

2015

CAR RACE GAME



SUBMITTED BY:

SWETHA SREE,
VANITHA NIMMAREDDY,
MAMATHA EMIREDDY,
SRIKAR REDDY KATTA

UNDER THE GUIDANCE OF:

DR. SHARAD SHARMA
ASSOCIATE PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE
BOWIE STATE UNIVERSITY

INTRODUCTION

For this Car Racing Game, we would like to accomplish a video game imitating the existing games. The race track environment has a sky and terrain with race track on it. User will have option to play either in night mode or in day mode. We have six checkpoints and three laps set in the game. Basically it is six checkpoints in every lap and once player finishes the third lap the game comes to an end. In the game we will have player car and other cars moving. Other cars are the opponent cars which are designed using artificial intelligence. There are certain way points set for the opponent cars to travel along the race track. Colliders are set to the cars so that they behave like actual cars when collision takes place. Once the game finishes it pops up a different screen asking if user wants to play again.

GAME THEME

- The car game has a race track and set of cars moving in loop in a predefined path.
- It has a player car which can be controlled by the user.
- There is a timer which calculates the time taken by the player car to complete the game.
- A set of checkpoints placed around the track.
- When the player car passes these checkpoints the checkpoint value gets incremented.
When the checkpoint value reaches 5, the lap value increments.
- The game stops when the car completes 3 laps in the race track.

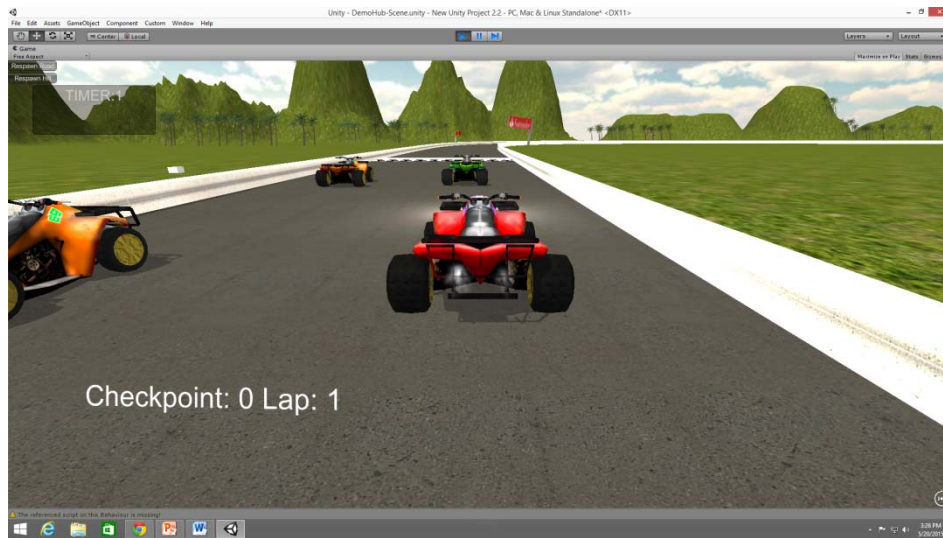


Figure 1: Day mode

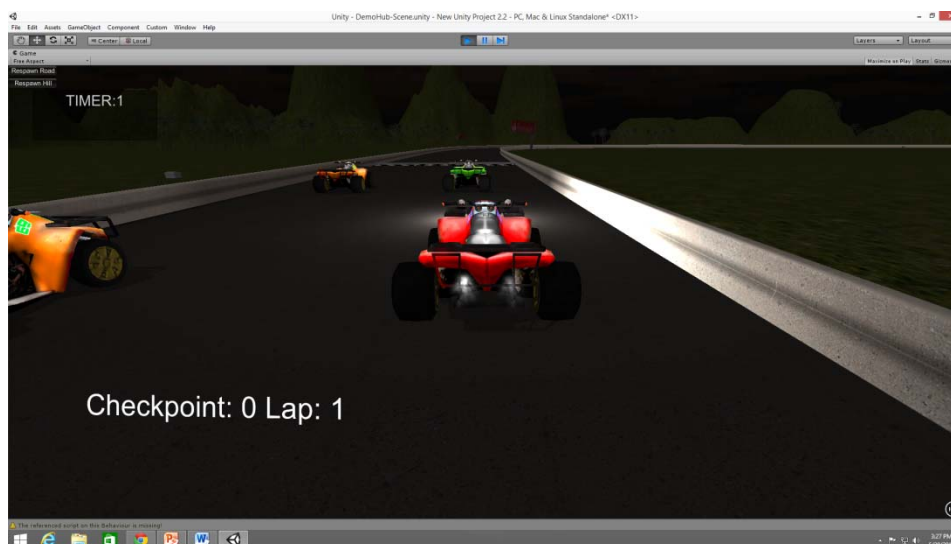


Figure 2: Night mode

SYSTEM REQUIREMENTS

➤ Desktop:

- o OS: Windows XP+, Mac OS X 10.7+, Ubuntu 10.10+, SteamOS+
- o Graphics card: DX9 (shader model 2.0) capabilities; generally everything made since 2004 should work.

- o CPU: SSE2 instruction set support.
- o Web player supports IE, Chrome, Firefox, Safari and others.
- iOS: requires iOS 6.0 or later.
- Android: OS 2.3.1 or later; ARMv7 (Cortex) CPU or Atom CPU; OpenGL ES 2.0 or later.
- Blackberry: OS 10 or later.

MODELING

- The Car racing game environment has a racing track with laps, check points, trees, flags and lights around the stadium. It also includes landscaping elements like ground and sky.
- We used sketchup tool for modeling the race track.
- We imported flags and Santander boards from Sketchup.
- Car models and avatars are downloaded from the asset store of unity.

FUNCTIONALITY

The game includes following functionalities

LIGHTS:

Lights are an essential part of every scene. Lights define the color and mood of our 3D environment.

Spot light:

A Spot Light has a specified location and range over which the light falls off. However, the spot light is constrained to a angle, resulting in a cone-shaped region of illumination. The center of the cone points in the forward (Z) direction of the light object. Spot lights are generally used for artificial light sources such as flashlights, car headlights and searchlights.

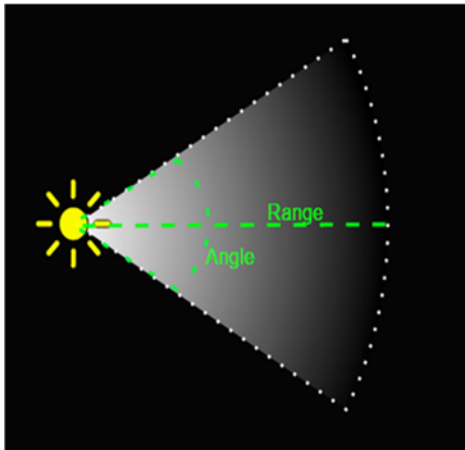


Figure 3: spot light



We used these spot light in Night Mode Scene, where the environment has a setup of night feel(dark). Spot lights helped us to provide the glowing head light to the car.

TIMER:

Timer measures the time taken by player car to cover 3 complete laps.



Figure 4:Timer

KEYBOARD FUNCTIONALITIES:

Key board functionality is used to move player car in the race track. Cars starts moving upon pressing W, A, S, D keys or by using arrows. And spacebar is used for applying brake.

COLLISION DETECTORS:

Box colliders are used in order to prevent the car moving out of the track.

AUDIO FILES:

To give effect of car motion, we includes sounds for moving car, sound when break is applied, and a hit sound when a car collides with the track.

TEXTURES:

Texture is added to the surface of a terrain to provide fine details. Every object in the game is given unique texture to make it look as realistic as possible.

SKY:

We added two different sky's to two different play modes. A day sky to the day mode and starry night sky to the night mode to differentiate between day and night.

SENSORS:

we added sensors to detect the checkpoints by the player car. As the wheel colliders of the player car touches the checkpoint it senses and increases the checkpoint value, which we used to calculate the laps travelled by the car.



Figure 5: check point and lap system

The above calculated lap count is used as a key to stop the car game.

AVATARS:

we includes avatars around the track to give a feel of audience.



Figure 6: Avatars around the track

INTERACTIVITY:

we used the following set of user triggered events

- ✓ selecting the play mode.
- ✓ Moving the car around track.
- ✓ applying brake

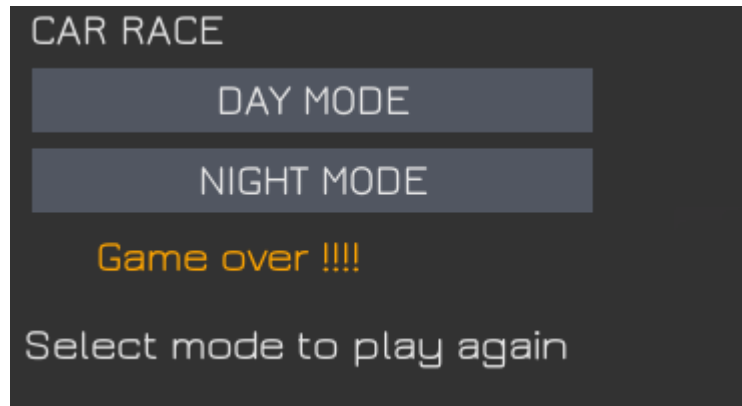


Figure 7: "Game Over" Screen

FUTURE DEVELOPMENTS

- Make the game a multiplayer.
- Track each player timer details and print the best winner and place of the player by comparing the scores.