



Horizon Chase Turbo

Tokyo Throttle

Racing Simulation Game

Jarl Ramos | Moises Fuentes | Jesus Quinonez | Irene Chavez |
Spencer Denney

CMPS 3350 | Fall 2022

Overview:

In a distant city, the blur of neon lights and the sound of summer cicadas are a mesmerizing experience. The blinding billboards, the serene mountains. A clash of harmony and disharmony. There is no better representation of this than on the streets of Tokyo. Experience quality racing like that of *Need For Speed* and *Forza Horizon*. Rev up, punch the gas, and shred the track in **Tokyo Throttle** – a racing game with traffic to avoid, cars to race and cities to experience behind the wheel of your car.

Technical Requirements:

- Programmed in C++
- Framework (Cars as a starting point)
- OpenGL for graphics

Features:

- **Game Mechanics:**
 - Main menu that appears before game starts; press “S” to start
 - The player will drive a car through a racetrack
 - Racetrack will be populated by other cars
 - Win Conditions:
 - The player must drive while avoiding obstacles in order to obtain the highest score
 - The game is over when the player crashes into another car
 - Game Over screen will display when player crashes
 - High score board will display the highest scores
- **Controls & Perspective:**
 - Control the car using standard “WASD” movement
 - Press ‘Q’ to quit the game
 - Third-person perspective
- **Graphical & Environmental Design:**
 - The game will feature 3D racing environments
 - The car will be designed in Blender
 - The car model will feature spinning wheels
 - Riveting soundtrack and ambiance that will really make your engines rev (drift phonk, eurobeats, etc.)
 - Skylines of cities like Tokyo, Seoul, and Shanghai
 - The racetrack will feature other cars
 - Collision detection through usage of hitboxes

Overall Goals:

- Include a third person view of a custom car
- Implement win and game over conditions
- Include more realistic car physics
 - i.e. add a speed limit to where car won’t go faster than a specified velocity during a certain time interval
- Have obstacles spawn randomly on the incoming track
- Have actual track maps (2 - 3 at most) each with different music and locations. (Seoul, Shanghai, Tokyo; .png backgrounds will reflect location of track).