testing** assets go in _Developers/YourName (example _Developers/jackhk)

All **final** assets go in ProjectName/ (example ProjectName/UI /Target.png)

Libraries, add-ons and anything external go in Plugins (example Plugins/TextMeshPro)

Always Use Appropriate Folders

Expect structure to look something like this:

Example of ProjectName/

ProjectName

— *_Levels*

— *Effects*

— *Materials*

— *Prefabs*

— *Scripts*

— *Sound*

— *UI*