

Illumisoft

Skill Drive

[Get Started](#)

Welcome.....	2
About the Game.....	2
Rules	2
Difficulty.....	2
Supported Input.....	2
Scenes.....	3
Build Settings.....	3
The following screenshot shows the Build Settings. The first and second slot are reserved for the main menu and the level selection. All following slots can be used for the game levels.	3
Prefabs.....	5
Level System.....	6
Level Data.....	6
Creating new Levels.....	6
The Systems Prefab.....	7
Main Camera	8
Post Processing.....	9
Music Manager	10
Scene Loader	10
The Game Manager.....	11
Adjusting the Start Delay and Time Limit	11
The Vehicle.....	12
Adjusting the Driving Mechanics and the Appearance	12
Support.....	13

Welcome

Thank you for buying Skill Drive! This document has been designed to help you getting started with the project. The first chapters will give you an overview of the rules of the game and how the project has been designed and is structured. In the later chapters you will be introduced to the most important game prefabs and elements, like the game manager and the players vehicle.

About the Game

Rules

The rules of the game are pretty easy. In each of the **15 levels**, the player has **10 seconds** to collect all the **blue cubes**. For this task it is not important in which order the cubes are collected.

Difficulty

The game starts pretty easy, but the difficulty of the game increases over time, with the later levels being really hard to master.

Supported Input

The game supports keyboard and gamepad input using Unity's new Input System.

Scenes

To keep the content of the game clean and easy to work on, it has been split up among different scenes. All scenes of the project can be found in the *"Assets/Illumisoft/Skill Drive/Scenes/"* folder.

- **Menu**

The main menu of the game and entry point.

- **Level Selection**

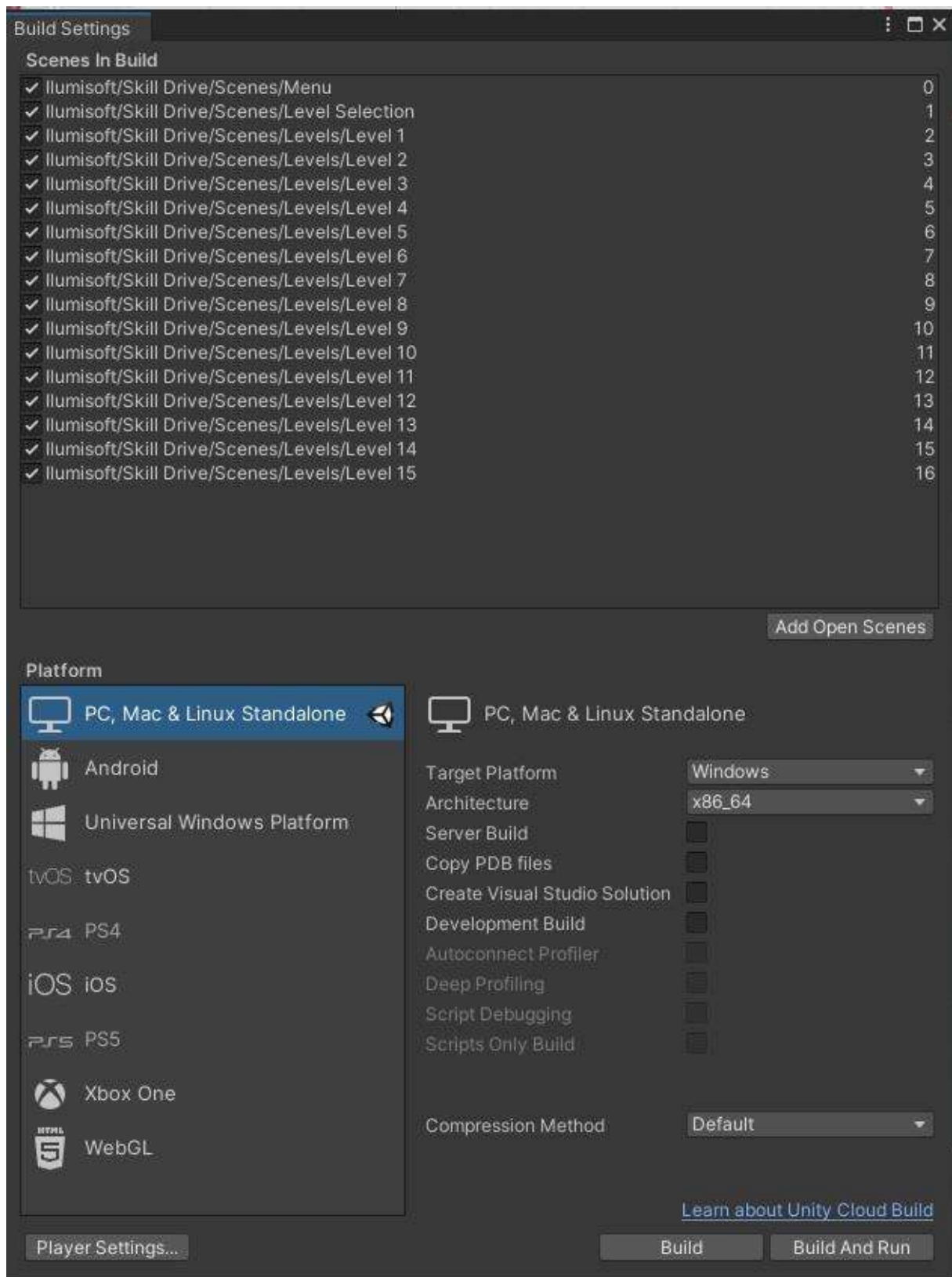
This scene contains a preview of each level and uses virtual cameras to animate the level selection.

- **Level 1-15**

Each level can be found in its own scene.

Build Settings

The following screenshot shows the Build Settings. The first and second slot are reserved for the main menu and the level selection. All following slots can be used for the game levels.

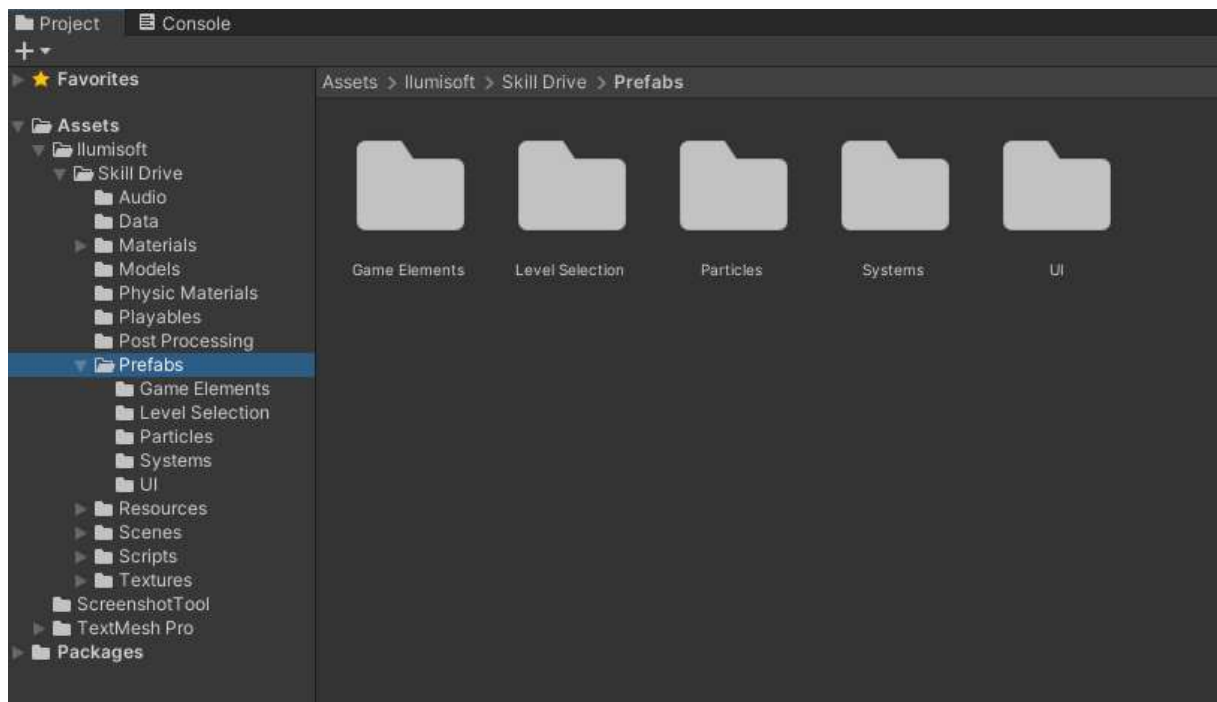


Prefabs

We created prefabs for the most important game elements, such the game systems, the player vehicle, the level previews, etc. This allows you to make adjustments to many aspects of the game without adjusting all scenes individually.

All prefabs can be found in the "**Assets/Illumisoft/Skill Drive/Prefabs/**" folder.

In the later chapters of this guide, you will be further introduced to the system prefabs, the game manager and the vehicle setup.

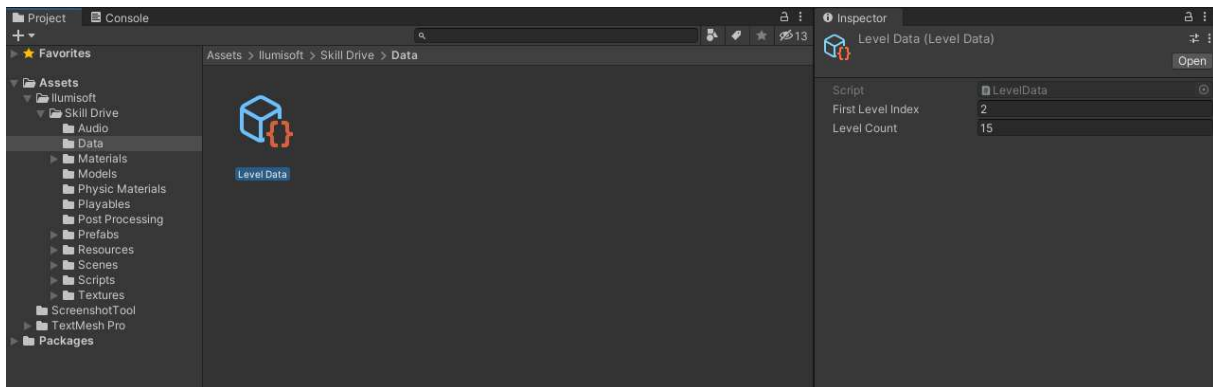


Level System

We use a predefined scene order scheme, in order to make it easy to detect the number of levels and the index of the currently loaded level via script.

Level Data

This is a simple scriptable object you can find in the "**Assets/Illumisoft/Skill Drive/Data/**" folder. In order to work properly the level system needs to know, which scene is the first level (by its build index) and how many levels the game contains. E.g., in our default setup level 1 has the build index 2 and the game has a total number of 15 levels. If you want to add a new level you just need to adjust the **Level Count** field.



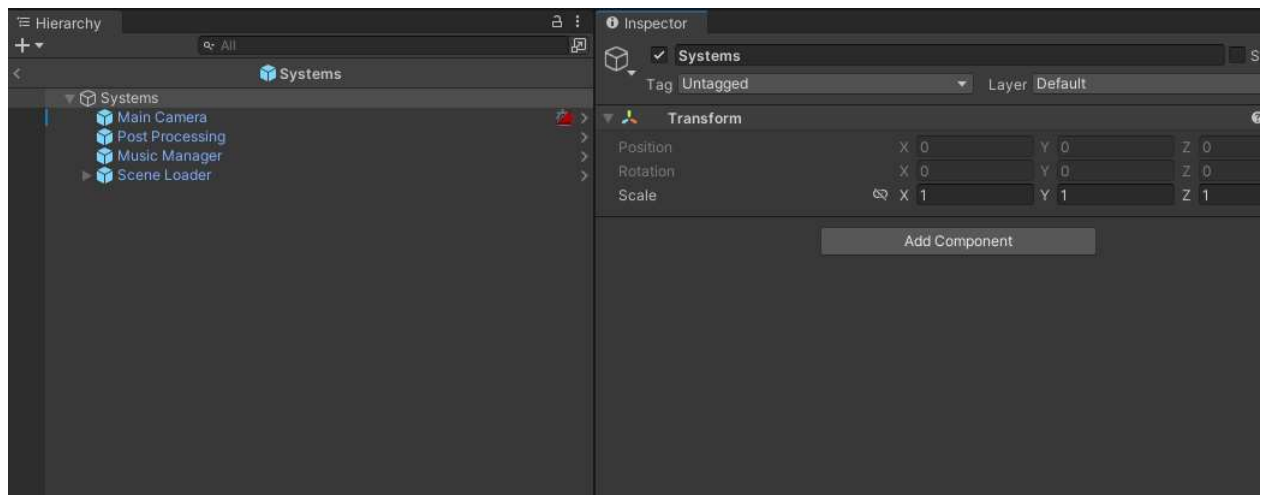
Creating new Levels

In order to create a new level, you need to

- **Create a new scene for your level.** The easiest way to go here is to duplicate another level and simply adjust the position of the player, the camera position and rotation as well as the environment and collectables as you wish. You can add of course any custom elements you want to your level.
- **Create a preview of the level for the level selection:** To create the previews, we created new prefabs, copied the environment and the collectables of the levels and scaled them down until the previews looked properly. These preview prefabs are then used in the level selection scene.
- **Add the scene to the build settings and adjust the level data:** Make sure, the new level is in the build settings scene list and adjust the **Level Count** field in the **Level Data**

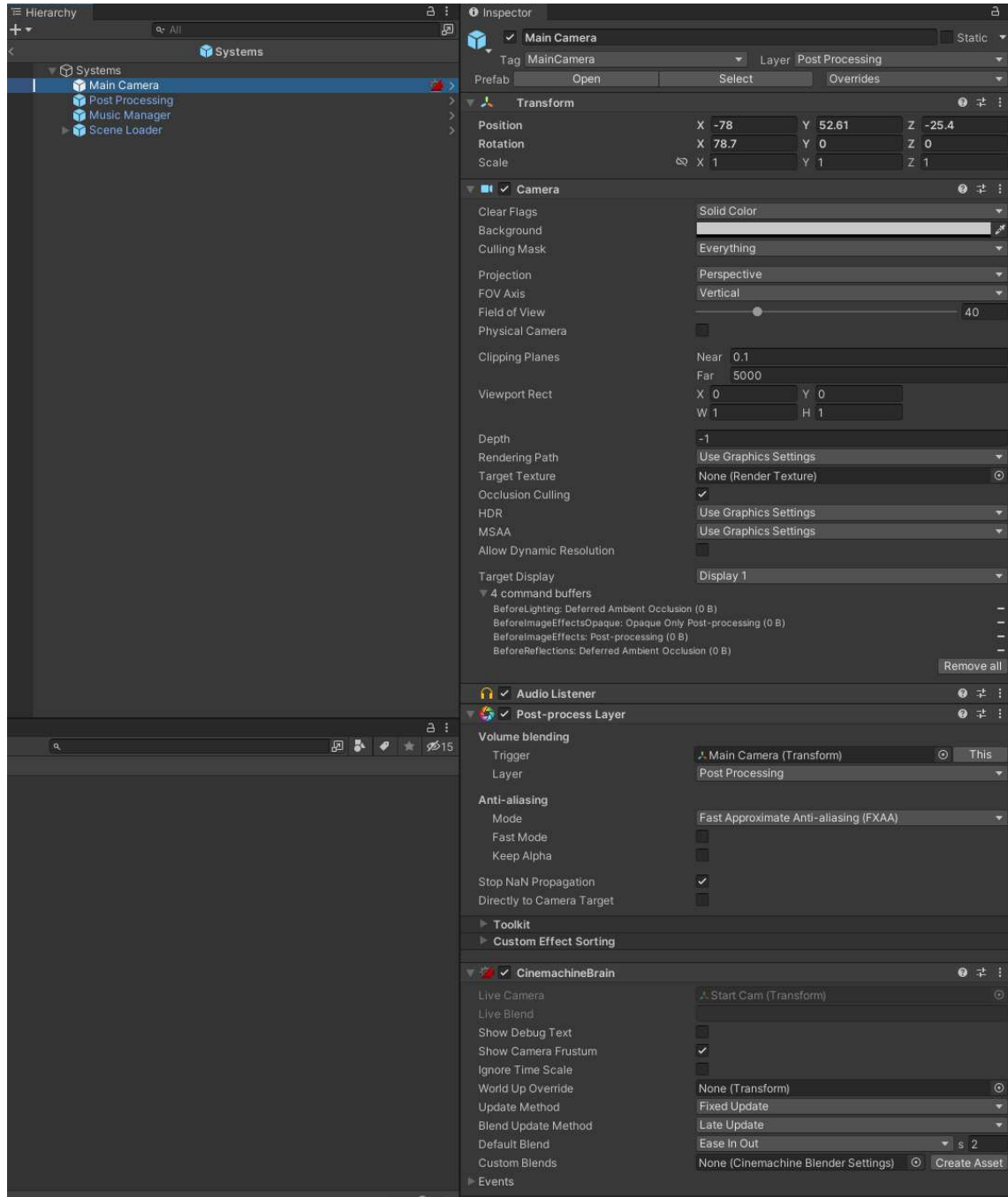
The Systems Prefab

To make it easy to make changes to the setup and the configuration of the game, we created a “**Systems**” prefab containing all the core game systems and elements that are used in each scene of the game, like the **camera setup**, **Post Processing** configuration, the **music manager** and the **scene loader**. You can find the prefab in the *Assets/Illumisoft/Skill Drive/Prefabs/Systems* folder.



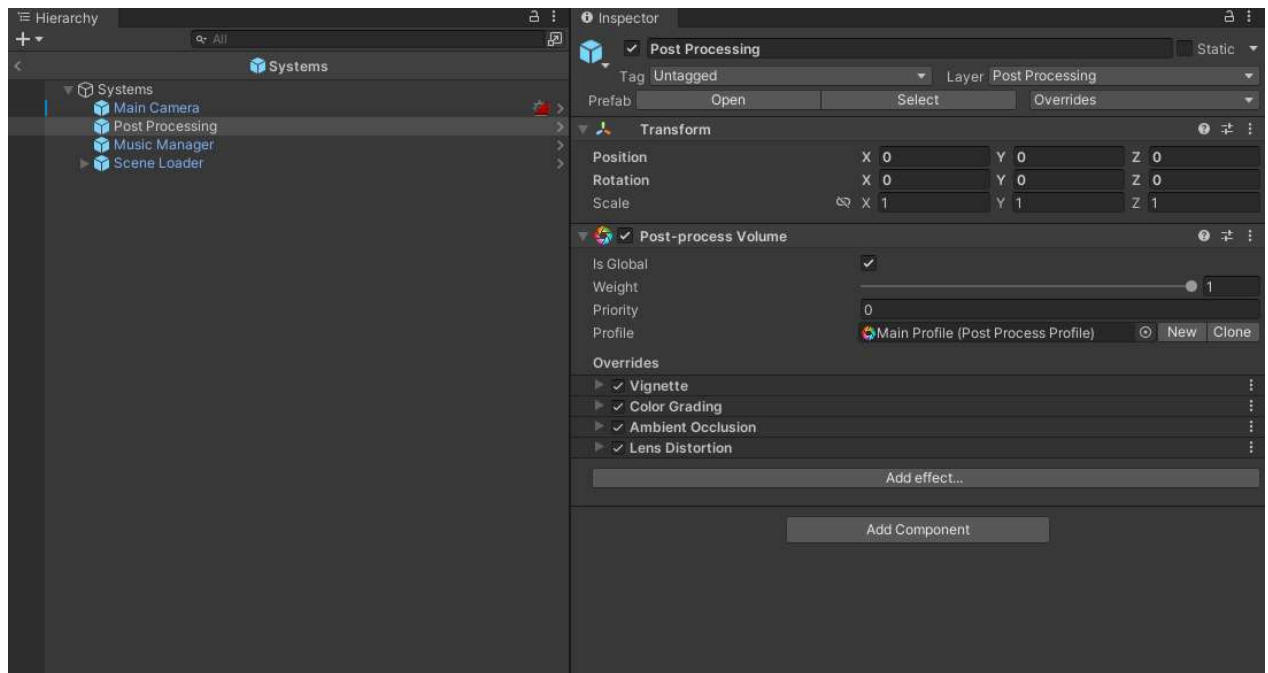
Main Camera

The **Main Camera** prefab, contains the main setup of the game camera, which includes the setup of a **Post-process Layer** with Anti-aliasing and a **Cinemachine Brain** component, which is required by the Virtual Cameras used with Cinemachine.



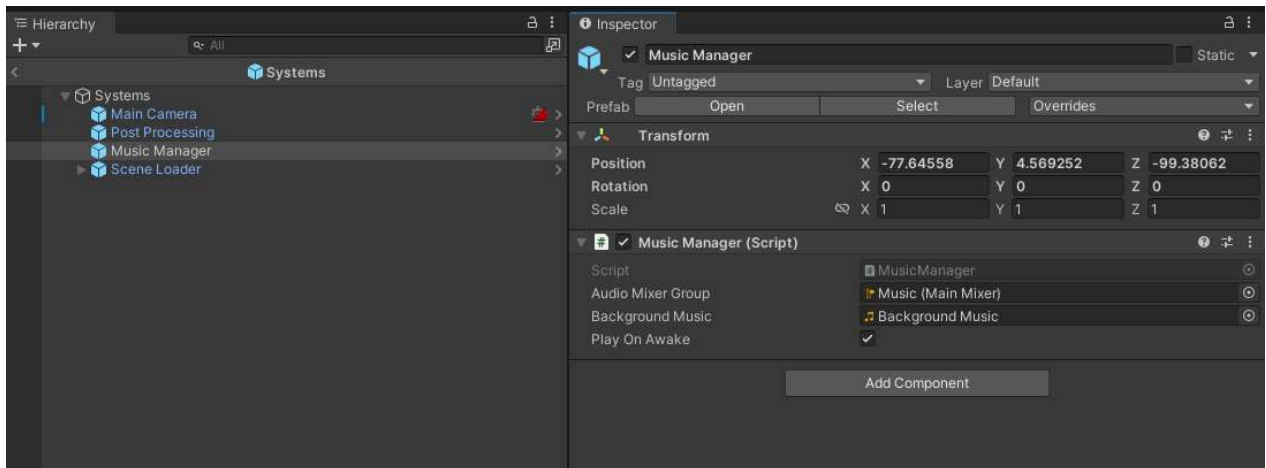
Post Processing

The **Post Processing** Prefab contains the global **Post-process Volume** used by the game. You can make adjustments here, if you want to change the color grading, AO or other effects. You can also create overwrite volumes, with a higher priority if you want to create Post Processing volumes that are only applied to a specific scene.



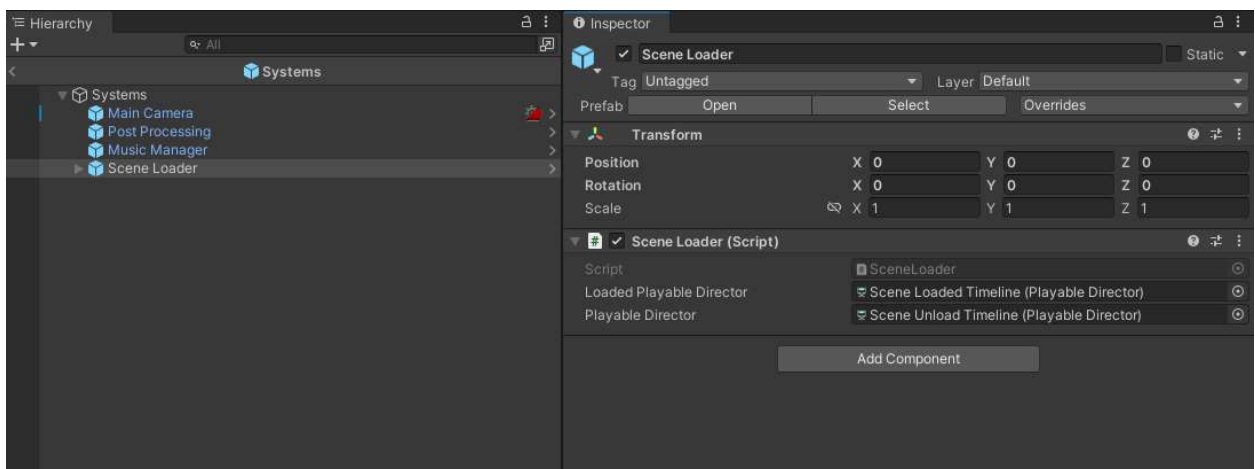
Music Manager

The **Music Manager** allows you to configure the background music of the game, which is played in the background and not being interrupted by scene loads. To allow this, it creates a persistent Music player game object, which is added to the **DontDestroyOnLoad** scene. In the music manager you can configure which song should be played, specify the Audio Mixer to be used and whether the music should be played automatically when the game is started.



Scene Loader

The Scene Loader is used to load the different scenes in the game. It smoothly fades a scene out, when loading a new one and also fades the new one in. To do so, it uses Timelines animating an overlay canvas, which sits on top of the screen. If you want to change the look of the fade in and fade out, you can simply create your own timeline animations and assign them in the scene loader via the inspector.

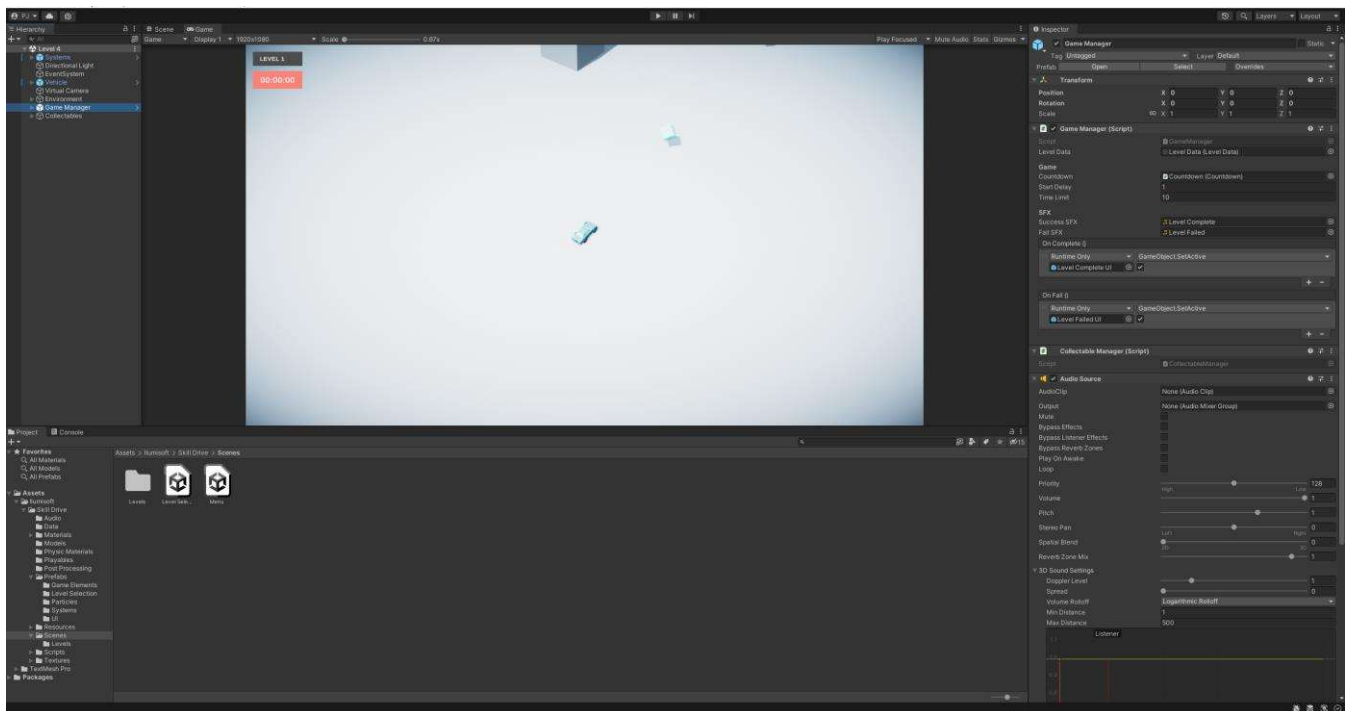


The Game Manager

The game manager controls the flow of the game when a level is loaded and the game rules. It automatically sets up the Countdown and ends the game when the player has collected all cubes or is out of time.

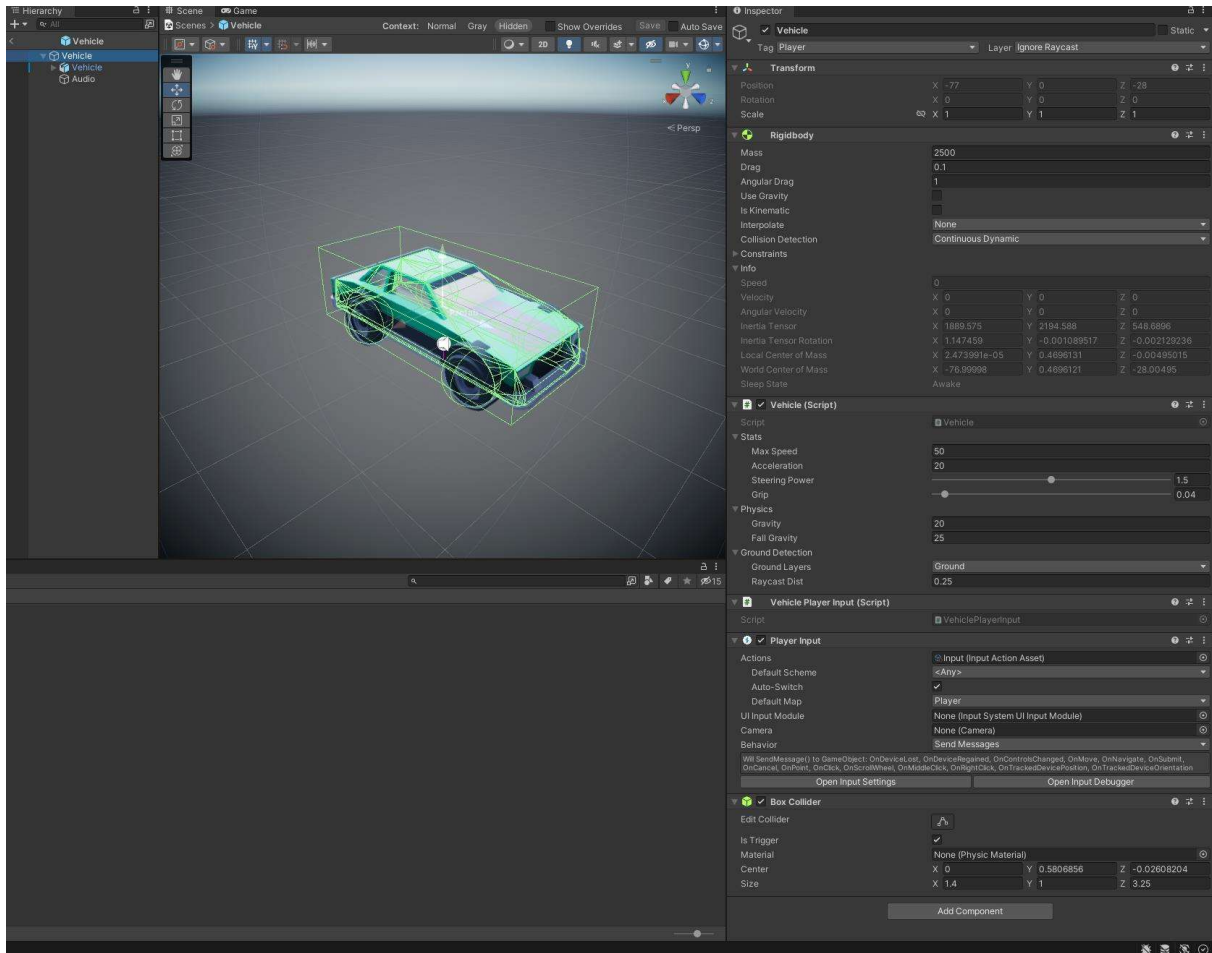
Adjusting the Start Delay and Time Limit

The game manager allows you to configure the **start delay** (so that the countdown does not start instantly when the scene is loaded), the **time limit**, the **sound effects** that should be played when the player succeeds or fails and which UI elements should be shown when the game is over.



The Vehicle

The Vehicle prefab (*Assets/Illumisoft/Skill Drive/Prefabs/Game Elements/Vehicle*) contains the player's car. It has a vehicle component, which handles the driving logic, like steering, acceleration and general movement of the player's vehicle.



Adjusting the Driving Mechanics and the Appearance

You can configure the **max speed, acceleration, steering power and grip** of the vehicle in the inspector of the **Vehicle component**. Here you can also adjust the **gravity** force and which layers should be used for the ground detection.

If you want to customize the appearance of the player's vehicle you can simply replace its model with your own. In that case make sure to add sphere colliders to each wheel of the car. Additionally, a Trail Renderer component has been added to the back wheels to create the car tracks while driving.

Support

Do you have a question or need help? Don't hesitate to get in touch with us via email!

Email: support@ilumisoft.de