

City Car Driving Simulator

Thank you for purchasing and using City Car Driving Simulator! This documentation will guide you on how to install the package and about overall informations.

City Car Driving Simulator	1
Installation	2
Project Settings	2
Welcome Window	3
Toolbar Menu	4
CCDS Settings	4
Player Vehicles	5
Scene Managers	5
Project Hierarchy and Locations of the Assets	7
Player Vehicles	7
Traffic Vehicles	7
Scriptable Objects (CCDS_Player Vehicles, CCDS_Settings, etc...)	7
AI Vehicles (Used in missions)	7
Mission Objects (Trailblazer obstacles, checkpoints, etc...)	7
Markers	7
UI	7
Cameras	8
Miscellaneous (Sirens, smoke, debris, etc...)	8
Necessary Components	8
Necessary Components For Player Vehicles	8
Necessary Components For Main Menu Scene	9
Necessary Components For Gameplay Scene	10
Adding And Customizing Modifications	11
Changing Wheels	11
Adding Wheels to Main Menu UI	12

Common Mistakes	12
Support.....	14

Installation

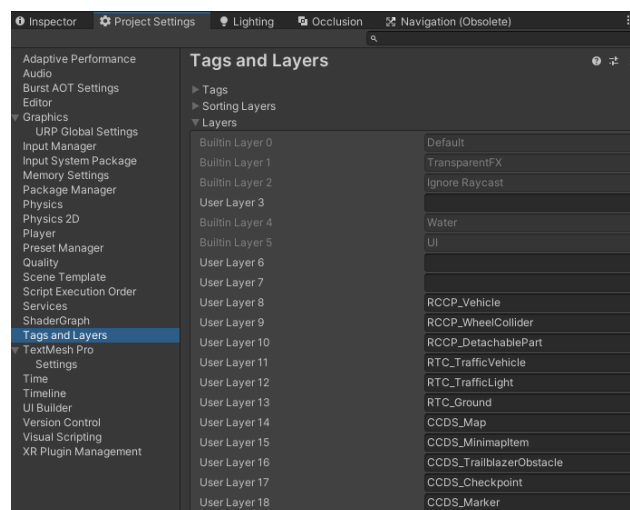
Project has been exported with **Unity 2022.3.11f1** and will work on same or newer versions. After the importing process, welcome window will. Check your console if you're having any errors. You can ignore yellow warning messages, they will appear only once. Yellow warning errors are related to collider polygon count, Unity already reduces them to work with it.

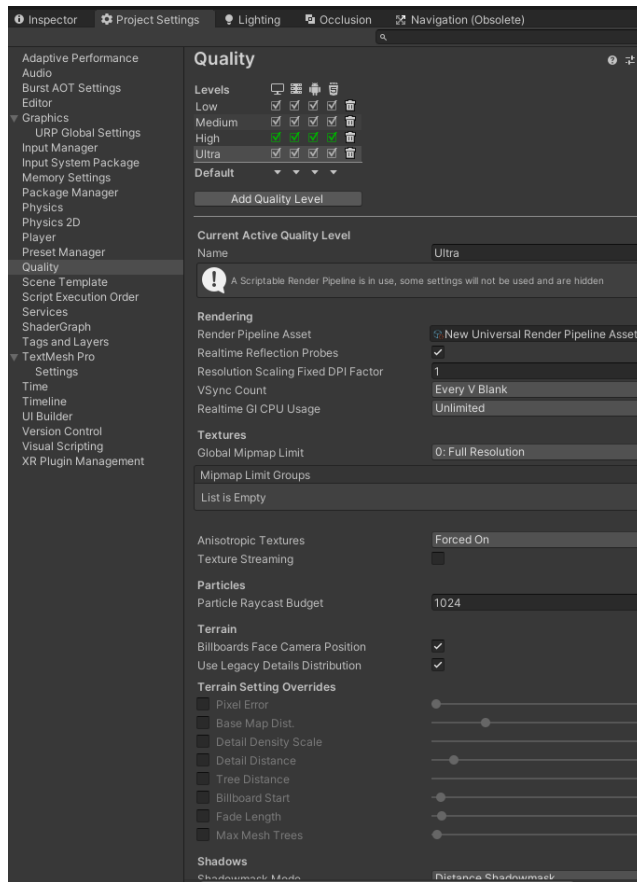
All project settings including tags, layers, inputs, player, and physics settings will be overridden, be sure to import the package into the new fresh project. If you attempt to import the package to any existing project, all project settings mentioned above will be overridden.

Please don't change directory of the project, otherwise some editor scripts won't work.

Project Settings

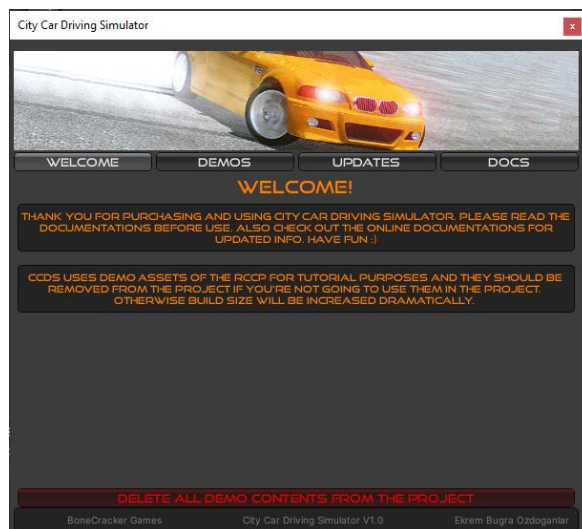
You might want to check those project settings after the import (**Edit → Project Settings**).





Welcome Window

CCDS comes with one main menu scene, and two gameplay scenes which can be found in the “**Scenes**” folder of CCDS. Also, you can access to them by welcome window of the CCDS (**Tools** → **BCG** → **CCDS** → **Welcome Window**).

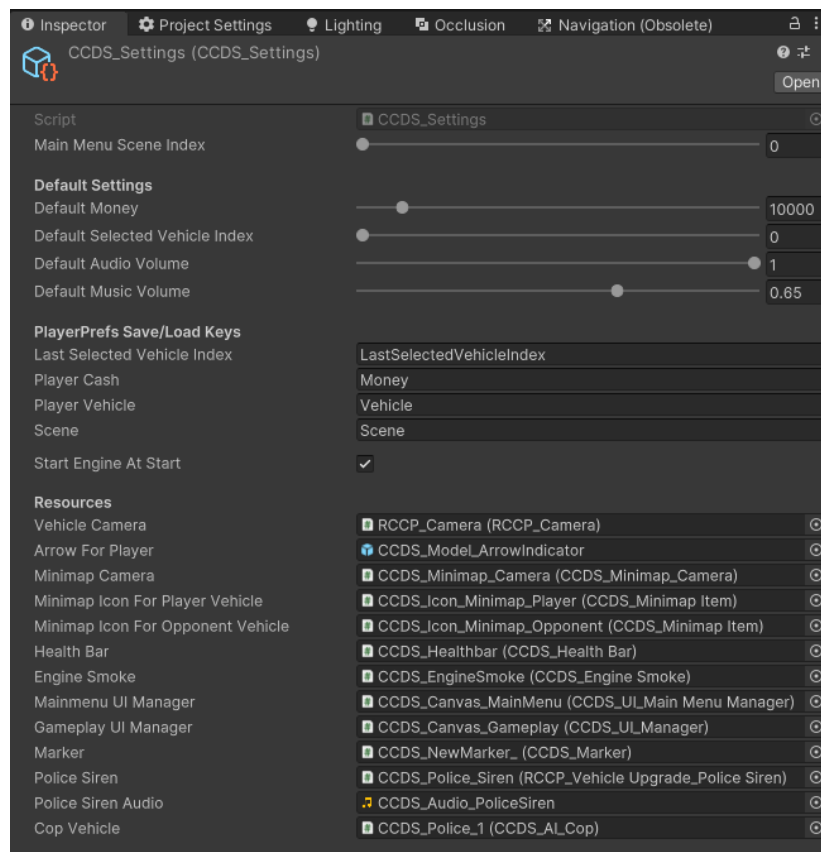


Toolbar Menu

Test the game without making any changes to be sure everything is working fine. After the test, you can reskin the project the way you like. You can access toolbar menu of the CCDS from the **Tools → BCG → CCDS**.

CCDS Settings

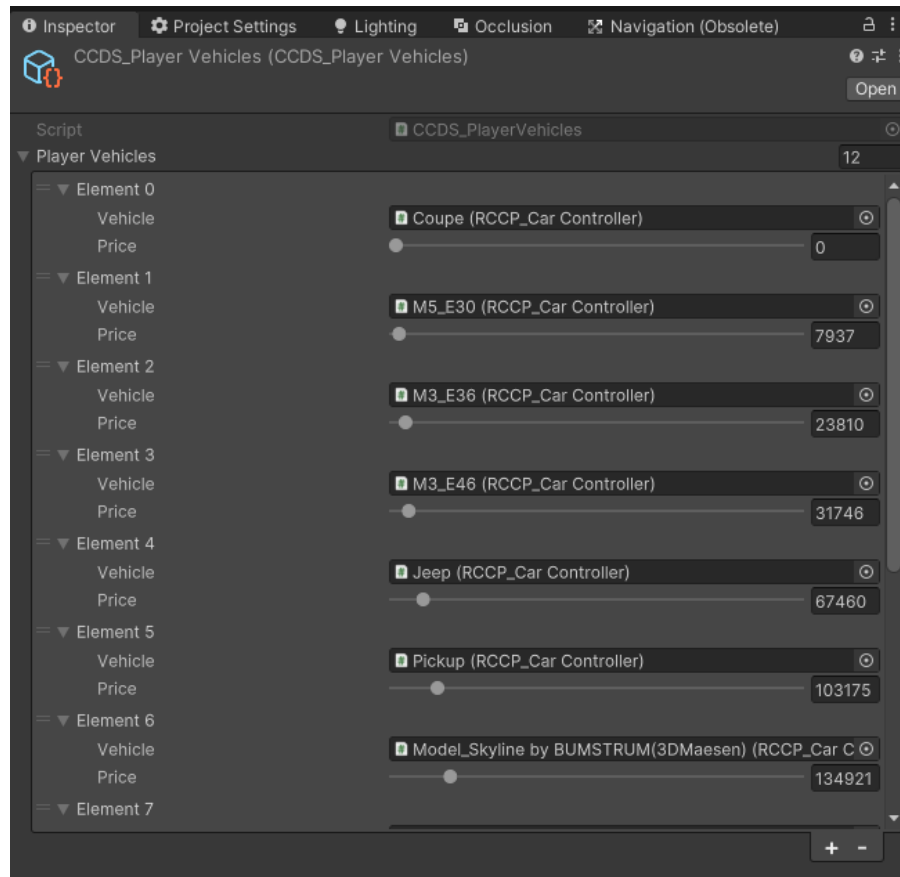
Here are the general shared settings of the CCDS called **CCDS_Settings** which can be accessed by the **Tools → BCG → CCDS → Edit Settings**.



Note: Hover your mouse cursor on any property you'll see in the inspector panel for detailed information about it. Instead of explaining them one by one in the documentation, I thought this would be a better way.

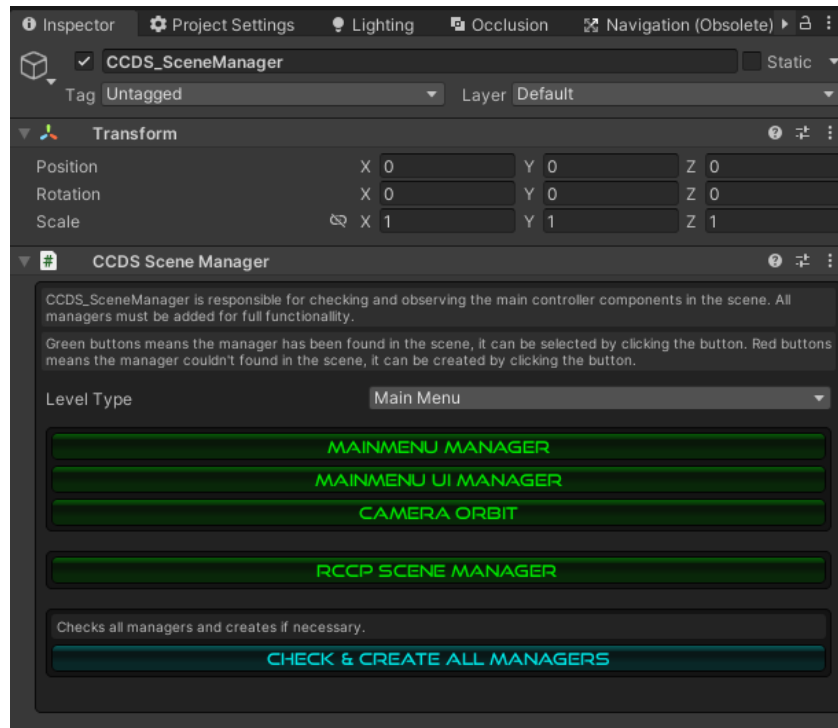
Player Vehicles

Here are the selectable and playable player vehicles called **CCDS_PlayerVehicles** which can be accessed by the **Tools → BCG → RCCP → Player Vehicles**. This topic has been explained very detailed in a separate documentation named “**How to Create / Configure / Add / Remove Player Vehicles**”.



Scene Managers

Scene managers are responsible for checking all necessary components in the scene, make sure everything is running ok. This main scene manager can be found in the CCDS scenes named **CCDS_SceneManager**. It has two options for main menu, and gameplay. It will check everything in the scene and warn you if anything is misconfigured. **CCDS_SceneManager** can be created by **Tools → BCG → CCDS → Create → Scene Managers → Main Menu / Gameplay → Scene Manager**.



Green buttons mean that manager is exist in the scene and selectable. Red buttons mean that manager does not exist in the scene, clicking it will create it. Please click “Check & Create All Managers” button to check all managers once.

Project Hierarchy and Locations of the Assets

Every asset has been categorized in separated folders. Location of player vehicles, scenes, prefabs, and resources have been explained below;

Player Vehicles

CCDS/Prefabs/Player Vehicles

Traffic Vehicles

CCDS/Prefabs/Traffic Vehicles

Scriptable Objects (CCDS_Player Vehicles, CCDS_Settings, etc...)

CCDS/Resources

AI Vehicles (Used in missions)

CCDS/Prefabs/AI Vehicles

Mission Objects (Trailblazer obstacles, checkpoints, etc...)

CCDS/Prefabs/Missions

Markers

CCDS/Prefabs/Markers

UI

CCDS/Prefabs/UI

Cameras

CCDS/Prefabs/Cameras

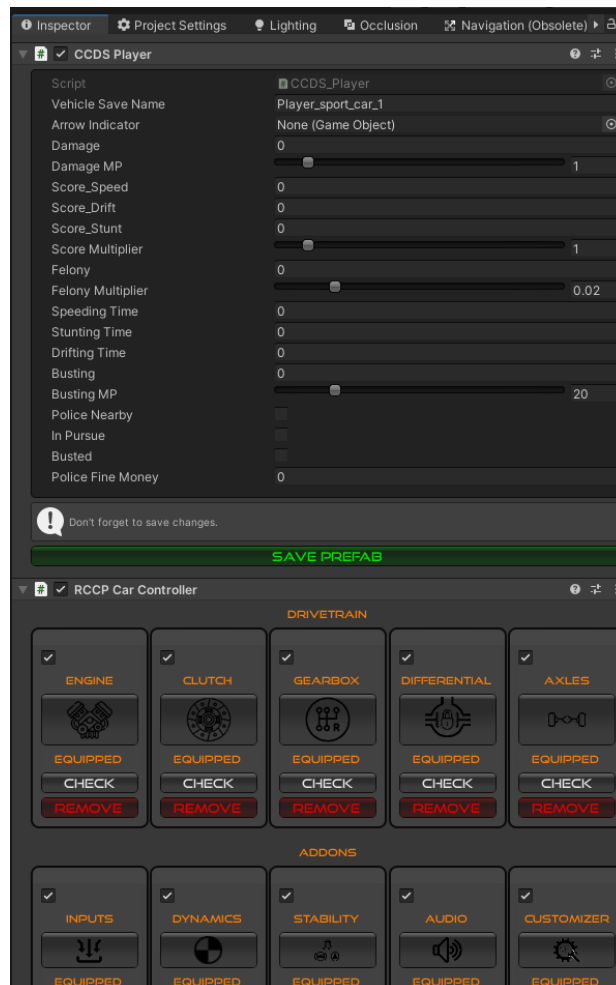
Miscellaneous (Sirens, smoke, debris, etc...)

CCDS/Prefabs/Misc

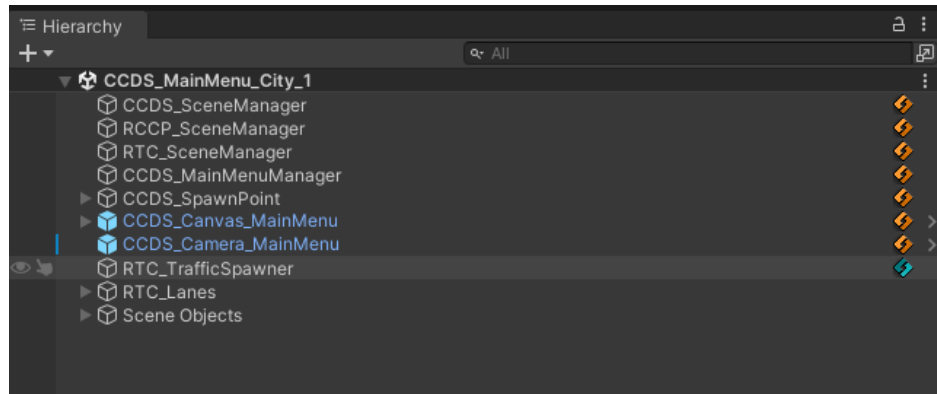
Necessary Components

Necessary Components For Player Vehicles

RCCP_CarController, CCDS_Player



Necessary Components For Main Menu Scene



CCDS_SceneManager (Main manager scene, all necessary systems can be created by this manager)

RCCP_SceneManager,

CCDS_MainMenuManager

CCDS_SpawnPoint

CCDS_Canvas_MainMenu

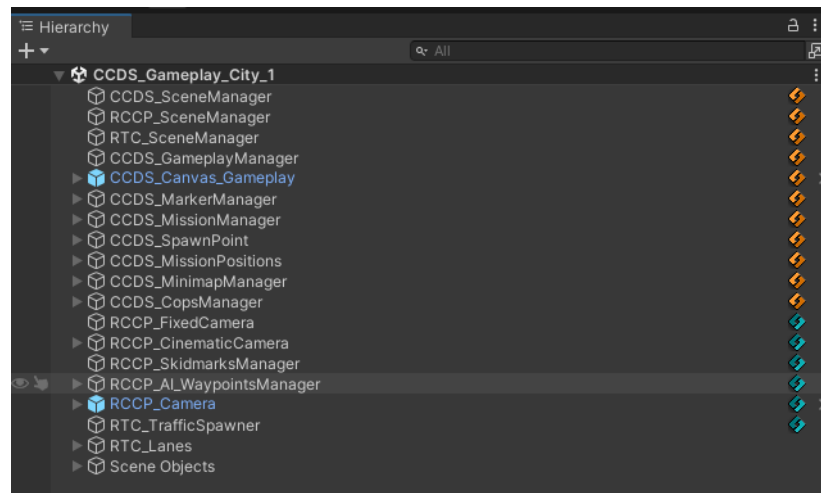
CCDS_Camera_MainMenu

RTC_SceneManager (Only for traffic system if you're going to use)

RTC_TrafficSpawner (Only for traffic system if you're going to use)

RTC_Lanes (Only for traffic system if you're going to use)

Necessary Components For Gameplay Scene



CCDS_SceneManager (Main manager scene, all necessary systems can be created by this manager)

RCCP_SceneManager,

CCDS_GameplayManager

CCDS_Canvas_Gameplay

CCDS_MarkerManager

CCDS_MissionManager

CCDS_SpawnPoint

CCDS_MissionPositions

CCDS_MinimapManager

CCDS_CopsManager

RCCP_Camera

RTC_SceneManager (Only for traffic system if you're going to use)

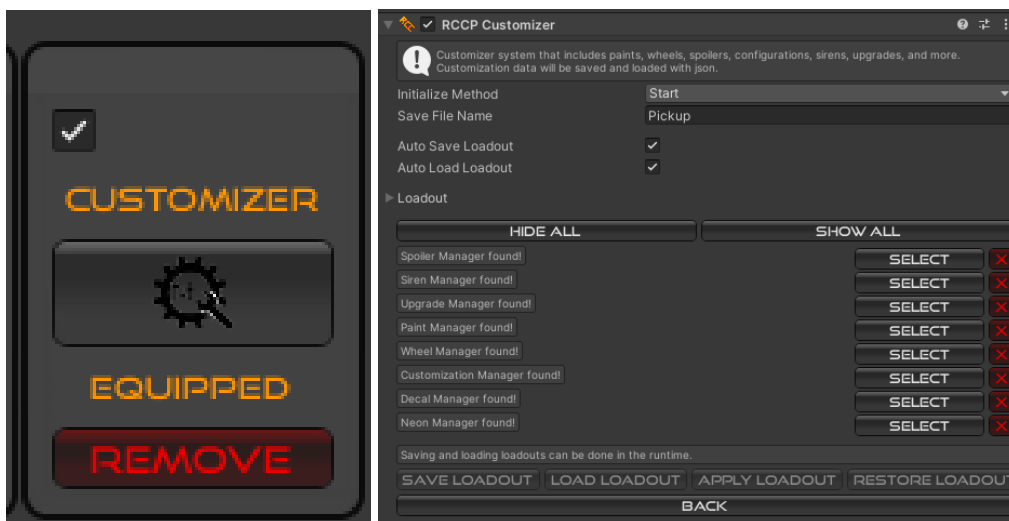
RTC_TrafficSpawner (Only for traffic system if you're going to use)

RTC_Lanes (Only for traffic system if you're going to use)

Adding And Customizing Modifications

All modification process are handled by [RCCP_Customizer](#) attached to the vehicle as an addon component. This script is managing all main modders, which is paints, decals, upgrades, wheels, neons, spoiler, and sirens. In order to create a modder, simply click create button and editor script will create and set it directly for you. For ex, your new vehicle is missing spoilers, so you have to click create button for spoilers. All you have to do is changing their locations.

For spoiler locations, you need to hold each spoiler and position it properly. The same thing is required for decals, their position is very important. For neon and siren, you need to position them. The working logic of the scripts is quite simple. They only make the gameobjects active or passive. They do not change their positions. That's why your positioning is very important and quite simple. Script is just enabling / disabling target gameobjects, that's all.



Changing Wheels

The location of all changable wheels is [CCDS/Prefabs/Wheels](#). The list of prefabs here is [CCDS/Resources/CCDS_SelectableWheels](#). All wheels are just gameobjects. You can first create the prefab of the wheel model you want to use in your game and then add it to the list.

Adding Wheels to Main Menu UI

All wheel UI buttons in the wheel selection panel (in the main menu) contain the “[RCCP_UI_Wheel](#)” script. Script has a target index value. All you have to do is enter the index value of which tire that button represents. For example, the wheel you added is in the 16th row of the list. The index number of your button should also be 15 (because 0 is the first index). The game will automatically save and activate that wheel in other scenes.

Note: If the wheel is facing the opposite direction, set the X scale of the prefab to a negative value.

Common Mistakes

Inability to use the vehicle properly;

Accurate physics cannot be achieved if your vehicle's RCCP structure is not set up properly. For example, the vehicle may pull too much to the right or left. Accelerating and braking may not smooth and accurate. You can take a look at the installation video of RCCP again.

Failure of the points system;

There is no [CCDS_Player](#) script attached to your vehicle. Be sure your vehicle prefab has this script attached.

No car spawn;

Your vehicle does not have [CCDS_Player](#) script. Be sure your vehicle prefab has this script attached. Or your scene doesn't have gameplay manager and other necessary managers.

Scene won't open / wrong scene opening;

Your scenes are not added in the [Build Settings](#). Be sure all your scenes have been added to the build list. Each scene button in the main menu UI canvas has a target build index value. If this index value is not equal to your scene index, wrong scene will be loaded.

New vehicles are not selectable;

They are not included in the [CCDS_PlayerVehicles](#) list. Make sure your prefabs have been added in the [CCDS_PlayerVehicles](#) (*Tools → BCG → CCDS → Player Vehicles*).

Changes I made to my vehicle don't take effect;

You must save every change you make to your prefab. When you make changes on your vehicles, they are not directly saved in the prefabs. After selecting your vehicle in the scene, you can save the prefab changes with the "[Override](#)" button at the top right of your Inspector panel. Or just click the "[Save Prefab](#)" button on your inspector panel while it's selected.

Note: You do not need to do this if you are editing the prefab directly in your project. For example, you double-clicked the vehicle prefab in **Prefabs/PlayerVehicles** and you are editing directly, no need to override, or save the prefab. Because you are directly editing the prefab, not just gameobject on your scene.

It gives an error when I select spoiler, siren, decal, or neon on the modification screen;

Your vehicle does not have proper installation for mods. Check out [RCCP_Customizer](#) attached to your vehicle, and check all main modders here.

Modifications (Decal, neon, spoiler, siren) don't work;

There are seven separate upgrade systems that must be added to a vehicle. These are decals, spoilers, paint, neon, sirens, wheels, and upgrades. Instead of adding them one by one, you can create them directly from the [RCCP_Customizer](#) attached to your vehicle. Simply click "[Create](#)" button to create corresponding modder. If a system does not exist, the modification won't work. For example, you don't want to apply spoiler modification, simply do not create it. At the main menu, UI button for spoiler customization will be disabled for this vehicle, because spoiler modification doesn't exist on this car. It won't throw an error if your car doesn't include any modification system.

I am getting SDK or NDK error for Android builds;

Make sure you have Android SDK and JAVA installed on your computer. First of all, try to create another blank project and build an APK to avoid wasting time. If you can get an APK in an empty project, you can also get it in this project as well. Usually the problems are that the Android SDK is not installed, or the JAVA is not installed. Unity asks for your permission to install the Android SDK during installation. You can do this by clicking yes. If not, you can open Unity Hub and add your Unity version from the Installs section by clicking the gear icon and adding it from [Add Modules](#).

I am getting GRADLE error on Android builds;

Remove 4 ticks from **Edit --> Preferences** and re-enable them again. These are the set of options for **SDK, Java, Gradle**. You may want to restart Unity after making this.

This documentation covers the general information about the installation and how the project works. Creating / editing / removing vehicles and scenes have been explained in separate documentation.

Support

Please include your invoice number while sending an email to me.

Contact: bonecrackergames@gmail.com