KARABUK UNIVERSITY
FACULTY OF ENGINEERING

A
PROJECT
REPORT
ON

"CARBUMP! GAME"

Submitted by

Özkan İMRE
Rüveyda SOLMAZ

Under the support and guidance of

Asst.Prof.Dr. Nesrin AYDIN ATASOY
Department of CE

Department of Computer Engineering
Karabük University, Engineering Faculty, 78050 KARABUK / TURKEY

KARABÜK-2020

Table of Contents

Introduction

A game is much more meaningful than software. Various contents should be provided to make it enjoyable. Therefore, the characters, location and design of the game are significant. The more the player is willing to play, the higher the quality of the game. As a result, this project tried to show that it is a fun game, although it has a simple design and gameplay.

The vehicles have been specially selected to be compatible with each other. The driveway was built as a double lane. The driveway was tried to be likened to the real asphalt road. Sound has been added to indicate an accident when the vehicles are hit.

1.1 Purpose

It should appeal to all ages for many players to play. Therefore, it is aimed at players of all ages to play a simple car game. On the other hand, it is expected that the player will want to play more games as loses.

1.2 Gameplay

A simple 2D game based on a basic design was designed. There are 8 types of cars in the game. One of them is called "player". This is the car the player will use. The player will control the car with the left and right keys. This game aims to make the highest score without hitting cars. The score increases as the player do not hit vehicles. As the scoreboard increases, the speed in the player's car increases. The reason for this is to make the game hard. The moment it hits the vehicles, the game is over. The highest score table is no longer zero. The highest score the player makes is written there. The player aims to beat the high score in the next game. There is a reward system in the game. The player: win bronze if score between 800-1500, silver if score between 1500-3000, and gold if score higher than 3000.

1.3 Design

The design must include specifications that meet the system's functionality requirements. That is why cars were specially selected.

While playing the game, two vehicles appear as obstacles. After passing those vehicles, other cars appear as two obstacles. A random selection is made between 7 cars.

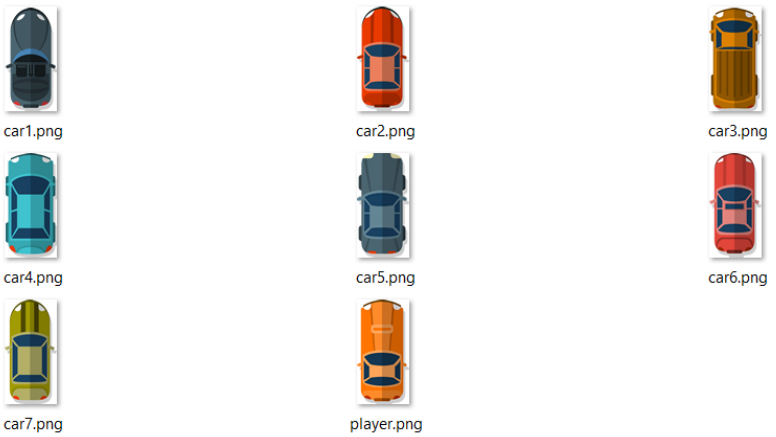

Figure 1- Transparent Car Pictures

Using the photoshop program, the icon was made for the .exe file. It also appears in the upper right corner when playing the icon game.

Figure 2- CarBump Icon

It was made in the design file called "GameBoard" which is a windows form project in Visual Studio. It is designed by selecting various options (font, color, image...) from the "Properties" section.

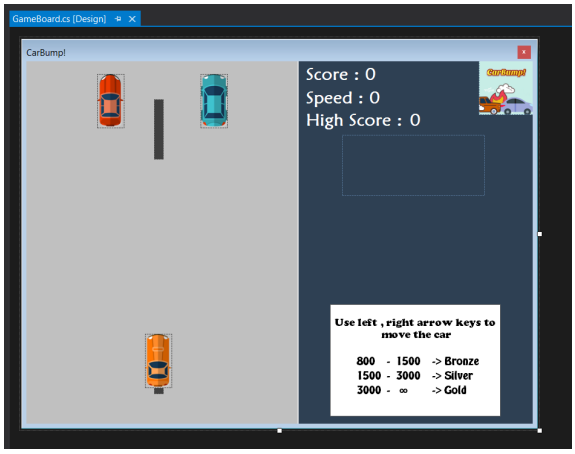

Figure 3- GameBoard.cs

Trophies were made for the reward system using the photoshop program.

Figure 4- Trophies

When the player scores less than 800 points, "You lose" will appear.

Figure 5.1- You Lose Image

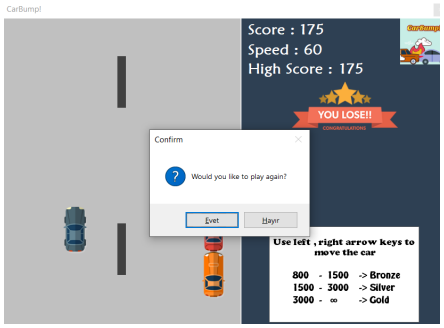

Figure 5.2- "You Lose" Appeared

1.4 Media

"hit.wav" is played when the car hits another vehicle. It was used to give a real crash look.

Tools and Languages

2.1 Visual Studio

The game was made using "windows form". Windows Forms is a UI framework for building Windows desktop applications. It provides one of the most effective ways to create desktop applications based on the visual designer provided in Visual Studio. Functions such as drag-and-drop placement of visual controls make it easy to build desktop applications. Using Windows Forms, you can develop graphically rich applications that are easy to deploy, update and work with when offline or connected to the Internet. The Windows Forms application can access the local hardware and file system of the computer running the application.

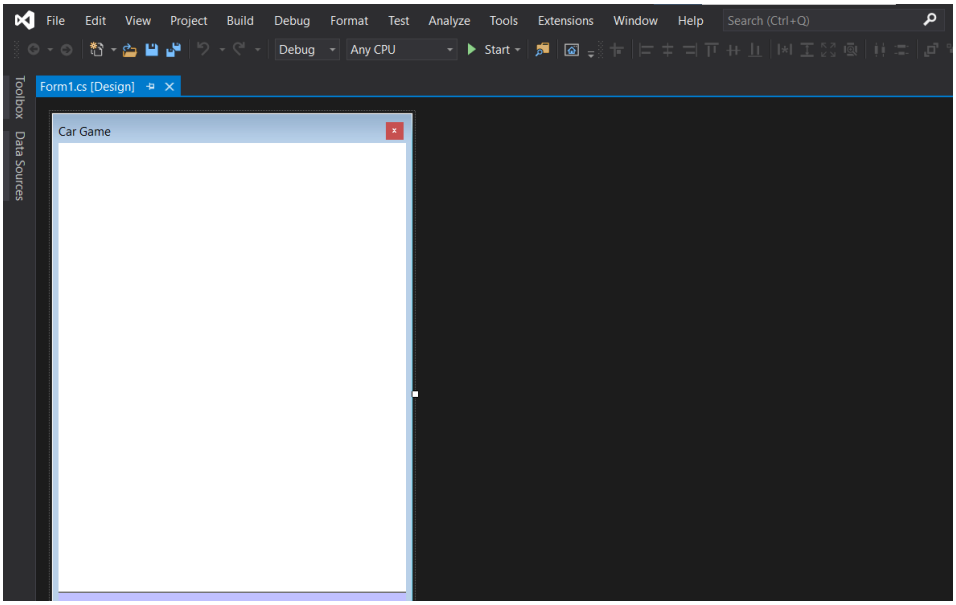

Figure 6- Windows Form in Visual Studio

2.2 C#

The game was written in C# language. Since the C# language is a .NET language, it supports language interoperability. It can access code written in any .NET compatible language and can also inherit classes written in these languages. A portable executable file in C# language can contain any number of classes. In C# language, variables of primitive data types are more powerful. The C# code during compilation generates a ".exe" file, also known as a portable executable file. The ".exe" represents the project's game.

2.3 Adobe Photoshop CS6

When creating icons, you almost always get the best results by creating pixel by pixel or manually fine-tuning the icon. The smaller the size of the icon, even if it starts with a vector, the more significant to create the icon in a pixel environment. You can let Photoshop control certain pixel sizes or add effects to icons.

The transparency of the picture is very significant in this project. Besides, Windows Form is slowing down the game speed of ultra-high-quality images. Therefore, its size must be adjusted. Since this program was used, icon and png files were easily created within the project.

Flowchart Showing the Project Cycle

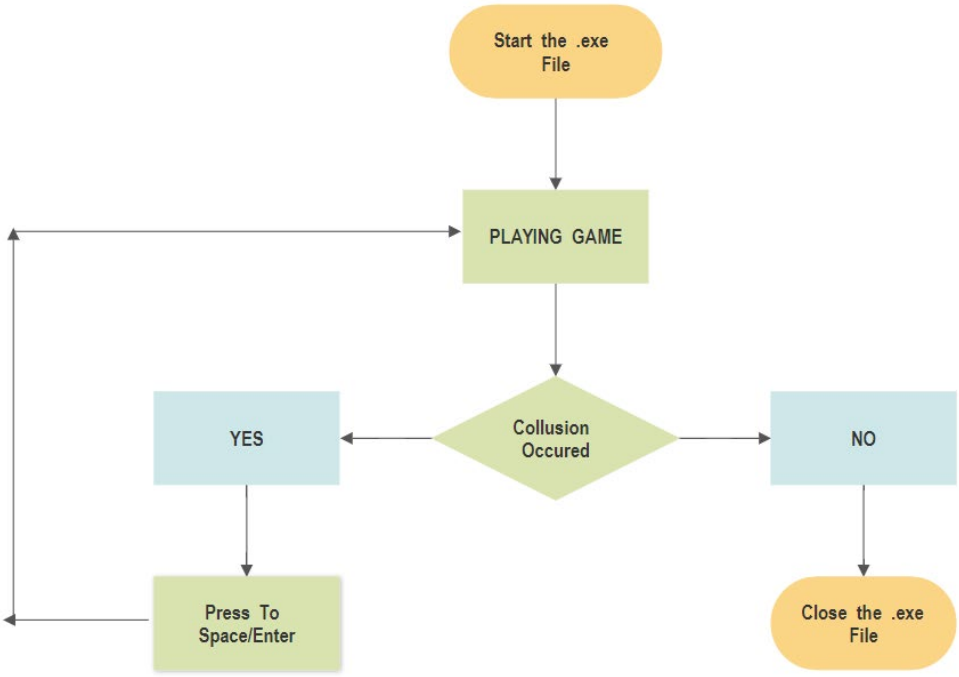

Figure 7.1- Project Cycle

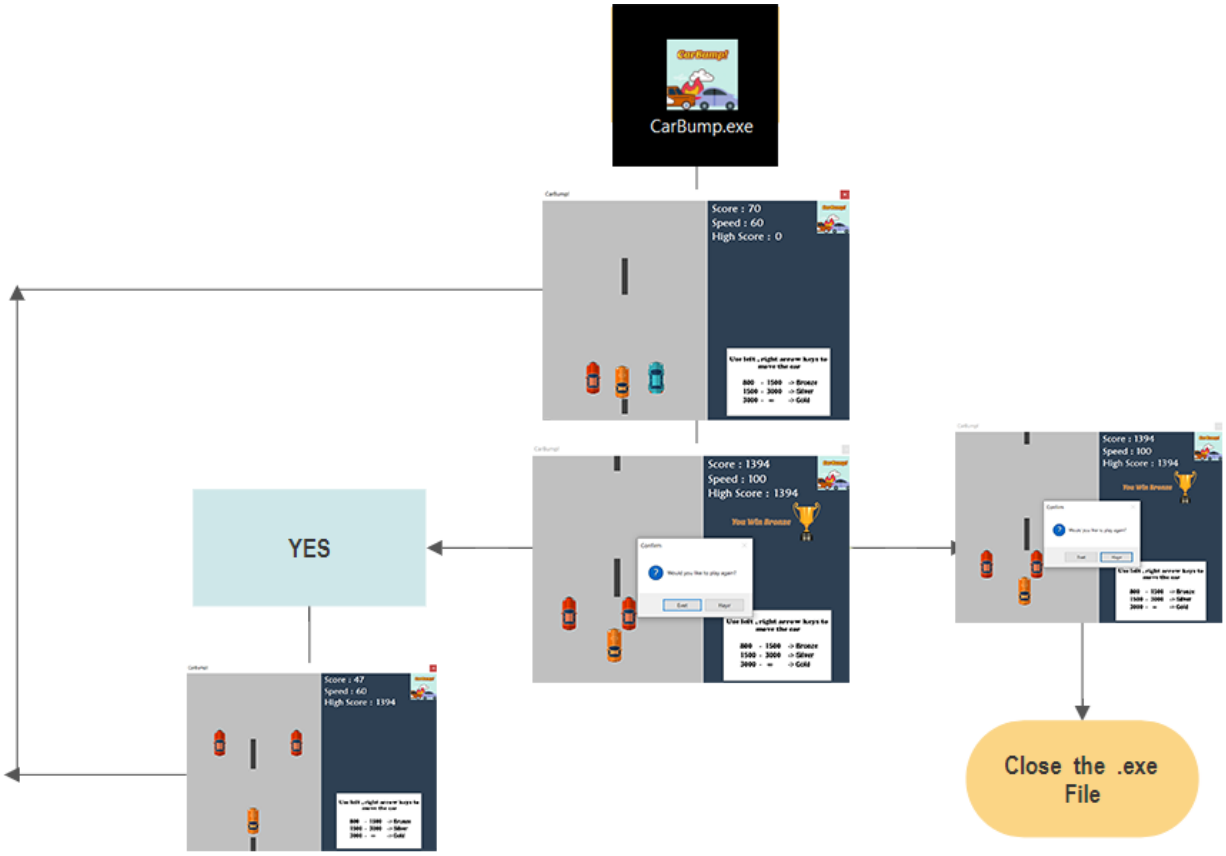

Figure 7.2- Project Cycle with Sample Picture from Game

Source Code

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO;

namespace CarBump
{
    public partial class carGame : Form
    {
        public carGame()
        {
            InitializeComponent();
            HighScore();
        }

        //variables
        int speed = 6;
        int score = 0;
        bool left = false, right = false;


        Random r = new Random();


        //high score
        private void HighScore()
        {

            string userName =
System.Security.Principal.WindowsIdentity.GetCurrent().Name;
            string user = Environment.UserName;
            string FilePath = @"C:\Users\" + user + @"\Documents\Car Game High
Score.txt";

        Start:
            if (File.Exists(FilePath))
            {

                StreamReader Sr = new StreamReader(FilePath);
                lblHighScore.Text = Sr.ReadToEnd();
                Sr.Close();
                int HighScore = int.Parse(lblHighScore.Text);

                if (score > HighScore)
                {
                    lblHighScore.Text = score.ToString();
                    StreamWriter Sw = new StreamWriter(FilePath);
                    Sw.Write(lblHighScore.Text);
                    Sw.Close();
                }
            }
            else
            {
                using (StreamWriter Sw = File.AppendText(FilePath))
```

```
            {
                Sw.Write("0");
                Sw.Close();
            }
            goto Start;
        }

    }

    //key down
    private void carGame_KeyDown(object sender, KeyEventArgs e)
    {
        if (e.KeyCode == Keys.Left)
        {
            left = true;
        }
        else if (e.KeyCode == Keys.Right)
        {
            right = true;
        }

    }

    //key up
    private void carGame_KeyUp(object sender, KeyEventArgs e)
    {
        if (e.KeyCode == Keys.Left)
        {
            left = false;
        }
        else if (e.KeyCode == Keys.Right)
        {
            right = false;
        }
    }


    //game timer
    private void gameTimerEvent(object sender, EventArgs e)
    {
        score++;
        lblScore.Text = "Score : " + score;
        lblSpeed.Text = "Speed : " + (speed * 10);


        //player
        if (right == true && player.Left < (pnlGame.Width - player.Width -
5))
        {
            player.Left += speed;
        }
        else if (left == true && player.Left > 5)
        {
            player.Left -= speed;
        }

        //roadline
        roadLine1.Top += speed;
        roadLine2.Top += speed;
        roadLine1.Left = pnlGame.Width / 2 - 8;
        roadLine2.Left = pnlGame.Width / 2 - 8;
```

```
            if (roadLine1.Top > pnlGame.Height)


            {
                roadLine1.Top = -roadLine1.Height;
            }
            if (roadLine2.Top > pnlGame.Height)
            {
                roadLine2.Top = -roadLine2.Height;
            }


            //car1 car2
            car1.Top += speed;
            car2.Top += speed;
            if (car1.Top > pnlGame.Height)
            {

                car1.Top = -car1.Height;
                car1.Left = r.Next(5, (pnlGame.Width / 2) - car1.Width - 15);
                int carImage = r.Next(1, 8);
                if (carImage == 1) car1.Image = Properties.Resources.car1;
                else if (carImage == 2) car1.Image = Properties.Resources.car2;
                else if (carImage == 3) car1.Image = Properties.Resources.car3;
                else if (carImage == 4) car1.Image = Properties.Resources.car4;
                else if (carImage == 5) car1.Image = Properties.Resources.car5;
                else if (carImage == 6) car1.Image = Properties.Resources.car6;
                else car1.Image = Properties.Resources.car7;

            }
            if (car2.Top > pnlGame.Height)
            {
                car2.Visible = false;
                car2.Top = -car2.Height;
                car2.Left = r.Next((pnlGame.Width / 2) + 5, (pnlGame.Width -
car2.Width - 5));
                int carImage = r.Next(1, 8);
                if (carImage == 1) car2.Image = Properties.Resources.car1;
                else if (carImage == 2) car2.Image = Properties.Resources.car2;
                else if (carImage == 3) car2.Image = Properties.Resources.car3;
                else if (carImage == 4) car2.Image = Properties.Resources.car4;
                else if (carImage == 5) car2.Image = Properties.Resources.car5;
                else if (carImage == 6) car2.Image = Properties.Resources.car6;
                else car2.Image = Properties.Resources.car7;
                car2.Visible = true;
            }

            //speed control
            if (score > 500) speed = 8;
            if (score > 1000) speed = 10;
            if (score > 1500) speed = 12;
            if (score > 2000) speed = 14;
            if (score > 2500) speed = 16;


            //game over
            if (player.Bounds.IntersectsWith(car1.Bounds) ||
player.Bounds.IntersectsWith(car2.Bounds))
            {
                HighScore();
                System.Media.SoundPlayer crash = new
System.Media.SoundPlayer(Properties.Resources.hit);
                crash.Play();
```

```
                gameTimer.Stop();

                gameOver.Visible = true;

                if (score > 0 && score <= 800)
                {
                    gameOver.Image = Properties.Resources.lose;
                }

                if (score > 800 && score <= 1500)
                {
                    gameOver.Image = Properties.Resources.bronze;
                }

                if (score > 1500 && score <= 3000)
                {
                    gameOver.Image = Properties.Resources.silver;
                }
                if (score > 3000)
                {
                    gameOver.Image = Properties.Resources.gold;
                }


                if (MessageBox.Show("Would you like to play again?", "Confirm",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)
                {

                    GameReset();

                }
                else
                    this.Close();

            }
        }

        // game reset
        private void GameReset()
        {

            gameOver.Visible = false;

            speed = 6;
            score = 0;
            left = false;
            right = false;

            car1.Left = r.Next(5, (pnlGame.Width / 2) - car1.Width - 15);
            car1.Top = -car1.Height;

            car2.Left = r.Next((pnlGame.Width / 2) + 5, (pnlGame.Width -
car2.Width - 5));
            car2.Top = -car2.Height;

            player.Left = pnlGame.Width / 2 - 22;

            gameTimer.Start();

        }
 }
}
```