

MS Vehicle System

Created by: Marcos Ismael Schultz

Age: 20

Fórum: www.schultzgames.com

Email: marcos11-24@hotmail.com

This asset is a basic version of the active 'MS Vehicle System'.
Link: <https://www.assetstore.unity3d.com/en/#!/content/88035>

'MS Vehicle System free version' is a simplified version of the asset, and contains:

- Scene controller, allowing you to get out and get into vehicles in a simple way.
- Automatic gears system for the vehicle.
- Controller for cameras, including 4 different modes of handling.
- Complete torque distribution system for the vehicle, allowing precise adjustment.
- Brake system.
- Steering wheel system for the vehicle.
- Simple system of sounds for the vehicle, including brakes sounds, engine, beats.
- Simple system of skid marks.

'MS Vehicle System' also includes 2 sample scenes and includes 5 demonstration vehicles, which allows you to test the systems.

The asset also comes with pre-fabricated objects (prefabs), making it easy to use the system with objects already configured.

How to use:

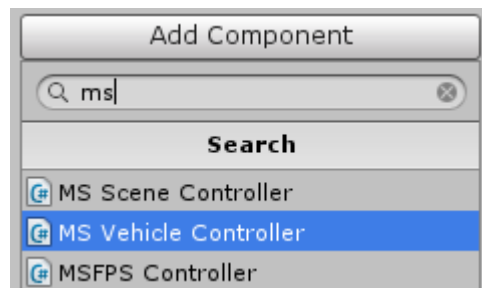
To use the system, you must first have a 3D model of a properly rotated and staggered vehicle, that is to say that the vehicle's rotation must be pointed towards the Z axis of the vehicle's main transform. Likewise, the wheels of the vehicle must also be rotated so that the Z axis of its transform is in the same direction as the Z axis of the main transform of the vehicle.

It is also good that the vehicle has scale 1 in all its meshes and transform. This implies in the main transform of the vehicle to have the scale (1,1,1), as well as all other transform of the vehicle, to the wheels ... This makes the simulation of physics more realistic.

Having the vehicle properly adjusted in rotation and scale, it is necessary to add a collider to the vehicle, to simulate physics and collisions. It is recommended that the collider cover the wheel rims to avoid surreal impacts. The image below illustrates the ideal position of the collider.



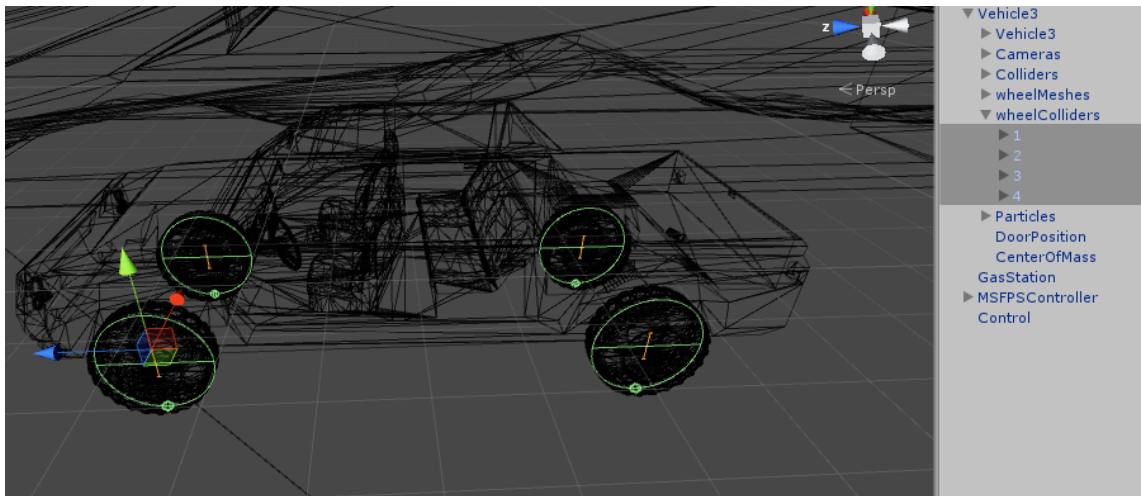
After that, the 'MS Vehicle Controller' component must be added to the vehicle main transform.



By doing this, the script will automatically add the 'Rigidbody' component to the main vehicle's transform, to simulate the physics of the vehicle.

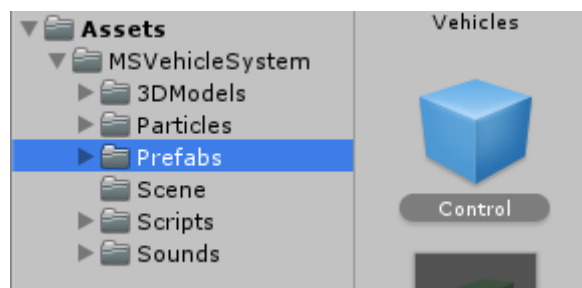


After doing all this, it's time to add the 'WheelCollider' component to each wheel of the vehicle by simply creating an empty object in the position of the center of the wheel, and adding to this object the 'WheelCollider'.



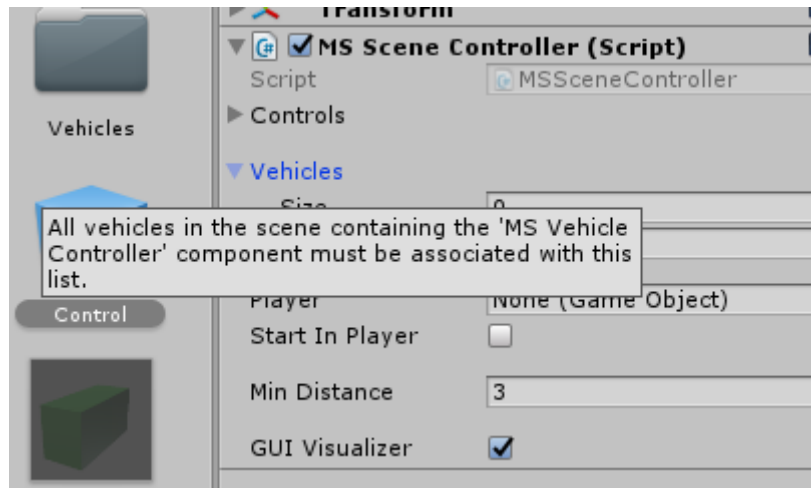
Now click on the main transform of the vehicle, go to the 'MS Vehicle Controller' component and configure the vehicle as you want, associating the wheels in their places, configuring the torques, among other things.

Having your vehicle set up, you also need to have a Controller for the scene, as it is responsible for informing the vehicle of the player's preferences. To do this, simply drag the prefab 'Control' to the scene. The prefab is in MSVehicleSystem> Prefabs.



After that, click the 'Control' that was added to the scene and configure it in the inspector, according to each variable.

The entire system has 'Tooltips' in the variables, making the system easy to implement, and helping to avoid errors. These Tooltips provide warnings that help the user to implement the system. To see the warnings, just rest your mouse over some variable, and the warning will appear.



Setting up the vehicle correctly as the 'Tooltips' indicate, and configuring the 'Control' object as their Tooltips indicate, the system should already work properly. If something is wrong, the system will provide alert warnings indicating what is wrong.

To help understand how the entire system works, the feature contains 2 demo scenes, with examples of how the system works. The scenes are:

Vehicles Arcade: A scene configured with 2 vehicles and an FPS player, allowing the player to enter and exit the vehicles, and test each one.

Vehicles Only: A scene with only vehicles, allowing to test each one.

Possible problems:

The wheelCollider component does not present a very realistic simulation when using actual values for the wheels, so it is advisable to leave each wheel with the mass at half the mass value of the vehicle. For example, if the vehicle has mass of 1000, each wheel must have a mass of 500, to give more realistic results.