Bilkent University

Department of Computer Engineering

# CS319

Object Oriented Software Engineering

# Final Report

OtoParker

**Group 7 - Vana Members:**

Arda Usman

Ali Çetin

Çelik Köseoğlu

Hüseyin Beyan

# Table of Contents

# 1. Changes In the Implementation

A. Game Management Subsystem

Game Manager class is replaced with Map Manager, and included all object instances are placed in this class.

B. Car Physics Class

This class added to calculate all car movements (position, direction, velocity, acceleration) in vectorial form.

C. Additional Features

The car can drift if the player wants to turn with an excessive speed.

D. Game Objects

Trash can are excluded in the implementation, since there were enough numbers of obstacles in other types. Also all obstacles are gathered in one class, to instantiate them one constructor is used.

E. Collisions

Collisions are handled in in Car class methods, not in a separate CollisionManager Class.

# 2. Status of the Project

We have successfully completed OtoParker as we mentioned in analysis and design reports. Also, managed to add the ability to drift the car for immense amounts of fun.

# 3. User's Guide

## 3.1.System Requirements

OtoParker is a game which implemented in java. For this reason, Java Run Environment(JRE) must be installed before playing game. You can download it from http://www.oracle.com/technetwork/java/javase/downloads/

Minimum System Requirements:

- Windows XP or Mac OS X Snow Leopard
- Pentium2 233 MHz CPU or higher.
- 256 MB of RAM or higher
- Screen resolution: 800x600
- Recommended system requirements:
- Windows 10 or macOS Sierra
- Intel i5 2 GHz CPU or higher
- 1 GB of RAM or higher.
- Screen Resolution: 800x600

## 3.2.Installation

Download and run otoparker.jar. Also you can run compile and run our code with a Java IDE.

## 3.3.Overview of the Game

When player chooses play option on main menu the game starts. There is a parking area on screen. There are parking lots, our car, stars, some obstacles like trees, other cars, traffic cones. Player can control our car and aim is parking this car to marked parking lot without crashing. If player's car hits an obstacle game will be over with failure and game over screen will be opened. Also obstacles can be destroyed with car weapon when player uses weapon of car.

Player can pick up Stars which distributed on various places on parking area. These stars can be used for upgrading car. Player can change model of car, colour, weapon and turning radius on upgrade car menu.

## 3.4.Game Objects Management Subsystem

### 3.4.1.Objects

**Car:** The car is our main car which should be parked.

**Bullet:** When player uses weapon of car there will be a bullet.

**Tree:** Tree is an obstacle. When car hit to the tree game over.

**Traffic cone:** Traffic cone is an obstacle.

**Parking area:** Parking area is an object which represent marked parking lot.

### 3.4.2.Controls

**Up Arrow:** Accelerate

**Down Arrow:** Back

**Left Arrow:** turn left

**Right Arrow:** Turn right

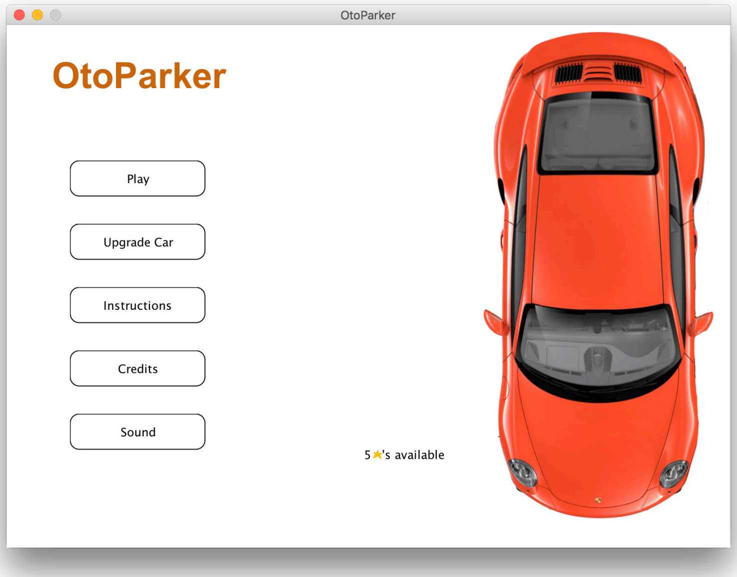
**B:** Brake

**N:** Boost (Nitro, Turbo, it flies (literally))

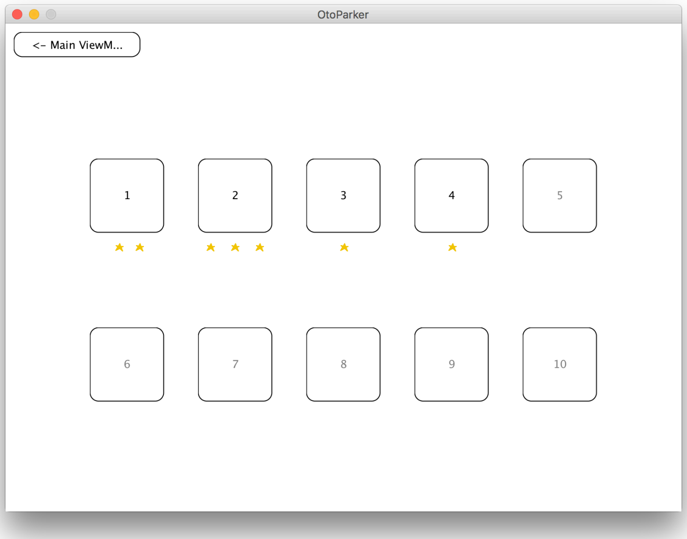
**Space:** Fire

## 3.4.3.Game Screenshots & Menus

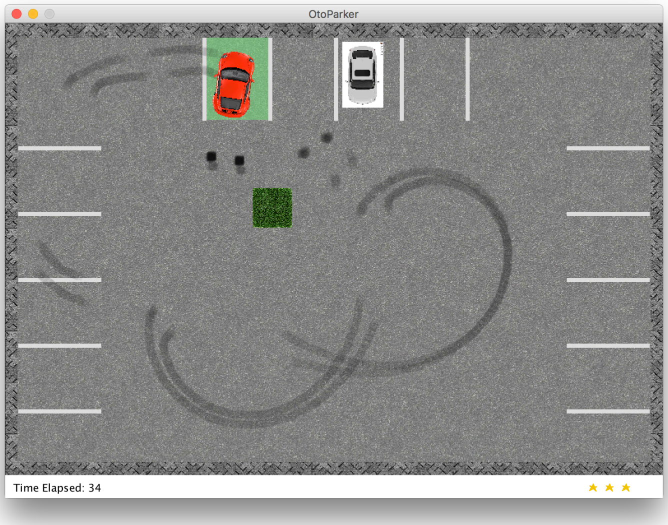
**Main Menu:** Opening screen is main menu and it contains 6 options: Play, instructions, credits, enable sound.



**Choose Level:** Player can choose level on this screen.



**Play:** This game screen and player will play game on this screen.



**Pause Menu:** When player press the pause option it appears.

**Game Over Menu:** Pops up when the player collides

**Credits:** Player can see developers of game.

**Instructions:** If player wants to learn how to play the game he or she can open this page and see instructions.