



**«Описание алгоритма
создания игр для мобильных
устройств на UNITY:
руководство пользования»**

Автор работы: **Власов Леонид Дмитриевич,**
обучающийся 10 “А” класса

Научные руководители:
Кригер Анна Олеговна,
Чашникова Ольга Леонидовна

Актуальность и проблема проекта

- Тема мобильных игр актуальна уже 28 последних лет. Каждый из нас, наверное, играл или слышал о различных видеоиграх. Они стали неотъемлемой частью жизни человека, в чью страну пришла компьютеризация. Наверное, каждый человек задавался вопросом, как же устроена та или иная игра.
- Передо мной возникла **проблема**: как начать создавать мобильные игры, доступными мне средствами.

Цели и задачи

- **Цель проекта:** исследовать всевозможные для школьника средства создания мобильных игр и создать свою игру на выбранном движке.
- **Задачи проекта:**
 - Изучить историю создания игр.
 - Рассмотреть виды мобильных игр.
 - Проанализировать интернет-источники по созданию мобильных игр.
 - Рассмотреть технологию создания мобильных игр.
 - Разработать свою игру.

История создания мобильных игр

- Уже в далеких 90-ых учёные начали разработки своих первых мобильных игр и с течением времени люди всё прогрессируют и прогрессируют в этой сфере. С появлением операционных систем Android и iOS термин «телефон» был заменен на «смартфон». С каждым годом смартфоны становились значительно мощнее, открыв новые возможности в создании игр для разработчиков. Игры переходили на новый уровень дизайна, производительности, играбельности и т.д.



Самые популярные игры на 1990-2023гг.

Нынешнее время:

2010 – 2020гг.:

- Apex Legends Mobile (Вершина легенды мобайл).
- Tower of Fantasy (Башня фантазии).
- Diablo Immortal (Диабло Бессмертный).

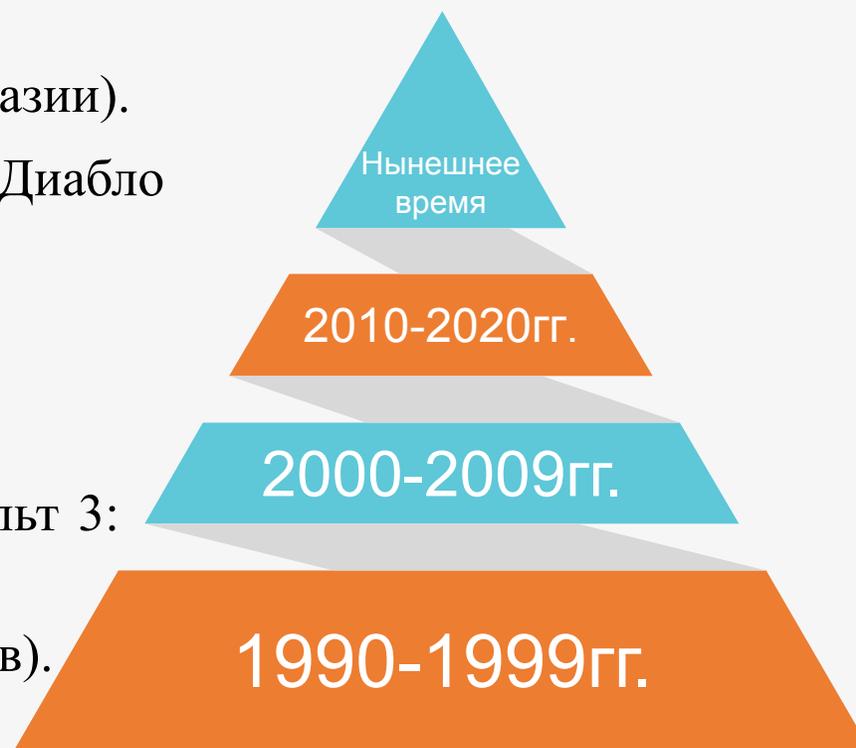
- Fruit Ninja (Фруктовый Ниндзя).
- Clash of clans (Война кланов).
- Subway surfers (Сёрферы метро).

2000-2009гг.:

- Snake (Змейка).
- Asphalt 3: Street Rules (Асфальт 3: Уличные правила).
- Bounce Tales (Истории отказов).

1990-1999гг.:

- Streets of Rage 2 (Улицы ярости 2)
- The Amazing Spider – Man (Удивительный Человек–паук)
- Sonic the Hedgehog (Ёжик Соник)



Обзор классификаций мобильных игр

- Термин «мобильная игра» — это такая игра, в которую можно играть в движении, используя свой переносной помощник. В ней, как и в компьютерной игре, есть свои правила и цели. Мобильный гейминг — говоря о нём, всегда идёт речь про небольшой дисплей, доступность в любое время, более частые и короткие, в сравнении с компьютерным геймингом, игровые сессии.

Классификация

О каждом понемногу

Ролевые игры

- главный герой в течении игры имеет возможность изменять свои характеристики

Головоломки/логические

- игроку предлагается решить какую-то сложную задачу

Настольные

- компьютерная реализация настольных игр, шахматы, карты, шашки, «Монополия»

Текстовые

- общение с игроком осуществляется посредством текстовой информации.

Файтинг

- игра состоит исключительно из поединков двух и более противников с применением рукопашного боя

Анализ игровых движков

- Основными средствами разработки игр являются игровые движки, которые отвечают за низкоуровневое описание физики игры, правил рендеринга графики и пр. и графические редакторы для отрисовки графики. Для разработки мобильной игры основным средством является игровой движок – программное средство, для разработки мобильных игр.



- Редактор Unity 3D имеет простой интерфейс, который легко настраивать, состоящий из различных окон, благодаря чему можно производить отладку игры прямо в редакторе.

- Существует большое количество игровых движков, но мой взор упал именно на движок под названием Unity 3D. Поэтому я остановился именно на Unity 3D и объектно-ориентированном языке программирования общего назначения C#.

Этап разработки графического оформления

- Прежде всего, стоит выделить основные понятия, такие как тайл (**Tile**) и спрайт (**Sprite**):
 - **Тайл (Tile)** – небольших размеров повторяющийся фрагмент, который служит для постройки изображений больших размеров. Часто используется для создания уровней для двумерных игр.
 - **Спрайт (Sprite)** – графический объект, представляющий собой растровое изображение. Используется в компьютерной графике как основная единица для анимаций двумерных объектов.

Этап разработки игры

- Для реализации мобильной игры был выбран игровой движок Unity.
- При запуске Unity мы увидим окно проекта (рисунок 1).
- Посередине находятся окно сцены, окно анимации, а также окно игры. Первое служит для создания общей композиции уровня и добавления новых объектов, второе представляет возможность для создания и редактирования анимаций объектов, третье окно представляет вид из камеры показывая, каким образом будет выглядеть игра на данный момент.

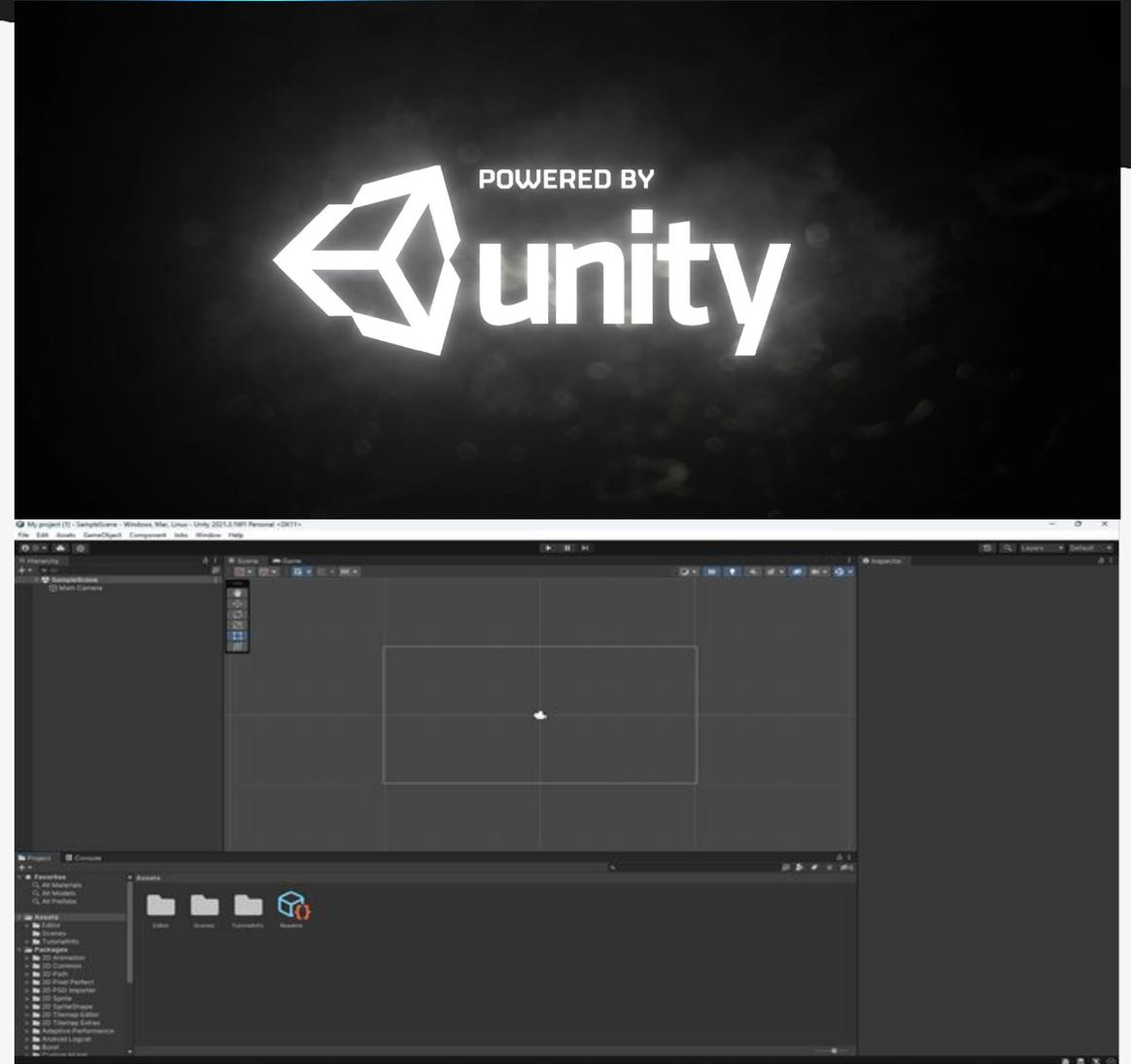


Рис. 1. Окно

Этап разработки игры

- Слева находится окно иерархии (рисунок 2), здесь показываются все объекты, участвующие в данной сцене. С самого начала здесь находится только камера.

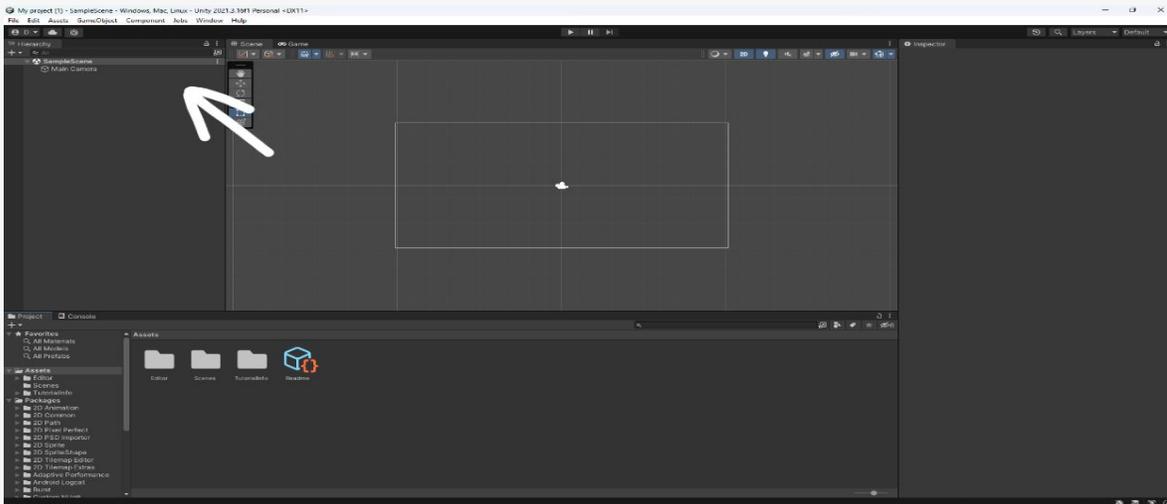


Рис. 2. Окно иерархии



Рис. 3. Окно инспектора

- Справа окно инспектора (рисунок 3), здесь отображаются все текущие свойства выбранного объекта.

Этап разработки игры

- Снизу находятся три окна: окно проекта, окно аниматора и консоль. В окне проекта отображаются все объекты, добавленные в текущую игру, в том числе скрипты, анимации и прочее. В окне аниматора создаются связи и правила перехода между различными анимациями.

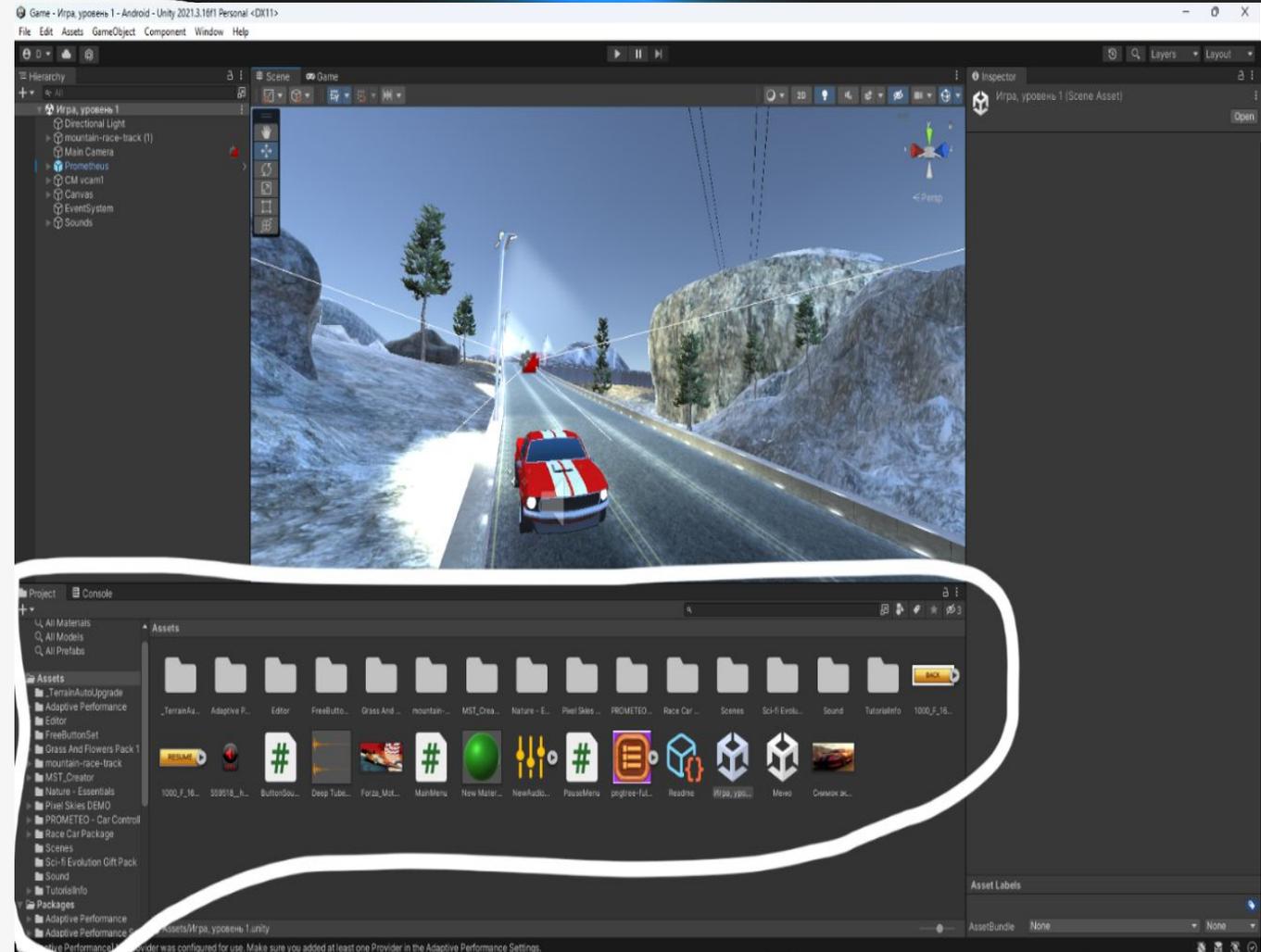


Рис. 4. Консоль и необходимые объекты.

Этап разработки игры

- Итак, мы создали сцену и придумали тему игры, а именно «Гонки по зимней трассе». Для помощи в создании игры мы воспользуемся таким приложением как Unity AssetsStore, в котором мы сможем найти готовые объекты, анимации, скрипты, проекты и многое другое. Но из этого всего мы возьмём только часть карты и модель машины.

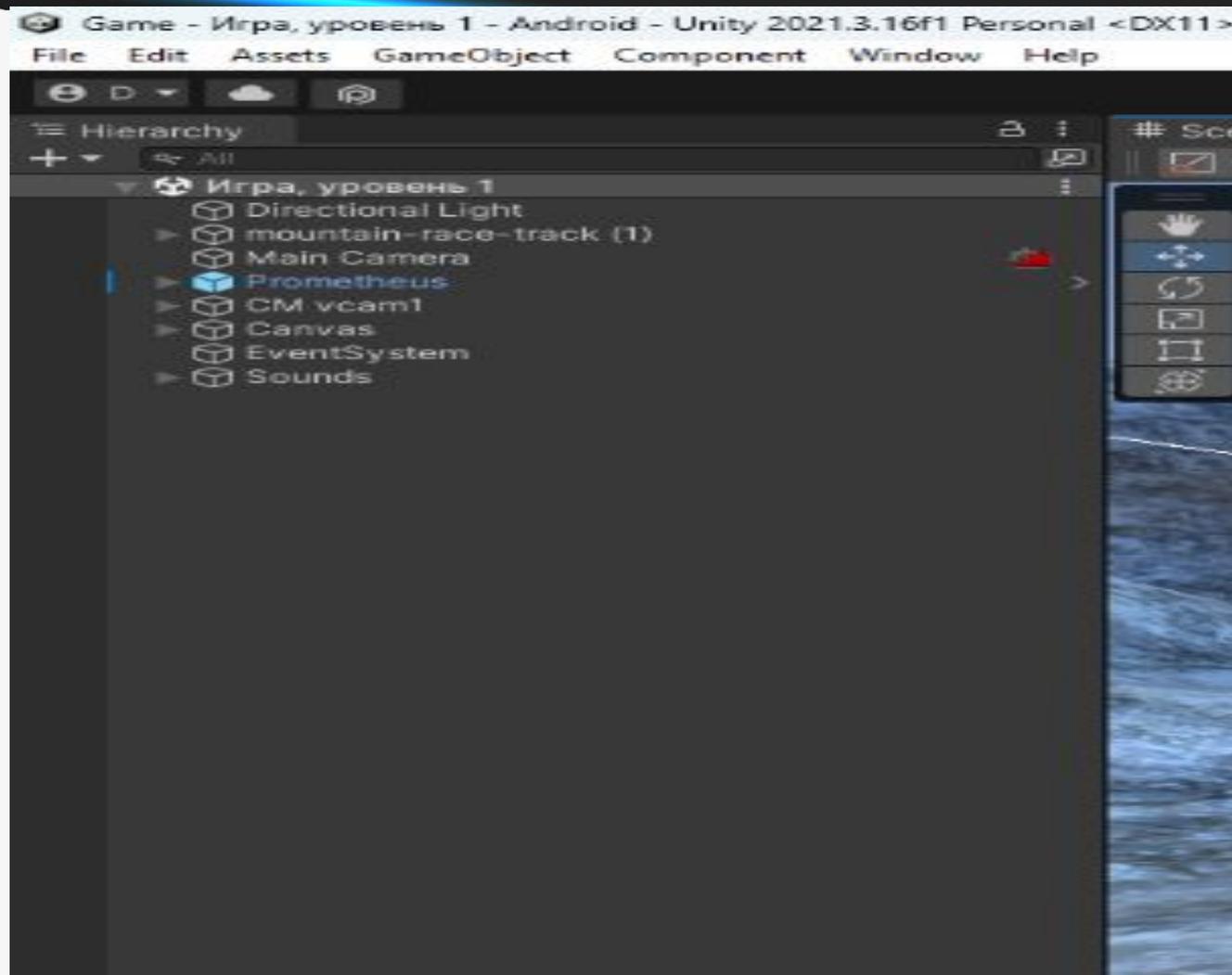
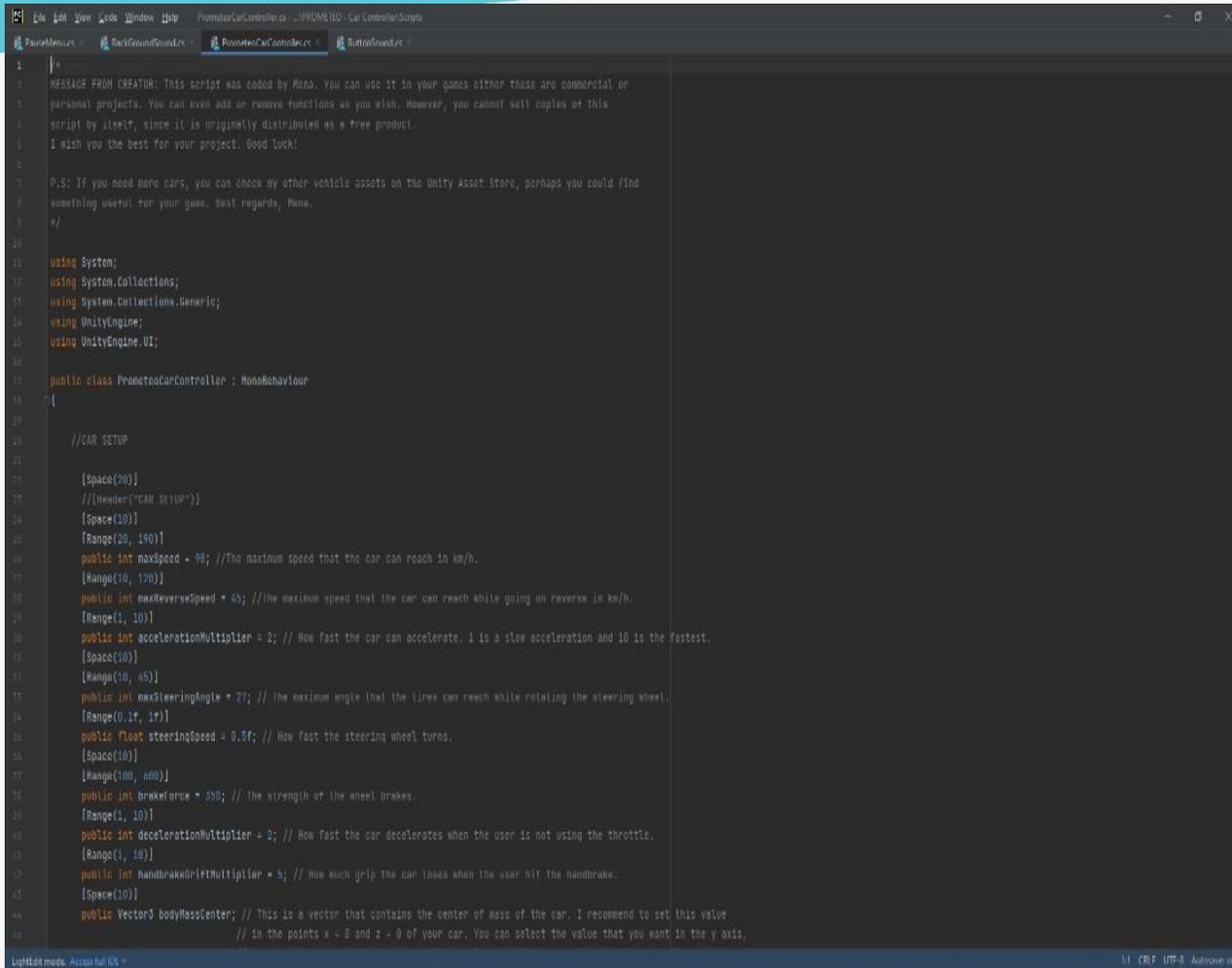


Рис. 5. Предлагаемые объекты для игры.

Этап разработки игры

A screenshot of a Unity C# script editor showing a script named 'PromoteoCarController'. The script includes a copyright notice from 'Mena' and a list of public variables for car parameters. The variables include 'maxSpeed', 'maxReverseSpeed', 'accelerationMultiplier', 'maxSteeringAngle', 'steeringSpeed', 'brakeForce', 'decelerationMultiplier', 'handbrakeRiftMultiplier', and 'bodyMassCenter'. Each variable is accompanied by a comment explaining its purpose and a range attribute.

```
1 *
2 MESSAGE FROM CREATOR: This script was coded by Mena. You can use it in your games either these are commercial or
3 personal projects. You can even add or remove functions as you wish. However, you cannot sell copies of this
4 script by itself, since it is originally distributed as a free product.
5 I wish you the best for your project. Good luck!
6
7 P.S: If you need more cars, you can check by other vehicle assets on the Unity Asset Store, perhaps you could find
8 something useful for your game. Best regards, Mena.
9
10 */
11
12 using System;
13 using System.Collections;
14 using System.Collections.Generic;
15 using UnityEngine;
16 using UnityEngine.UI;
17
18 public class PromoteoCarController : MonoBehaviour
19 {
20
21     //CAR SETUP
22
23     [Space(20)]
24     //Header("CAR SETUP")
25     [Space(10)]
26     [Range(20, 190)]
27     public int maxSpeed = 98; //The maximum speed that the car can reach in km/h.
28     [Range(10, 170)]
29     public int maxReverseSpeed = 45; //the maximum speed that the car can reach while going on reverse in km/h.
30     [Range(1, 10)]
31     public int accelerationMultiplier = 2; // How fast the car can accelerate. 1 is a slow acceleration and 10 is the fastest.
32     [Space(10)]
33     [Range(10, 45)]
34     public int maxSteeringAngle = 27; // The maximum angle that the tires can reach while rotating the steering wheel.
35     [Range(0.1f, 1f)]
36     public float steeringSpeed = 0.5f; // How fast the steering wheel turns.
37     [Space(10)]
38     [Range(100, 500)]
39     public int brakeForce = 500; // The strength of the wheel brakes.
40     [Range(1, 10)]
41     public int decelerationMultiplier = 2; // How fast the car decelerates when the user is not using the throttle.
42     [Range(1, 10)]
43     public int handbrakeRiftMultiplier = 5; // How much grip the car loses when the user hit the handbrake.
44     [Space(10)]
45     public Vector3 bodyMassCenter; // This is a vector that contains the center of mass of the car. I recommend to set this value
46     // in the points x = 0 and z = 0 of your car. You can select the value that you want in the y axis.
```

- Далее пишем скрипт на взаимодействие и управление машиной, а точнее джойстик и все кнопки для взаимодействия машиной. Но так как мы лишь пробуем себя в создании игры, то нам очень сильно помогут интернет поисковики.

Рис. 6. Часть скрипта (кода) для взаимодействия машиной.

Этап разработки игры

- В конце концов нам осталось сделать игровое меню нашего проекта, затем всё связать и наша игра будет завершена.
- Результат: Мы создали простенькую игру для мобильного телефона, с помощью сторонних помощников.

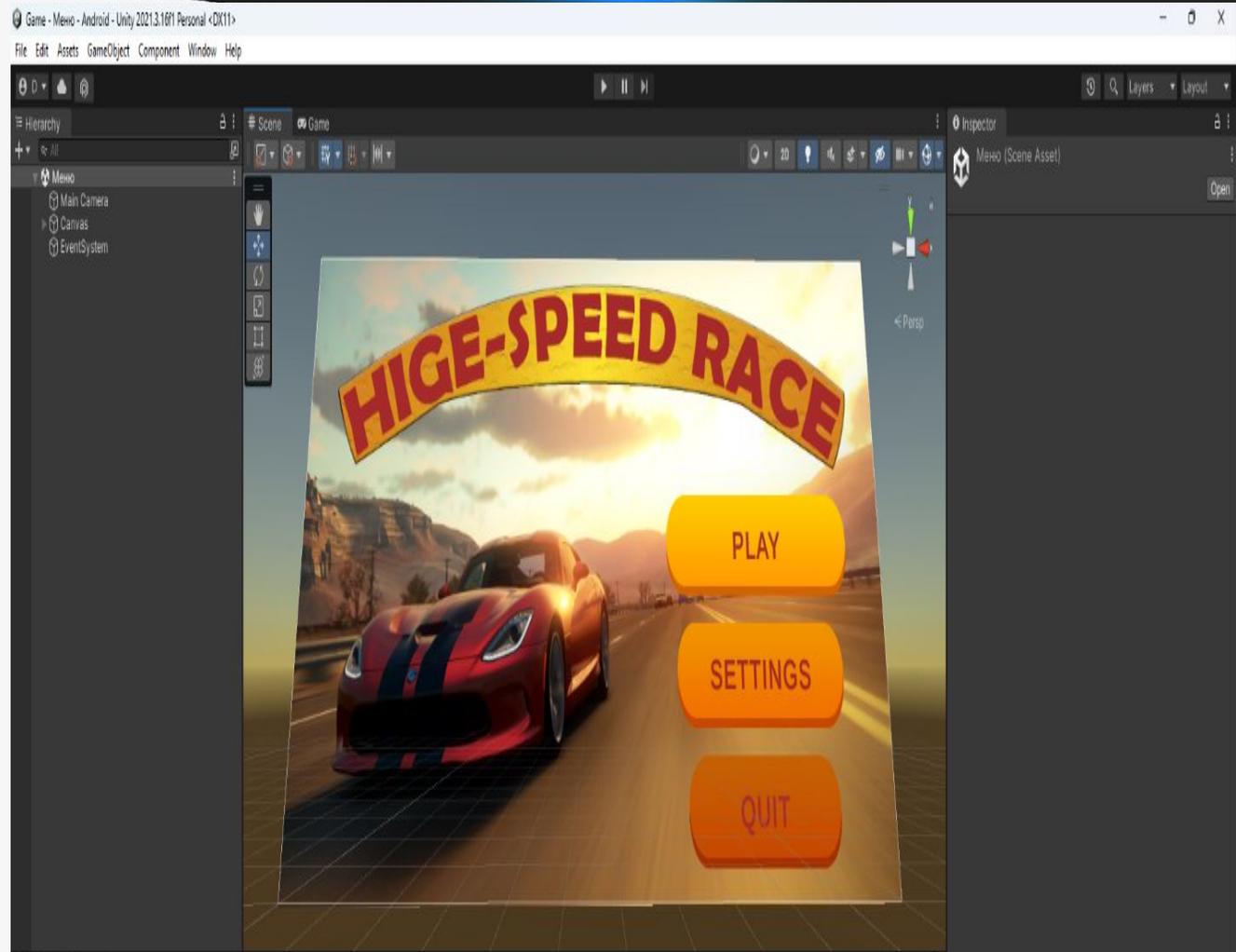


Рис. 7. Меню проекта



vk.com/stolbr

**спасибо за внимание
можно без аплодисментов,5
достаточно.**