Work journal:

Every update in the work journal took me approximately **8-12 hours** to complete. Each task required careful implementation and testing to ensure smooth functionality and optimal performance. From creating the basic mechanics of car movement and jump methods to adding visual effects, obstacles, shooting mechanics, and enhancing the UI, every step demanded attention to detail and rigorous development. Despite the time investment, these updates were essential in refining the overall gameplay experience and delivering a polished final product.

Update 1:

Create the Basic mechanic of the car movement and jump method using the wheel collider, which is built in unity.

create the car physics component - the rigid body component that gives the object physical movement around the environment that is affected by gravity.

create the camera to follow a script to follow that follows the car in the x, and y-axis offset from the car and stays on the same z-axis.

create a basic terrain and make a layer called "Ground" to make the car know to recognize where the ground surface is to avoid the car to unlimited jump in the air by using Raycast.

Update 2:

limit the car rotation to 45 and -45 in y-axis rotation Euler angles to make the car keep moving forward across the road.

auto rotates the car toward the direction of the road if not pressing anything or rich the boundaries on the sides.

add pointed lights to the car, pointed lights, and directional lights to the environment for better visual.

using particle system components to emit particles from the exhaust in relation to the speed of the car.

using the Trail Renderer component to see where the car is crossing.

add boundaries by clamping the z rotation of the car to avoid the player getting out from the road.

Update 3:

add obstacles to the road, I created a wall script that detects when the car collides with a wall. This is done by utilizing the OnTriggerEnter event, which is triggered when the car has the tag "Player". When the event is triggered, a method is executed to destroy the wall. This is achieved by using a dynamic Rigidbody component that detects the "MeshRenderer" components within the wall object and adds a Rigidbody to any mesh contained within the object, causing the wall to explode.

Adding the machine gun on the car (it didn't work yet) .

Additionally, I added post-processing effects to enhance the visual scene. Moreover, a Particle System component was included to visualize the dead zone area.

Update 4:

I create a gun shooting mechanic script, using Transform.translate to move the gun in the desired direction.

The shooting action should be triggered by pressing the "F" key.

Instantiate a bullet every 0.5 seconds. To visualize the bullet's trajectory, add a Trail Renderer component to it.

When the car enters the Dead zone or collides with the walls activate the onTriggerEnter event to destroy the car.

Update 5:

First, I set the mechanic of the camera to top-down perspective.

Also, I Created a shooter that fires balls from the air by using the Rigidbody component. The shooter detects the player using Physics.OverlapSphere and the layer associated with the player. It aims directly at the player with a random ratio between the minimum and maximum values.

In Addition, create a shield box that activates a shield on the car when the car collects the box using the OnTriggerEnter event to detect the player. When the shield is activated, the Boolean value inside the car script changes. The shield remains active for a random duration defined within the shield Box script, and it is deactivated using the Invoke built-in method in Unity. Set the shield within the obstacles script to protect the player from any obstacles in the scene.

Last update:

Final Touches on the UI: Implement the main menu script to incorporate the methods into the UI buttons. Additionally, create two more levels within the scene, each featuring different roads and crossings. Moreover, enhance the walls by introducing a health system, requiring more than one bullet to destroy them.