Hanzaki

Technisch Ontwerp

# Code Structure

**C#**

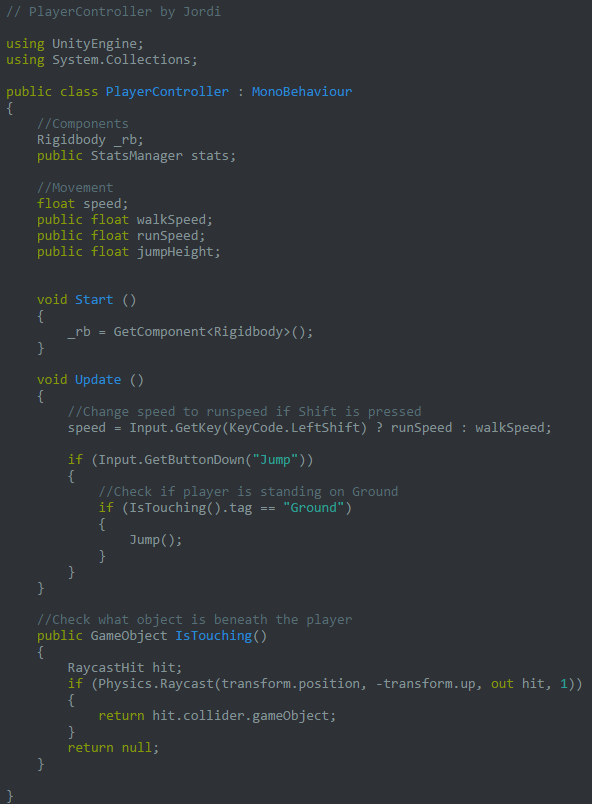
All scripts will be coded in Unity Script (C#)  
We chose this over Js to make optimal use of Unity Script and to create various classes.

**Classes**The script will be planned out completely before they will be added in the game. This will be displayed in a class diagram. The class diagram can be found in this document.

**Scripting Format**- Brackets on new lines.  
- Encouraged use of creating functions.  
- Private Component Variables with \_ in front of variable name.  
- Correct use of private vs public variables.  
- Comments on at least all functions. Comments above the commented function.  
- Start a new script with //[Scriptname] by [Scripter] in the first line.  
- Correct usage of capital letters in variables / classes. As taught in the lessons.

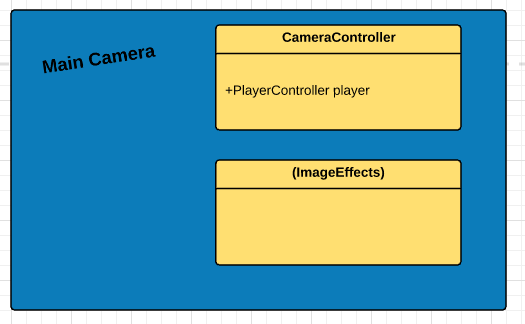
**Script validation**All scripts will be checked by the lead developer for approval.

Example Script



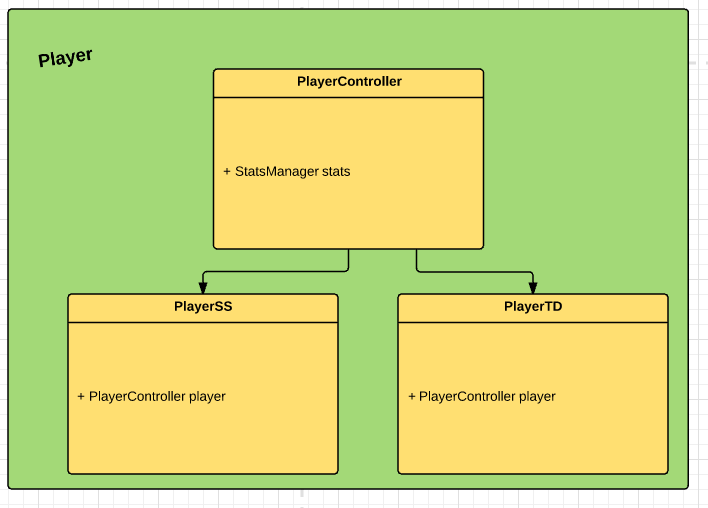
# Class Diagram

The big box is the prefab. The yellow boxes are the script components.



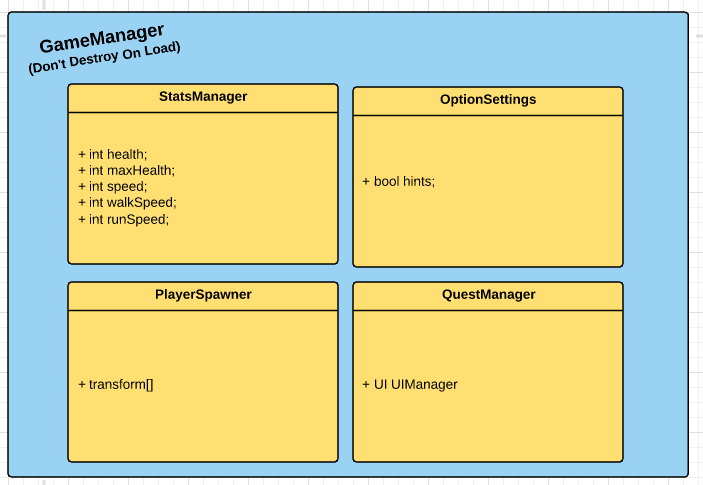
**CameraController**  
The Main Camera will be controlled by the CameraController script. With lerping we will achieve smooth movement for the camera and make restrictions at edges of the level.

**(ImageEffects)**There will be various ImageEffect scripts attached to the Main Camera



**PlayerController**Basic variables and functions that are used in both the Side-Scrolling and Top-Down sections will be in this script.

**PlayerSS**Unique Side-Scrolling variables and functions will be in this script. Inherits from PlayerController.  
  
**PlayerTD**Unique Top-Down variables and functions will be in this script. Inherits from PlayerController.



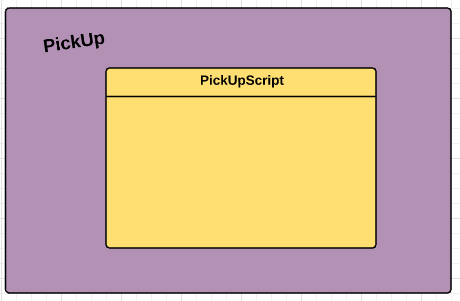
The GameManager will have a DontDestroyOnLoad function. This object will be instantiated in the BuildScene and keeps existing throughout the game.

**StatsManager**All stats can be accessed from here

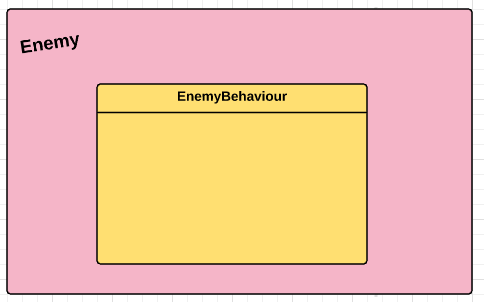
**OptionSettings**The options will be saved here.

**PlayerSpawner**When the player leaves a map, the PlayerSpawner will keep track of where he left so her re-enters on the correct location

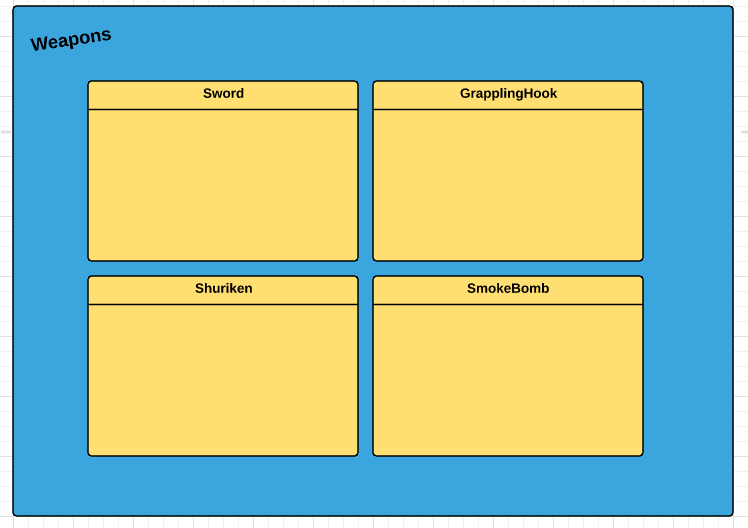
**QuestManager**We keep track of all quests here and their statuses.



**PickUpScript**PickUp Objects will have a script. This will function on collision enter with the Player.

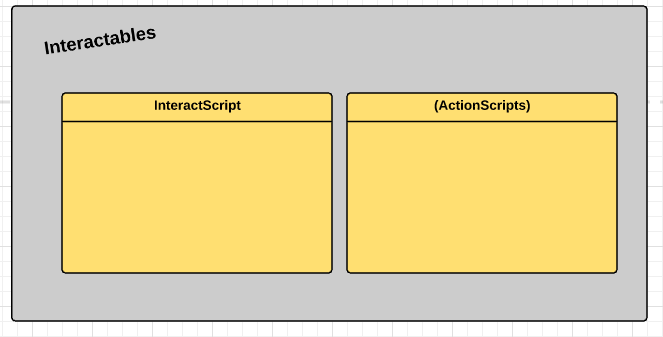


**EnemyBehaviour**The enemy AI will be in this script.



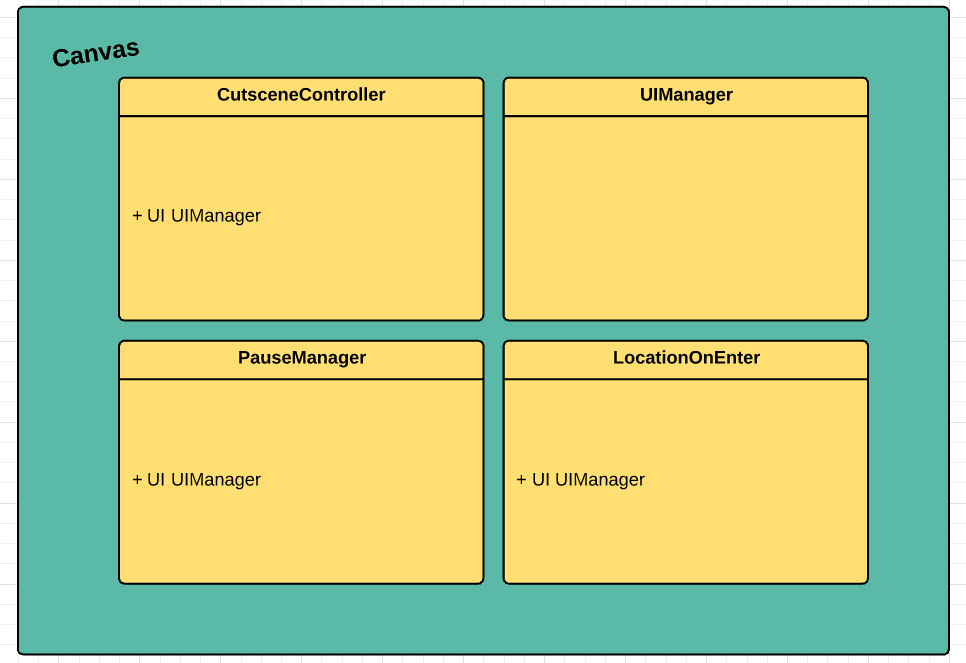
**Weapons**

All weapons will have their script. In each script the weapon can be controlled.



**InteractScript**

InteractScript will function as a link between the object that is interactable and the ActionScript that will be activated.

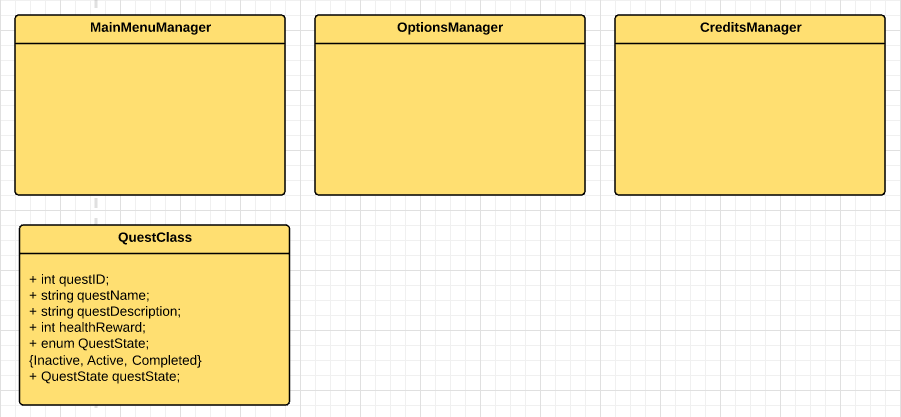


**CutsceneController**The cutscene controller will send text to the UIManager and will control the camera.

**UIManager**The UIManager will handle everything that happens on the HUD

**PauseManager**This manager will display the Pause and modify the timescale

**LocationOnEnter**When the player enters a new level, this script will play the animation that displays the name of that level



There are some other scripts.  
There are various scripts for handling the menus.  
QuestClass will be used for the questManager.