**Отчёт о реализации игры** Car\_Racing\_Game\_WPF\_MOO\_ICT**:**

* **Введение:**

В ходе этого проекта была создана базовая версия игры Car\_Racing\_Game\_WPF\_MOO\_ICT на языке программирования С#.

* **Настройка окна и инициализация xaml:**

<Window x:Class="Car\_Racing\_Game\_WPF\_MOO\_ICT.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:Car\_Racing\_Game\_WPF\_MOO\_ICT"

mc:Ignorable="d"

Title="Top Down Racing Game MOO ICT" Height="517" Width="525">

<Canvas Background="Gray" Name="myCanvas" Focusable="True" KeyDown="OnKeyDown" KeyUp="OnKeyUP">

<Rectangle Height="106" Width="20" Fill="White" Tag="roadMarks" Canvas.Left="237" Canvas.Top="-152" />

<Rectangle Height="106" Width="20" Fill="White" Tag="roadMarks" Canvas.Left="237" Canvas.Top="10" />

<Rectangle Height="106" Width="20" Fill="White" Tag="roadMarks" Canvas.Left="237" Canvas.Top="176" />

<Rectangle Height="106" Width="20" Fill="White" Tag="roadMarks" Canvas.Left="237" Canvas.Top="348" />

<Rectangle Tag="Car" Height="80" Width="55" Fill="Blue" Canvas.Left="90" Canvas.Top="56" />

<Rectangle Tag="Car" Height="80" Width="55" Fill="Purple" Canvas.Left="381" Canvas.Top="286" />

<Rectangle Name="player" Height="80" Width="55" Fill="Yellow" Canvas.Left="222" Canvas.Top="374" />

<Label Name="scoreText" Content="Survived: 00 Seconds" FontSize="18" FontWeight="Bold" />

</Canvas>

</Window>

 Создан экран игры с размером 517x525 пикселей и установлен заголовок окна.

•  **Загрузка ресурсов:**

// set up the player image and the star image from the images folder

playerImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\playerImage.png"));

starImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\star.png"));

// assign the player image to the player rectangle from the canvas

player.Fill = playerImage;

// set the default background colour to gray

myCanvas.Background = Brushes.Gray;**)**

switch (carNum)

{

case 1:

carImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\car1.png"));

break;

case 2:

carImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\car2.png"));

break;

case 3:

carImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\car3.png"));

break;

case 4:

carImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\car4.png"));

break;

case 5:

carImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\car5.png"));

break;

case 6:

carImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\car6.png"));

break;

}

// with each increment of the i we will change the player image to one of the 4 images below

switch (i)

{

case 1:

playerImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\powermode1.png"));

break;

case 2:

playerImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\powermode2.png"));

break;

case 3:

playerImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\powermode3.png"));

break;

case 4:

playerImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\powermode4.png"));

break;

}

// change the background to light coral colour

myCanvas.Background = Brushes.LightCoral;

Фоновое изображение, изображения машин и и машины игрока были загружены из соответствующих файлов.  
 Загруженные изображения были масштабированы до нужных размеров для использования в игре.

•  **Определение других машин:**

foreach (var x in myCanvas.Children.OfType<Rectangle>())

{

// if we find any rectangle with the car tag on it then we will

if ((string)x.Tag == "Car")

{

// set a random location to their top and left position

Canvas.SetTop(x, (rand.Next(100, 400) \* -1));

Canvas.SetLeft(x, rand.Next(0, 430));

// run the change cars function

ChangeCars(x);

}

// if we find a star in the beginning of the game then we will add it to the item remove list

if ((string)x.Tag == "star")

{

itemRemover.Add(x);

}

Были созданы другие машины и звездочка, меняющие положение во время игры.

•  **Инициализация переменных и объектов:**

**bird\_speed = 0**

**gravity = 0.2**

**jump = 1**

**game\_status = &apos;start&apos;**

**score = 0**

**wall\_group = pygame.sprite.Group()**

**spawn\_wall\_event = pygame.USEREVENT**

**pygame.time.set\_timer(spawn\_wall\_event, 1500)**

Были инициализированы переменные для управления игровым процессом, такие как скорость птицы, гравитация и т. д.  
 Создана группа спрайтов для хранения стен.  
 Установлен таймер для регулярного создания новых стен.

•  **Основной игровой цикл:**

if (score >= 10 && score < 20)

{

speed = 12;

}

if (score >= 20 && score < 30)

{

speed = 14;

}

if (score >= 30 && score < 40)

{

speed = 16;

}

if (score >= 40 && score < 50)

{

speed = 18;

}

if (score >= 50 && score < 80)

{

speed = 22;

}

}

private void OnKeyDown(object sender, KeyEventArgs e)

{

// key down function will listen for you the user to press the left or right key and it will change the designated boolean to true

if (e.Key == Key.Left)

{

moveLeft = true;

}

if (e.Key == Key.Right)

{

moveRight = true;

}

}

private void OnKeyUP(object sender, KeyEventArgs e)

{

// when the player releases the left or right key it will set the designated boolean to false

if (e.Key == Key.Left)

{

moveLeft = false;

}

if (e.Key == Key.Right)

{

moveRight = false;

}

// in this case we will listen for the enter key aswell but for this to execute we will need the game over boolean to be true

if (e.Key == Key.Enter && gameOver == true)

{

// if both of these conditions are true then we will run the start game function

StartGame();

}

}

private void StartGame()

{

// thi sis the start game function, this function to reset all of the values back to their default state and start the game

speed = 8; // set speed to 8

gameTimer.Start(); // start the timer

// set all of the boolean to false

moveLeft = false;

moveRight = false;

gameOver = false;

powerMode = false;

// set score to 0

score = 0;

// set the score text to its default content

scoreText.Content = "Survived: 0 Seconds";

// set up the player image and the star image from the images folder

playerImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\playerImage.png"));

starImage.ImageSource = new BitmapImage(new Uri("C:\\car-racing-game-wpf-images-mooict\\star.png"));

// assign the player image to the player rectangle from the canvas

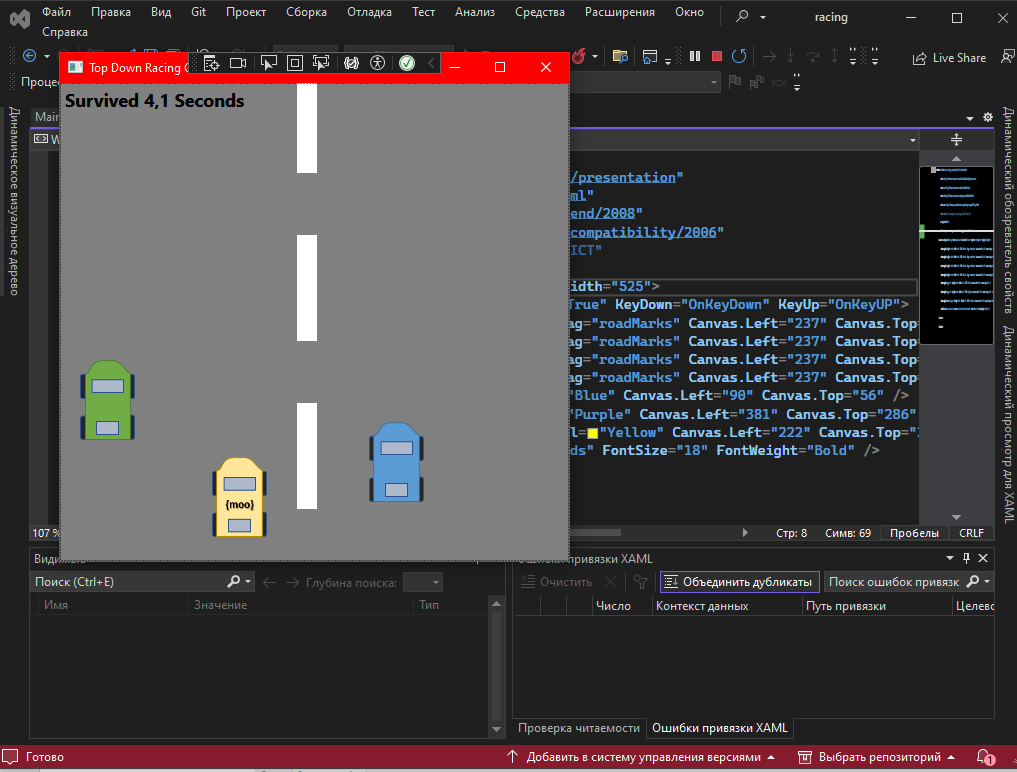
player.Fill = playerImage;

// set the default background colour to gray

myCanvas.Background = Brushes.Gray;

Создан основной игровой цикл, который обрабатывает события, обновляет экран и управляет игровым процессом.  
 В цикле обрабатываются события, такие как нажатия клавиш, создание новых объектов и завершение игры.

Проект может быть дальше развит и улучшен путем добавления новых функций и улучшения графики и игрового процесс

Картинка игры:

Ссылка на GIT : [https://github.com/SelArtSP](https://github.com/SelArtSPB/FB)B/FB