**Final Project Dimitrije Prosevski, Matthew Jackson**

**Introduction:**

In this project we created a derby style 3D vehicle game. We developed the project using Blender for our models and Godot for our game engine. Our goals were to develop vehicles controlled through the keyboard and allow for as much realistic physics as possible. We also wanted to include sound effects and a health system for the vehicles.

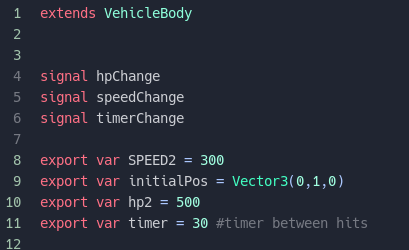
**Implementation:**

Our first step in creating this game was to import our models from Blender into Godot. This was made quite simple for us through the glTF export format within Blender. Once we had moved our models into Godot we had a couple immediate problems to fix. The models had no collision boxes attached to them and also needed a node structure for animation. We solved these problems by creating a node hierarchy rooted within a VehicleBody for each model. This allowed us to attach a collision box to each model and gave us some basic animation options in the node signals tab.

Our next step was to script accelerations and rotations to the wheels, and we did this within our Car1.gd and Car2.gd scripts. After doing this we realized for each model the wheels were not linked to any form of movement. We then added each wheel to its own VehicleWheel node, added traction and steering, and corrected the direction of movement to fit our models. The rest of our physics manipulation came through editing each VehicleBody’s weight property and the properties of each VehicleWheel under the wheel, suspension, and damping tabs. After completing this we then added the scripting needed to create a health system for the vehicles and allow for sound effects with collisions. The rest of our changes had to do with small things like lighting and finding audio files.

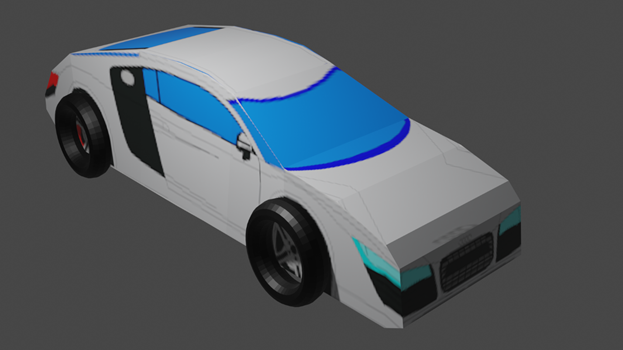
**Dimitrije’s edit of his part:**

My static model is a car (Audi). It has a body as one object, and 4 tires that are attached as children to the body. I built the body of the car using extrusion, rotation, scaling, and translation. Details were added using UV editing from the image of the car's front, top, back, and sides. I imported my car.blend into Godot and scaled, rotated the car to match Matthew’s car. Added the car body and all four wheels’ under appropriate nodes and saved it as a separate scene. Then, in the final scene I merged the car2.tscn into Final.tscn added scripts, signals and exported variables for the car.



All the collision detection is done in car2.gd script and its implemented in  This function is essentially a signal that triggers every time car1 body enters car2 body. Finally, I worked on splitting the screen, implementing the health system and the timer.

**Car2 Render:**



**Matthew’s edit of his part:**

In this project I created my car model (Car1) in blender through the use of UV mapping some images of a Buick Grand National to the extruded object body and wheels. After completing this I imported it into the godot project and added all of the basic elements (VehicleBody, collision box, etc…) to the object. I also added the functionality of the interpolated cameras to follow the vehicles. I then focused on adding realistic physics to the vehicles and driving capabilities to the cars. This was done in multiple scripts and mainly dealt with the environment. After adding physics I moved on to the scripting done for audio and allowing a reset of the game. This was done mainly in Final.gd.

**Car1 Render:**

****

**Results:**

We were able to hit most of our goals and the game engine renders two driveable vehicles with relatively good physics and collision effects. They both have a health system and audio effects. The collision boxes work decent considering they are simple rectangular boxes.

Here is a list of bullet points containing our final results:

* Physics environment
* Effective collision detection
* 3 screen splits (car1, minimap, car2)
* Health reduction on collision
* Game rules (players have to touch every 30 sec)
* Speedometer
* Controls (car1: WASD, car2: arrows)
* Signals for health change, timer change and hp change
* Audio effects for background and collision

**Conclusion:**

In this project we learned a lot regarding what it takes to build a 3D project within the Godot engine. We learned about some of the basic requirements needed for an object to simulate a vehicle. We learned how certain nodes in Godot work, like VehicleBody and VehicleWheel. And we were able to use them effectively in our project.

In the future we hope to improve our project by adding more scripting regarding the physics of our game. We also hope to create a better collision box for each of the models. And create more advanced tracks/arenas for the vehicles.