

Министерство образования, науки и молодежной политики
Краснодарского края
ГБПОУ КК ЕПК

Отделение школьного образования и прикладной информатики
Специальность 09.02.05 Прикладная информатика (по отраслям)

Студент И–41 группы

БАРСУКОВ РОМАН РОМАНОВИЧ

РАЗРАБОТКА 2Д ИГРЫ СРЕДСТВАМИ UNITY

Дипломный проект

Научный руководитель:
Рубан Александр Сергеевич

г. Ейск, 2020

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	2
ГЛАВА 1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И СРЕДСТВ РАЗРАБОТКИ	4
1.1 Анализ предметной области.....	4
1.2 Описание технологий и методов реализации программного продукта	5
ГЛАВА 2 РЕАЛИЗАЦИЯ ИГРОВОЙ ЛОГИКИ И КОМПОНЕНТОВ ПРОГРАММНОГО ПРОДУКТА.....	8
2.1 Проектирование игровой логики.....	8
2.2 Физическая реализация программного продукта.....	11
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ.....	34

ВВЕДЕНИЕ

С каждым годом популярность компьютерных игр растет, все больше и больше людей играют в компьютерные игры. Результатом технологического развития, а также массового распространения персональных компьютеров, планшетов и мобильных устройств, стал все более расширяющийся рынок развлечений, что является веским аргументом в пользу проектов по созданию новых игр. Актуальность разработки игр становится все более неоспоримой, рынок игр увеличивается в геометрической прогрессии. Самые удачные проекты продаются миллионными тиражами. Зарождение игровой индустрии началось в 1970-х годах одновременно с продажей первых компьютеров, рассчитанных на массового потребителя. За 40 с небольшим лет индустрия компьютерных игр развилась до таких масштабов, что уже во многом опережает своих ближайших конкурентов: киноиндустрию, музыкальную индустрию, шоу-бизнес.

Многие разработчики рассматривают индустрию компьютерных игр лишь как сектор экономики, в котором можно неплохо заработать. Но, на самом деле, создание игр — это ещё и инструмент культуры. Ретро игры были придуманы много лет назад, но и сейчас они пользуются большой востребованностью. Яркий пример - Mario, серия компьютерных аркад, в которые с удовольствием играют пользователи нескольких поколений. Но, наверное, она бы не была такой популярной, если бы не имела развитие. Super Mario Maker - игра, в которой игрок имеет возможность не только проходить уровни, но и создавать свои собственные.

По официальным данным на 2016 год компьютерными и видео играми увлекаются 78% интернет-аудитории страны, что составляет 58% от всего населения России. (Конечно, большую часть всех учтенных игроков занимают игроки в социальных сетях, играющее время от времени. Но и ядро игровой аудитории, играющее постоянно, растет из года в год). Игроки — это молодая и наиболее активная часть общества. В настоящий момент компьютерные игры

заменяют детям сказки, подросткам дают примеры для подражания, взрослым заменяют хобби и заполняют свободное время. Таким образом, содержание компьютерных игр производит большее воздействие на социум, чем содержание сказок, книг, комиксов вместе взятых. Игры стали очень важным элементом современной культуры, нельзя игнорировать это направление, нужно всеми силами содействовать лучшим представителям отечественной игровой индустрии. В данной дипломной работе будет разработана 2D игра на игровом движке «Unity5».

Актуальность темы дипломного проекта определена тем, что игровая индустрия очень быстро набирает обороты во всем мире, и приносит большие доходы как разработчикам, так и государству в виде налогов, и так же игры пользуются очень большим спросом во всем мире, любого возраста. В данном проекте было создано 2D игра на платформе «Unity 5», она была выбрана, потому что является бесплатной и проста в процессе разработке и не требует больших финансовых ресурсов.

Цель: разработка 2D игры средствами «Unity5».

Задачи:

1. Описать предметную область.
2. Рассмотреть технологии реализации программного продукта.
3. Спроектировать структуру 2D игры.
4. Разработать 2D игру.

Структура: работа состоит из введения, двух глав, заключения, списка используемых источников. Во введении обоснована актуальность темы исследования. В первой главе описан анализ предметной области и средств разработки. Вторая глава посвящена описанию процесса разработки прикладного программного продукта. Заключение отражает результат достижения цели работы. Список используемых источников насчитывает 20 источников.

ГЛАВА 1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И СРЕДСТВ РАЗРАБОТКИ

1.1 Анализ предметной области

Предметная область — это часть реального мира, подлежащая изучению с целью создания базы данных для автоматизации процесса управления. Предметной областью этого проекта является 2D игра на платформе «Unity 5» [3].

Компьютерные игры — это особый вид развлечений, который работает при помощи электронного устройства, оборудованного микроконтроллером. Многие из них ввиду наличия движущегося изображения, демонстрируемого на экране телевизора или компьютерного монитора, принято называть видеоиграми[2].

Если обратиться к истории появления и развития компьютерных игр, то датой рождения этой крупной индустрии можно назвать 1958 год. Тогда компьютер считался чем-то специфическим и даже в некоторой степени предметом роскоши, доступ к которому имели лишь избранные. Самую первую компьютерную игру написал один из студентов американского Массачусетского технологического института. Называлась она «В космос» (Spaceward) и работала на компьютере модели PDP-1, которая по размерам напоминала автомобиль. По сюжету в ней сражались космические корабли и задачей игрока было уничтожить как можно больше единиц противника.

Инди-игры — это те проекты, что создаются независимыми авторами. То есть людьми, которые работают не на крупных издателей вроде Activision, Ubisoft или EA, а на себя. Как правило, инди-студии состоят всего из нескольких человек. Встречаются и разработчики-одиночки[8].

В качестве дипломного проекта была выбрана тема, разработка инди-игры на платформе «Unity 5». Для разработки 2D игры была выбрана платформа «Unity 5», потому что является удобной программой для создания

игр.

Выбор платформы «Unity 5» обусловлен следующими причинами:

- является бесплатной версией;
- возможна разработка приложения для браузера;
- возможность написания игрового кода на трёх язык программирования JavaScript, C#, Boo;
- кроссплатформенность;
- производительность и масштабируемость;
- запуск любого приложения в веб-плагине.

Как видим «Unity 5» обладает большим преимуществом в разработке 2D игр.

1.2 Описание технологий и методов реализации программного продукта

Unity — межплатформенная среда разработки компьютерных игр, разработанная американской компанией Unity Technologies. Unity позволяет создавать приложения, работающие на более чем 25 различных платформах, включающих персональные компьютеры, игровые консоли, мобильные устройства, интернет-приложения и другие. Выпуск Unity состоялся в 2005 году и с того времени идёт постоянное развитие. Основными преимуществами Unity являются наличие визуальной среды разработки, межплатформенной поддержки и модульной системы компонентов. К недостаткам относят появление сложностей при работе с многокомпонентными схемами и затруднения при подключении внешних библиотек. На Unity написаны тысячи игр, приложений, визуализации математических моделей, которые охватывают множество платформ и жанров. При этом Unity используется как крупными разработчиками, так и независимыми студиями [2].

Как правило, игровой движок предоставляет множество функциональных возможностей, позволяющих их задействовать в различных играх, в которые

входят моделирование физических сред, карты нормалей, динамические тени и многое другое. В отличие от многих игровых движков, у Unity имеется два основных преимущества: наличие визуальной среды разработки и межплатформенная поддержка. Первый фактор включает не только инструментарий визуального моделирования, но и интегрированную среду, цепочку сборки, что направлено на повышение производительности разработчиков, в частности, этапов создания прототипов и тестирования. Под межплатформенной поддержкой предоставляется не только места развертывания (установка на персональном компьютере, на мобильном устройстве, консоли), но и наличие инструментария разработки (интегрированная среда может использоваться под Windows и Mac OS) [5].

Системные требования для платформы «Unity 5». Различные варианты работы допускают использование различного программного обеспечения на компьютерах, на которых функционируют те или иные приложения системы «Unity 5». В обобщенном виде информация об системных требованиях собрана в таблице 1.

Таблица 1 – Системные требования для «Unity 5».

	32-разрядная архитектура	64- разрядная архитектура
Процессор	Intel Pentium IV/Xeon 2,4 ГГц и новее	с архитектурой x86-64
ОЗУ	от 12 Гб	от 4 Гб
HDD	от 40 Гб	от 40 Гб
USB-порт	да	да
SVGA-видеокарта	да	да

Для хранения данных в Unity 5 можно использовать различные СУБД от самых маленьких до самых крупных, для данного проекта была выбрана СУБД SQLite поскольку она компактная, встраиваемая реляционная база данных. Исходный код библиотеки передан в общественное достояние [11].

SQLite не является отдельно работающим процессом, с которым взаимодействует программа, а предоставляет библиотеку, с которой программа компонуется и движок становится составной частью программы. Таким

образом, в качестве протокола обмена используются вызовы функций (API) библиотеки SQLite. Такой подход уменьшает накладные расходы, время отклика и упрощает программу. SQLite хранит всю базу данных (включая определения, таблицы, индексы и данные) в единственном стандартном файле на том компьютере, это является целесообразно для нашего проекта, поскольку это даст нам оптимизация и быстроту работы проекта [11].

Для работы в Unity 5 был выбран язык программирования C#, поскольку он является самым гибки удобным из доступных трех языков программирования. Поскольку C# поддерживает много различных библиотек и прост в написание программного кода.

Adobe illustrator — это программа для создания векторной графики. Часто она используется для создания иллюстраций, мультяшной графики, диаграмм, чартов и логотипов. В отличие от растровых изображений, в которых информация хранится в сетке отдельных точек, Illustrator при создании форм использует математические уравнения. Это позволяет менять размеры векторных изображений без потерь в разрешении [13].

Графическая часть была разработана в программе Adobe Illustrator, поскольку он является очень гибким и простым в использование, и очень хорошо подходит для создания графики для 2D игр [13].

Таким образом мы рассмотрели основные моменты предметной области и технологии реализации программного продукта, что позволит перейти к этапу физической реализации программного продукта

ГЛАВА 2 РЕАЛИЗАЦИЯ ИГРОВОЙ ЛОГИКИ И КОМПОНЕНТОВ ПРОГРАММНОГО ПРОДУКТА

2.1 Проектирование игровой логики

Для удобства программирования мы реализовали ряд диаграмм, которые отражают основные моменты функционирования программного продукта. На рисунке 1 представлена схема главного меню игры.

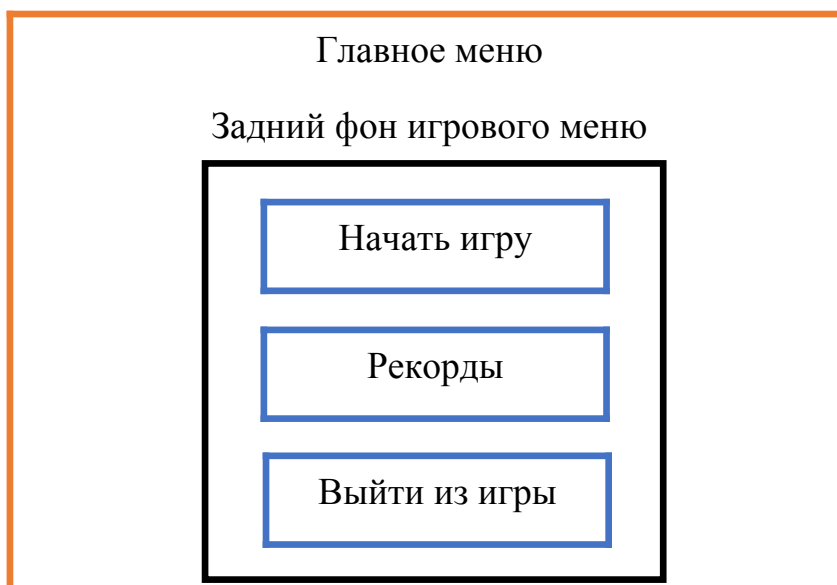


Рисунок 1 – Сцена главного меню

В Сцене главного меню будут располагаться три кнопки, при нажатие будут открывать следующие сцены такие как таблицу рекордов, саму игру, и выход из игры показан на рисунке 1.



Рисунок 2 – Сцена авторизации

В Сцене авторизации будет окно для ввода имени игрока, и кнопка, которая начинает игру и записывает введённое имя игрока рисунок 2.

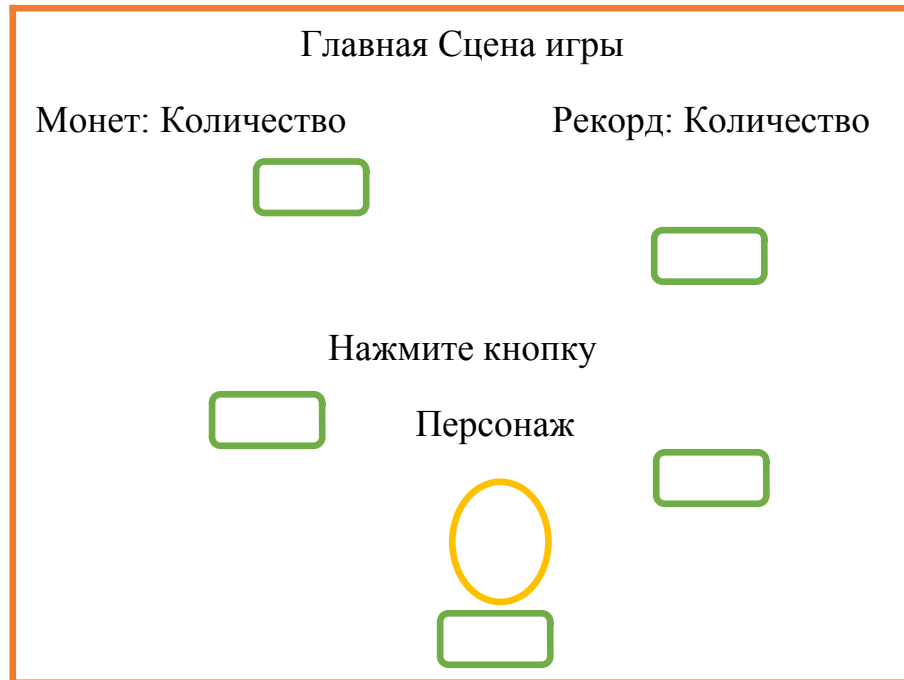


Рисунок 3 – Сцена игры

Сцена игры будет содержать в себе персонажа, два счётчика очков рекордов и счётчик рекорда, и игровые блоки, по которым будет прыгать персонаж рисунок 3.



Рисунок 4 – Сцена таблицы рекордов

Сцена таблицы рекордов предназначена для показа личной статистики игроков, а также показывает какой игрок набрал больше всего очков рисунок 4.

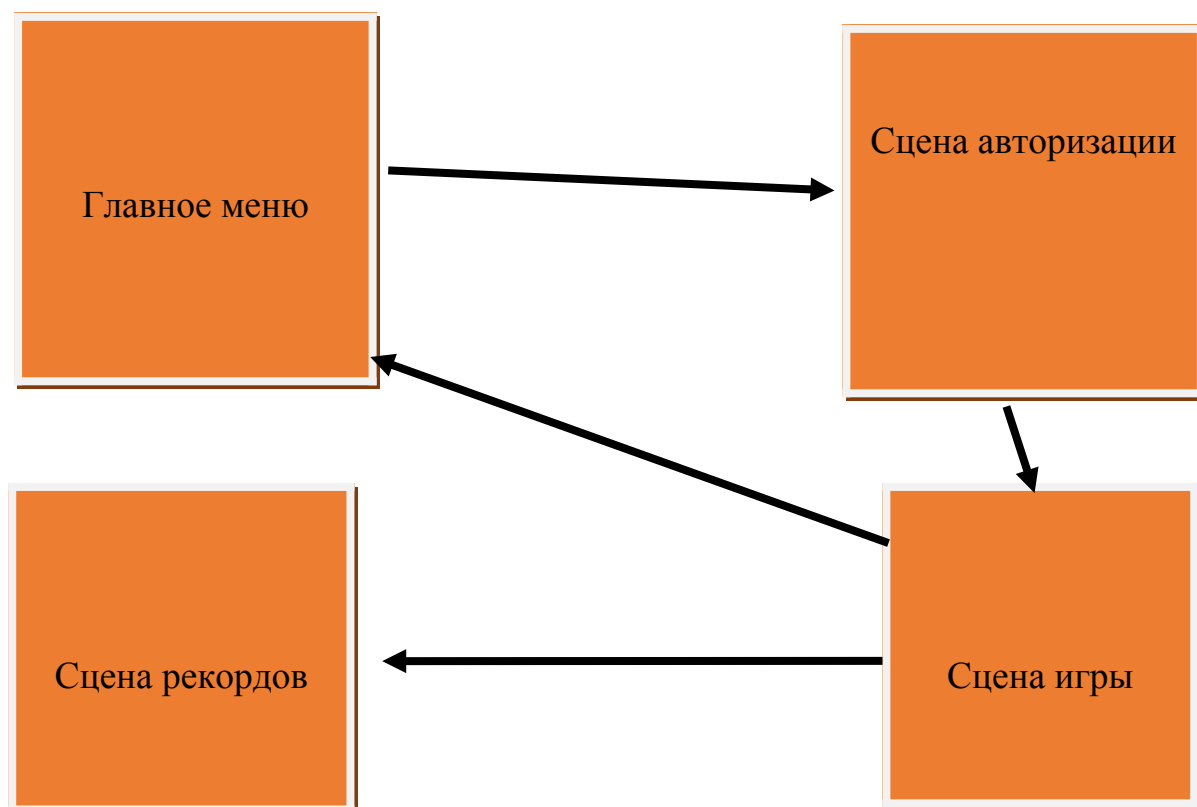


Рисунок 5 – Игровая логика

Взаимодействие между четырьмя сценами начинается с главного меню, после нажатия кнопки начать играть, игрок попадает в сцену авторизации, где он вводит своё имя и нажимает кнопку подключиться, и игрок падает в сцену с самой игрой, где игрок набирает очки и после окончания игры, игрок может выйти в главное меню с помощью кнопку, и зайти в сцену рекордов и посмотреть свой личный рекорд и рекорд других игроков рисунок 5.

Для хранения данных используется SQLiteStudio.

Таблица 2 – Структура таблицы «Player»

ID_player	Name	Score
1	Player	555
2	Player 1	450
3	Player 2	350
4	Player 3	270

5	Player 4	220
---	----------	-----

В данном разделе мы спроектировали игровую логику нашей игры, что позволит приступить к физической реализации программного продукта.

2.2 Физическая реализация программного продукта

Для того чтобы начать работу с Unity 5 необходимо открыть Unity Hub и нажать кнопку New, далее выбираем режим 2D и даём название нашему проекту и путь, где он будет находиться, и нажимаем кнопку Create рисунок 6.

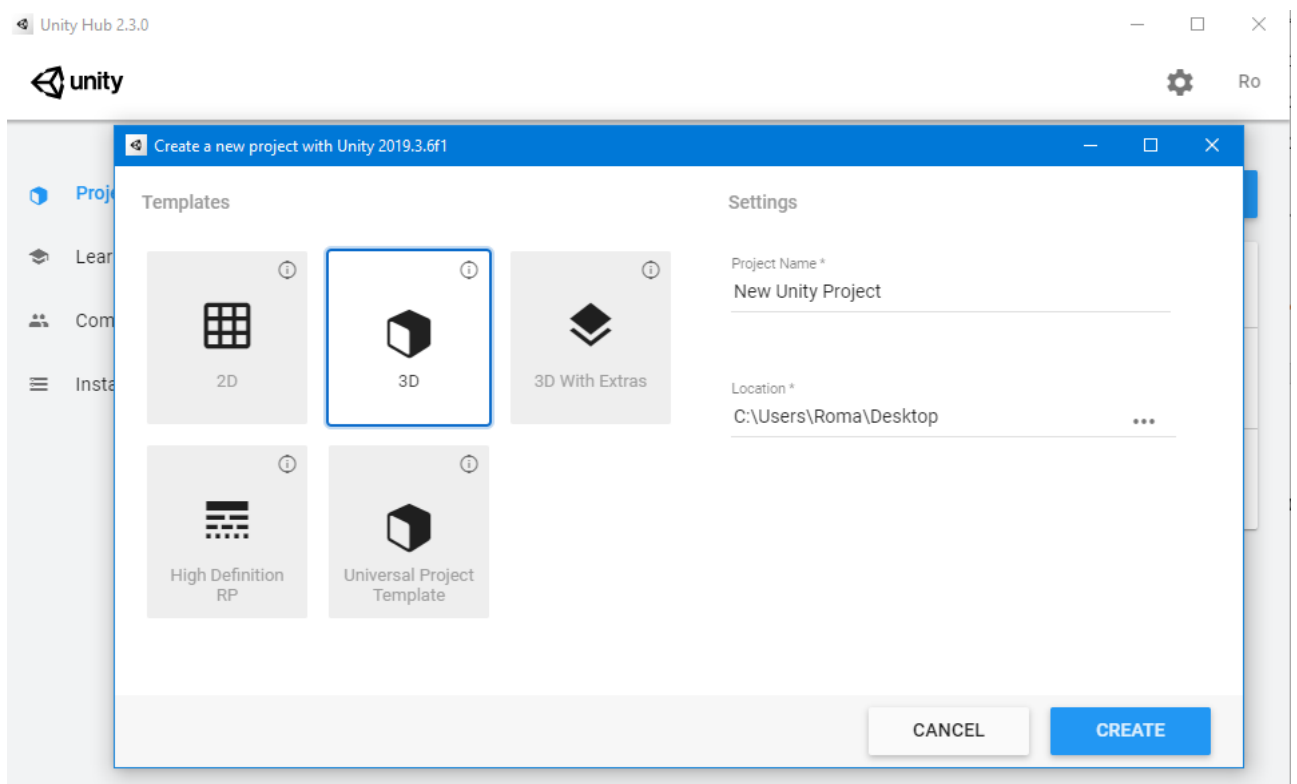


Рисунок 6 – Меню Unity Hub

Это главный экран нашего проекта в Unity 5, в котором будет производиться работа, по середине синее окно отображает нашу игру, в левом столбце показывается какие объекты находятся на сцене в вкладке Assets, показываются наши будущие файлы проекта рисунок 7.

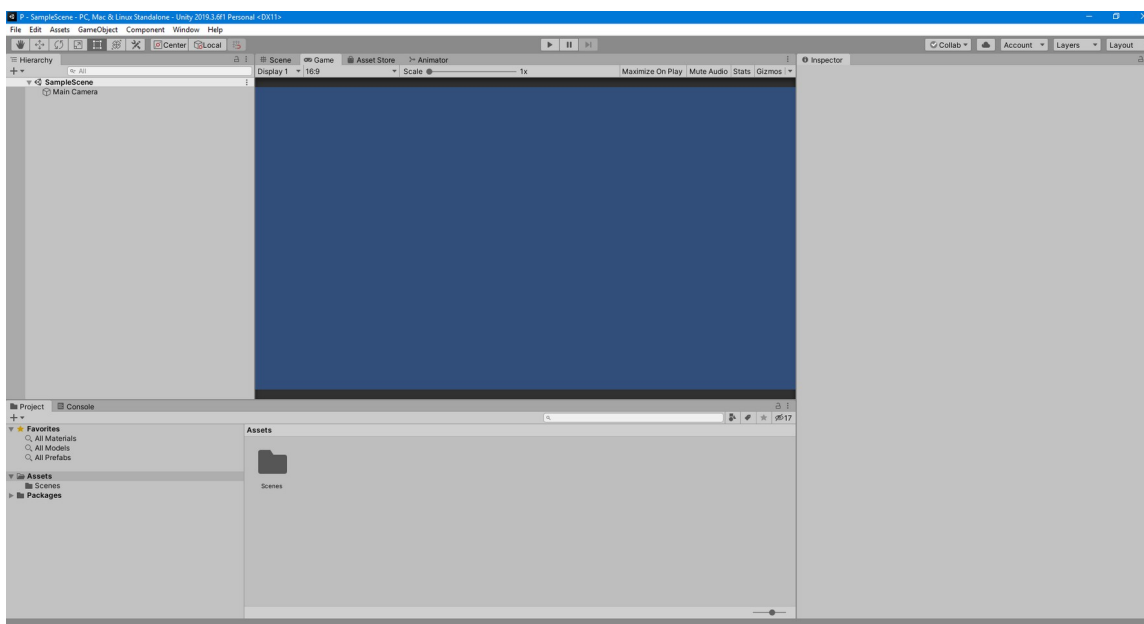


Рисунок 7 – Главное меню проекта

Для создания сцены меню, нажимаем правой кнопкой мыши по Scenes и создаем папку Menu в которой будут храниться файлы меню игры рисунок 8.

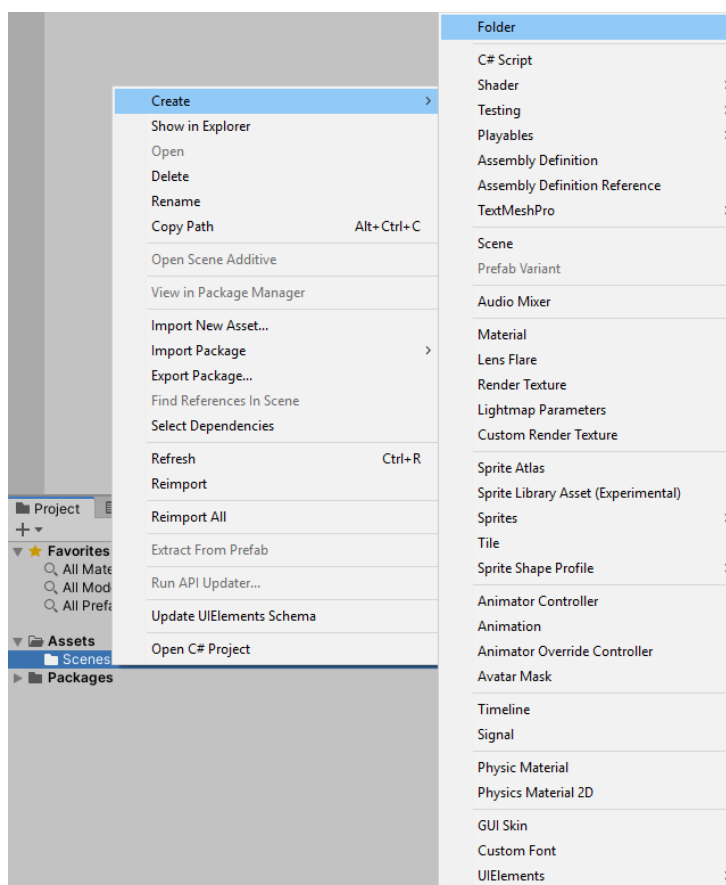


Рисунок 8 – Создание папки Menu

Далее нам нужно создать сцену, в которой будет отображаться наше меню, для этого нажимаем правой кнопкой мыши по пустому месту в нашей

новой созданной папке Menu, ищем пункт Create и в нём выбираем Scene и даём название меню рисунок 9.

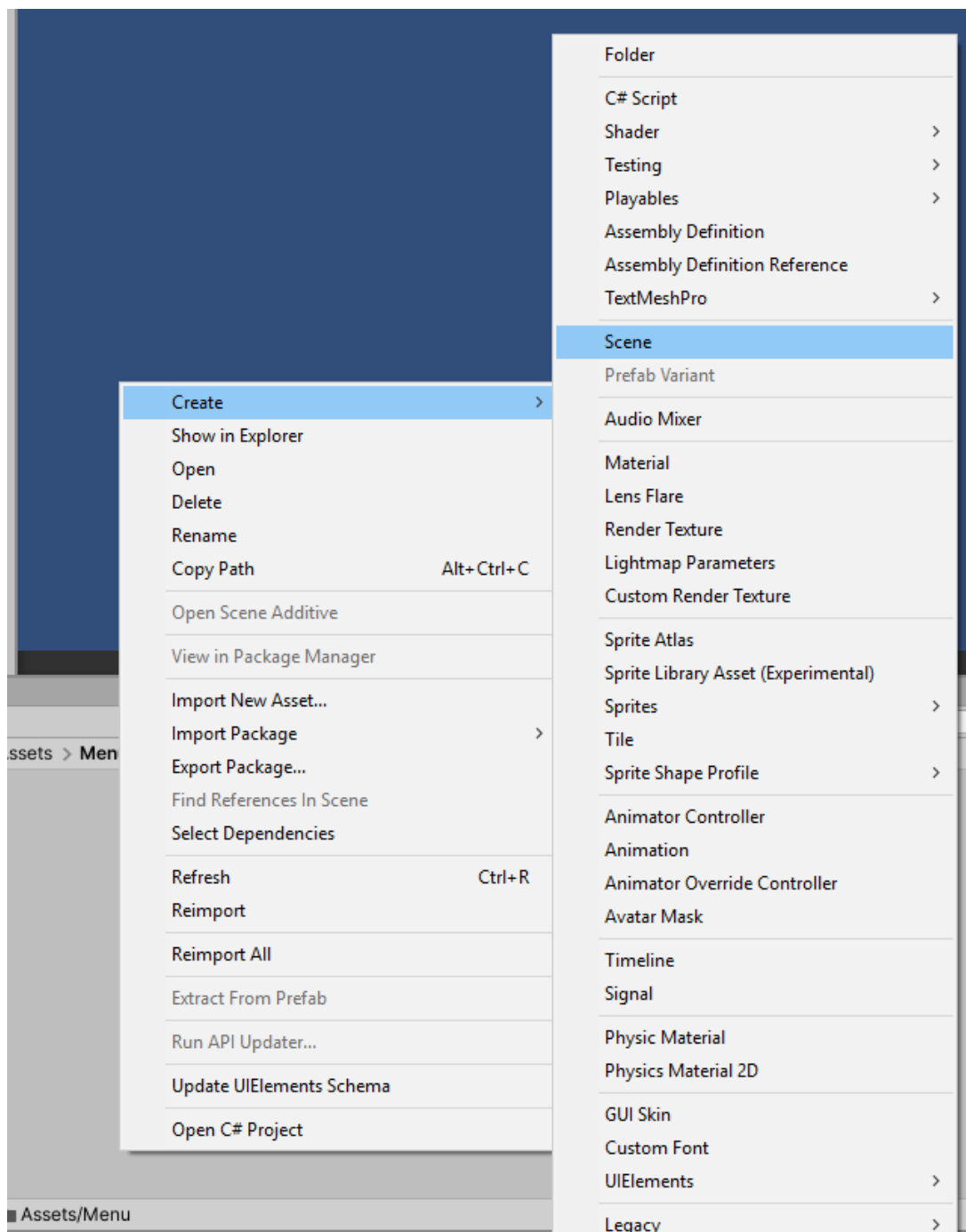


Рисунок 9 – Создание сцены меню

Далее нам необходимо создать Canvas в левой панели, для этого нажимаем левую кнопку мыши по панели и выбираем пункт UI и ищем Canvas и Panel и создаем эти два объекта рисунок 10.

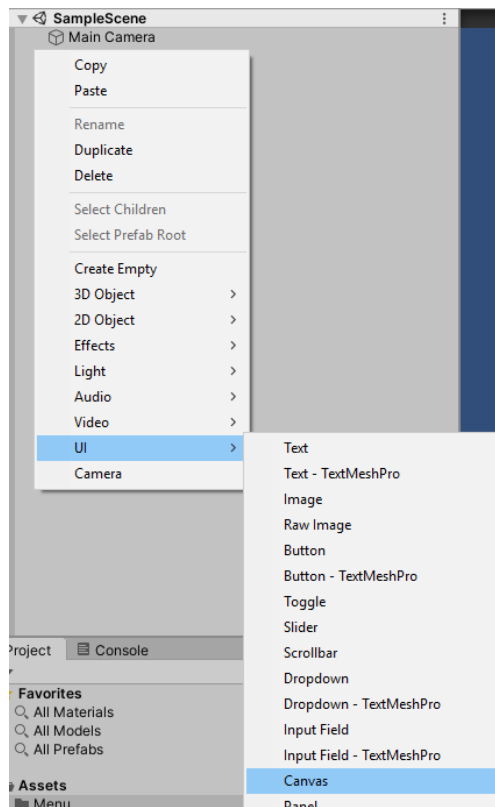


Рисунок 10 – Создание Canvas и Panel

Далее мы загружаем наш задний фон для меню в папку Menu, для этого просто его перетаскиваем в папку, далее мы открываем на левой панели объект Panel, и в правой части программы мы настраиваем наш задний фон и загружаем его в вкладку Source Image рисунок 11.

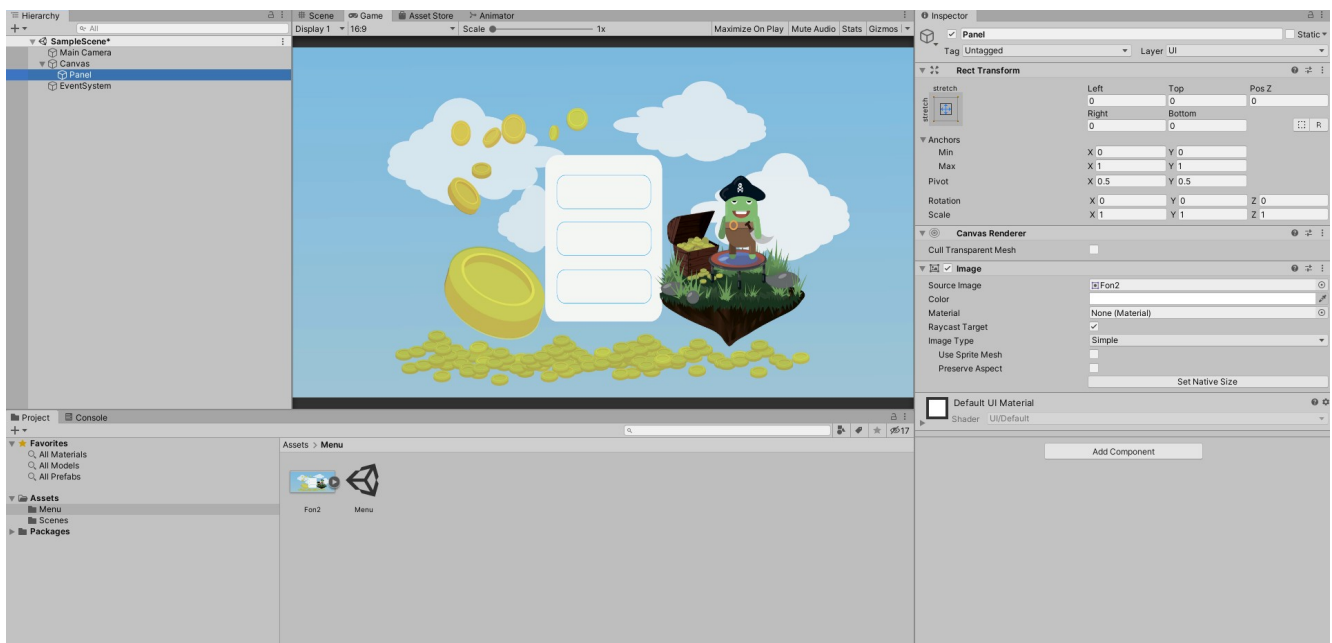


Рисунок 11 – Добавление заднего фона меню

Далее добавляем три кнопки для нашего меню, для этого нажимаем правой кнопкой мыши в левом меню и создаем button, для того чтобы они работали, необходимо создать скрипт, для этого мы создаем C# Script файл в нашей папки Menu рисунок 12.

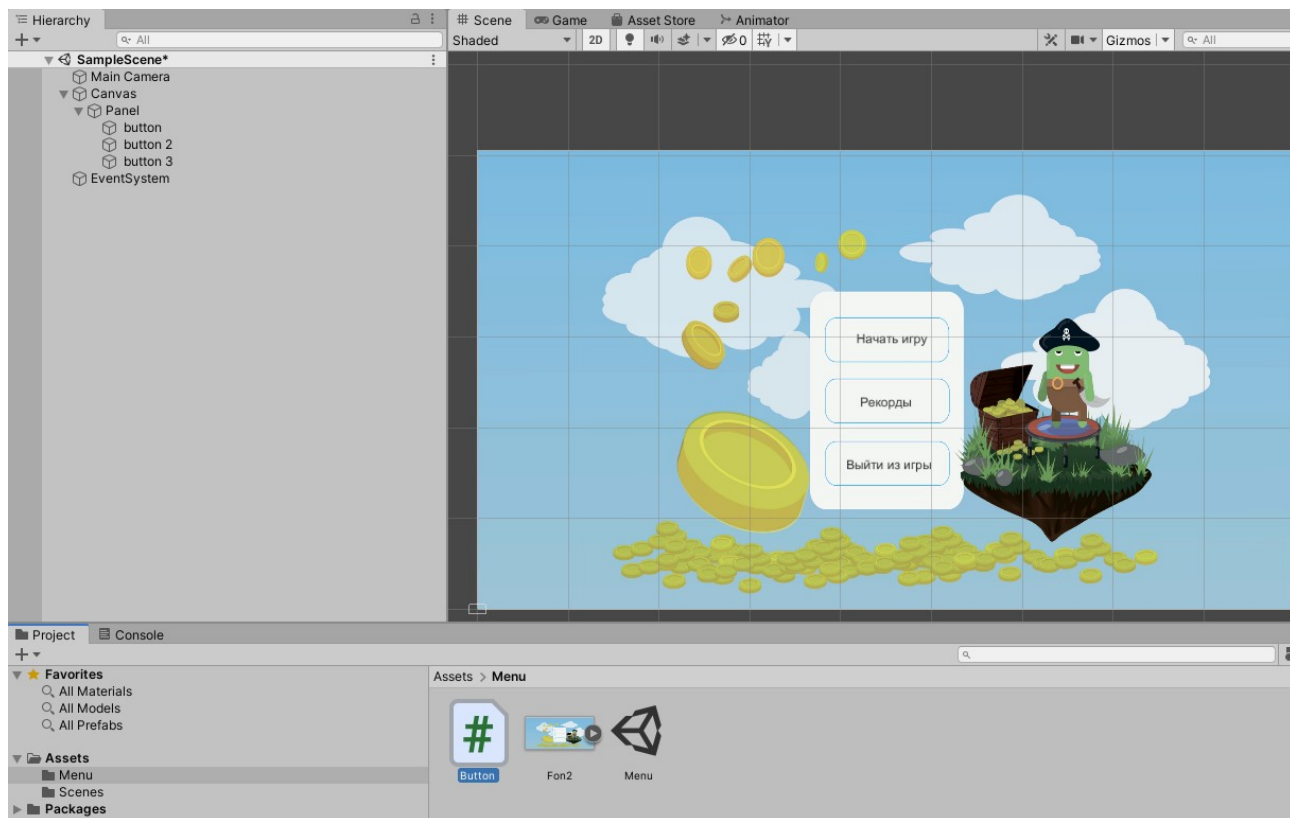


Рисунок 12 – Добавление кнопок и создание скрипта

Далее мы пишем скрипт для работы наших кнопок, и этот файл скрипта мы прикрепим к нашим кнопкам, для этого скрипт переносим на объект Main camera, и после этого нажимаем на нашу кнопку начать играть, присваиваем кнопке объект Main Camera и даём кнопке функцию, при нажатии на кнопку она открывает следующую сцену, и так проделываем с каждой кнопкой в нашем меню рисунок 13.

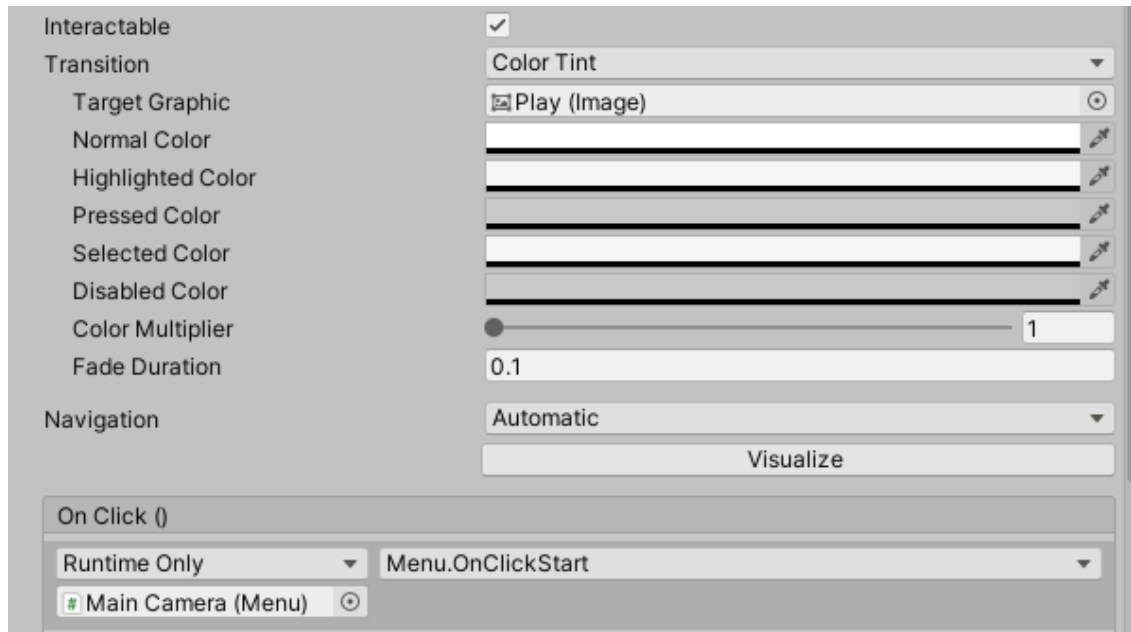


Рисунок 13 – Добавление кнопкам скрипт

Далее нам нужно создать папку и сцену авторизации и в них поместить наш задний фон, и так же создать объект Canvas, в котором будет наш задний фон и объект Panel, и для Panel создать кнопку и InputManager эти два объекта нужны будут для управление кнопкой подключения и формы авторизации, для работы кнопки нужно создать скрипт и прикрепить его к кнопке подключится рисунок 14,15.

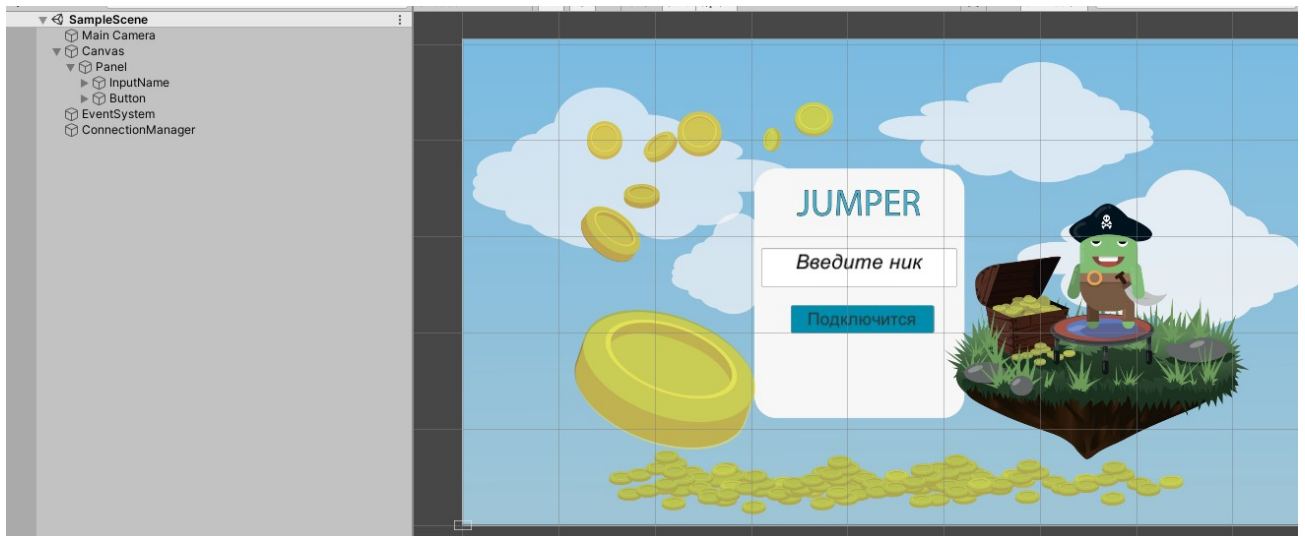


Рисунок 14 – Создание сцены авторизации

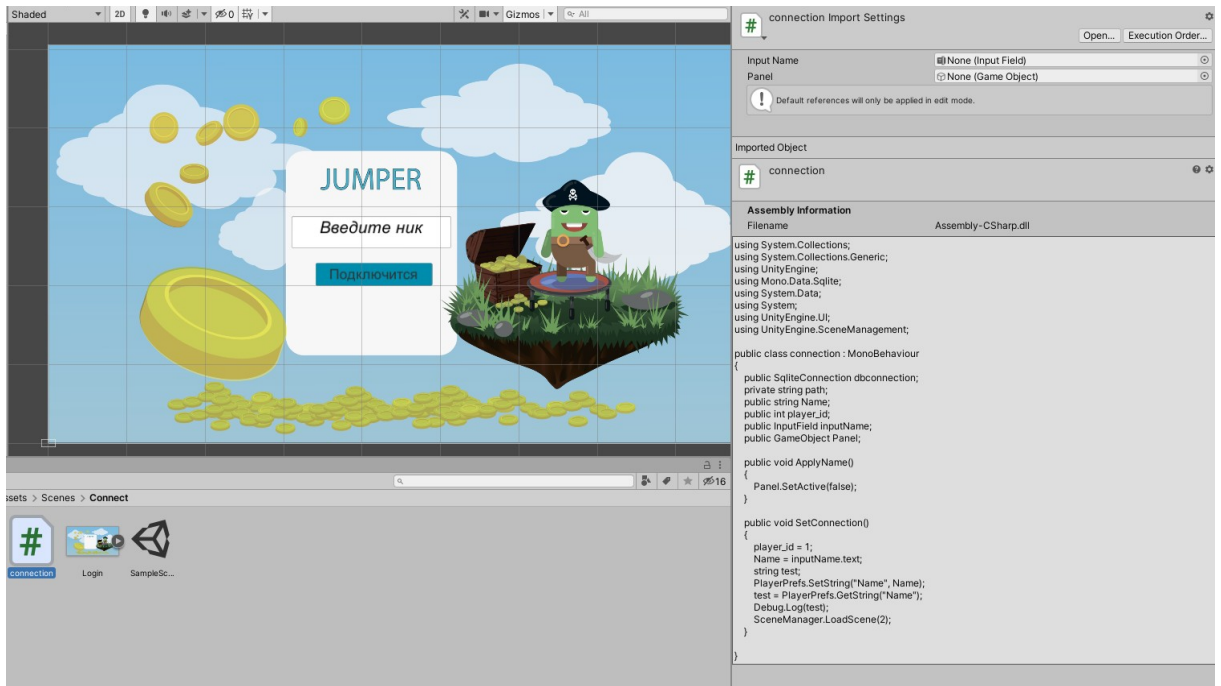


Рисунок 15 – Создание скрипта для работы подключения

Для того чтобы наше подключение работало, нужно открыть SQLite Studio, и в нем создать таблицу с данными, в которой будут храниться данные о игроках, для этого открываем программу и нажимаем создать таблицу и даём название Player для таблицы рисунок 16.

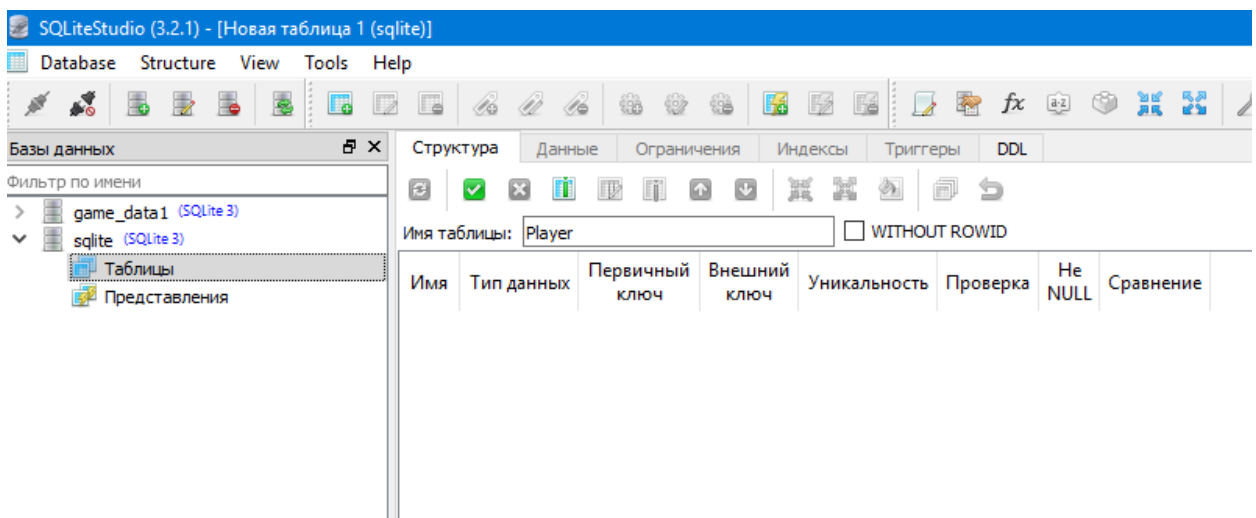


Рисунок 16 – Таблица Player

Далее нужно создать поля для нашей таблицы, для этого создаем столбцы, такие как ID_Player который будет отображать номер игрока, далее Name это имя игрока, а также Score в котором будут записываться количество очков, которые игрок заработал рисунок 17.

Структура									
Данные									
Ограничения									
Индексы									
Триггеры									
DDL									
Имя таблицы: <input type="text" value="Player"/> <input type="checkbox"/> WITHOUT ROWID									
Имя	Тип данных	Первичный ключ	Внешний ключ	Уникальность	Проверка	Не NULL	Сравнение		
1 ID_player	INTEGER								NULL
2 Name	TEXT								NULL
3 Score	INT								NULL

Рисунок 17 – Таблица Player

Заполненная таблица с игроками, готова для работы с игрой рисунок 18.

Табличный вид			
Форма			
ID_player	Name	Score	
1	2 Player2	434	
2	4 Player 5	343	
3	1 Player1	231	
4	5 Player4	231	
5	3 Player3	24	

Рисунок 18 – Заполненная таблица с игроками

После создание таблицы для нашей игры, нужно создать саму сцену с игрой, для этого создаём в папке Scenes файл Game и создаём такую же сцену и так же создаем Canvas рисунок 19.

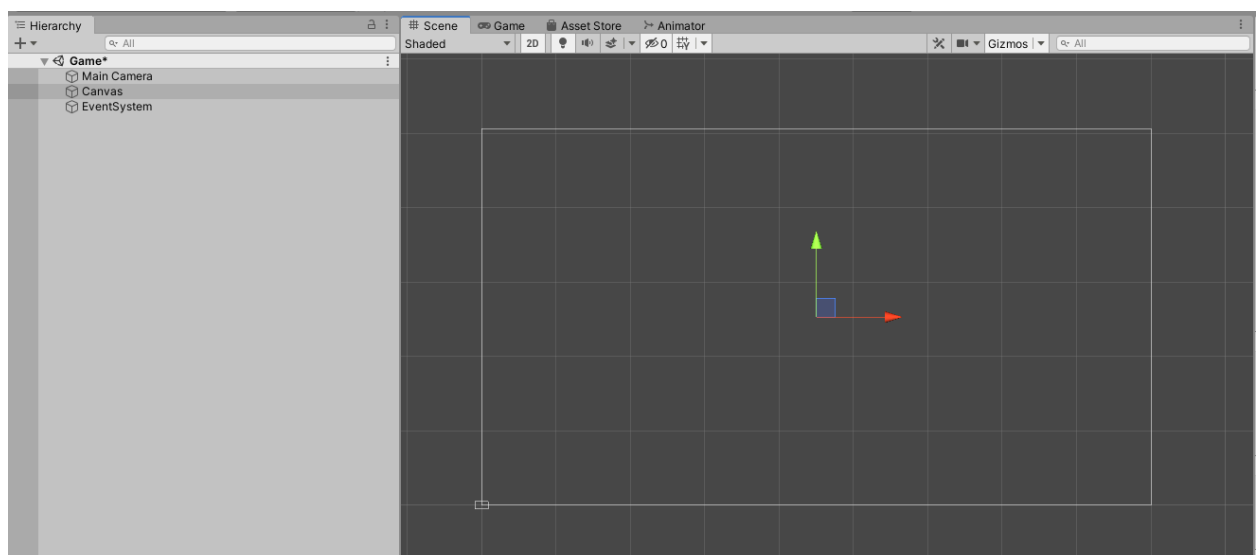


Рисунок 19 – Сцена игры

Далее нужно создать блоки, по которым персонаж будет прыгать, для этого нажимаем правой кнопкой мыши по левому меню и выбираем 2D object и выбираем вкладку Sprite и создаём его рисунок 20.

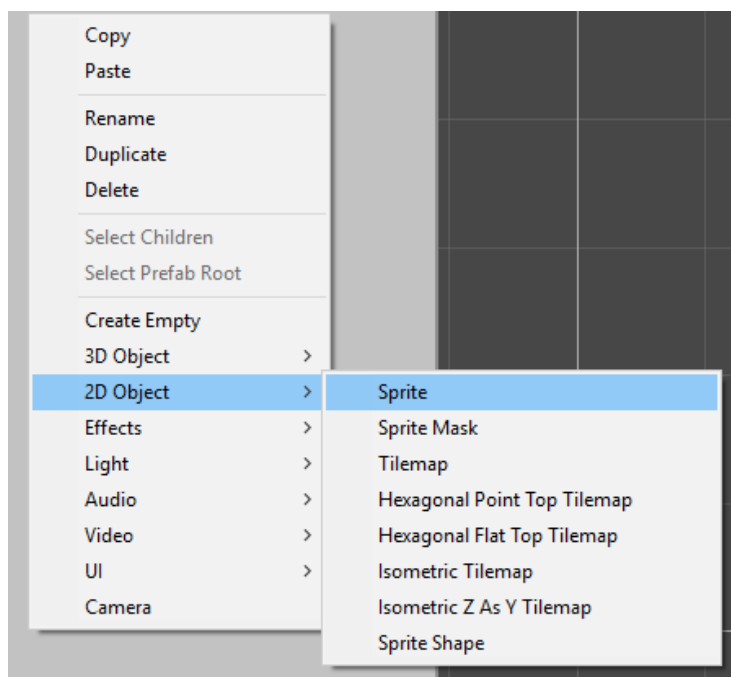


Рисунок 20 – Создание спрайта

Далее нужно настроить сам спрайт, для этого выберем его и в правом окне настраиваем его, нам нужно создать компоненты, для этого нажимаем на кнопку Add Component и создаем компонент Box Collider 2D для того, чтобы наш объект имел физические свойства, и наш персонаж мог по нему прыгать, далее создаем компонент Platform Effector 2D и в вкладке Sprite добавляем наш заранее подготовленный рисунок объекта, по которому персонаж будет прыгать рисунок 21.

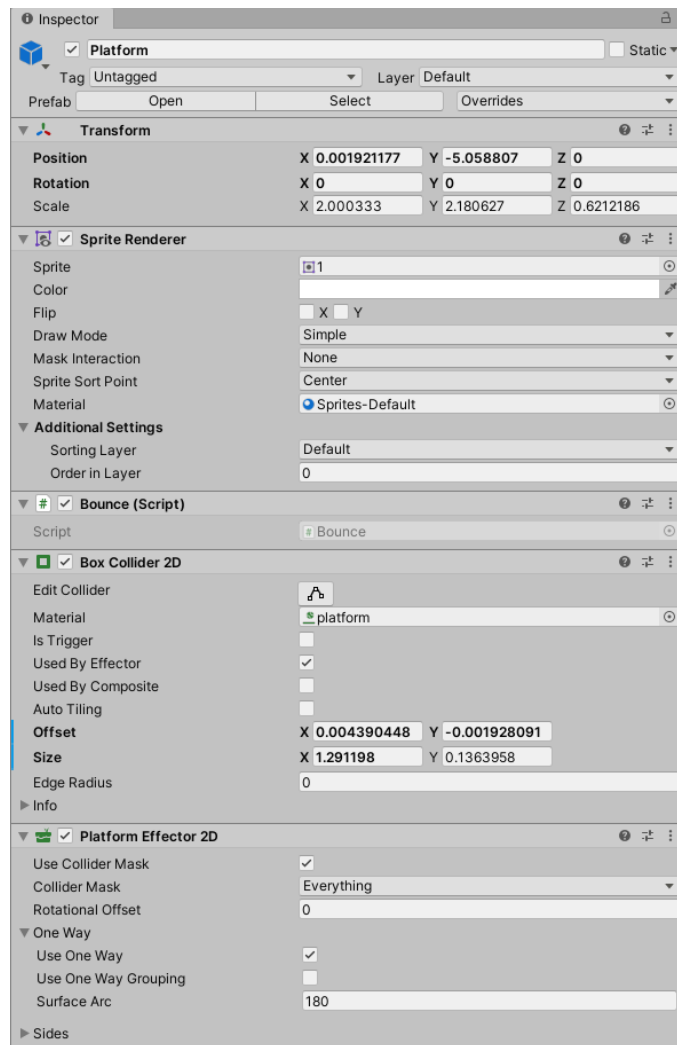


Рисунок 21 – Добавление компонентов

Далее после настройки, копируем наш объект и создаем ещё такие же объекты, и расставляем на сцене, как показано на рисунке 22.

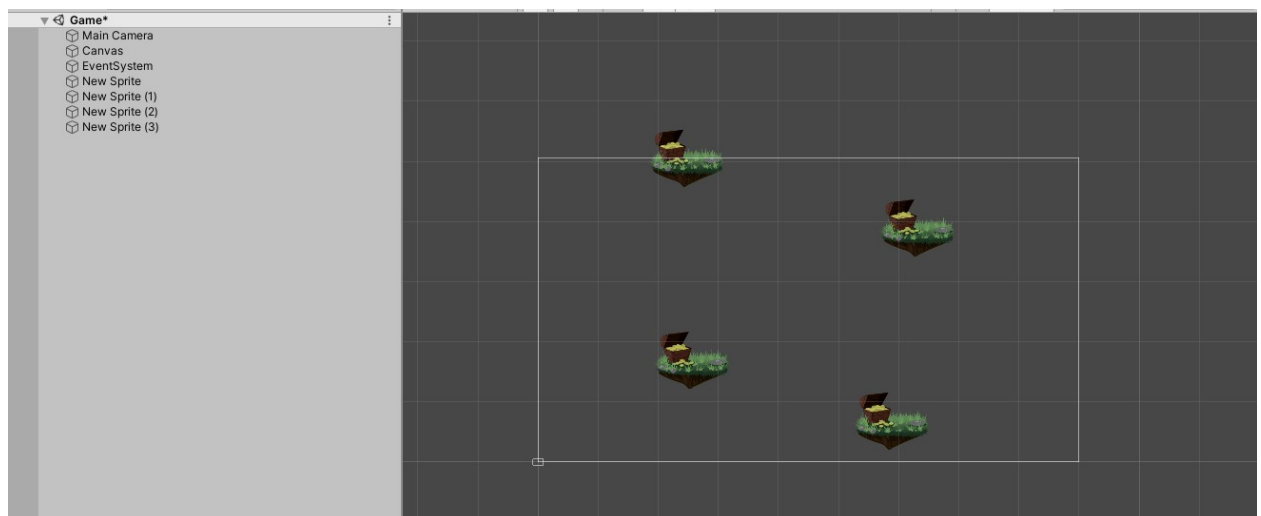


Рисунок 22 – Расстановка платформ

Далее нужно создать задний фон и самого персонажа на сцену, для этого

закидываем задний фон и персонажа в папку Game и переносим задний фон на левую сторону меню, и настраиваем его на сцене рисунок 23.

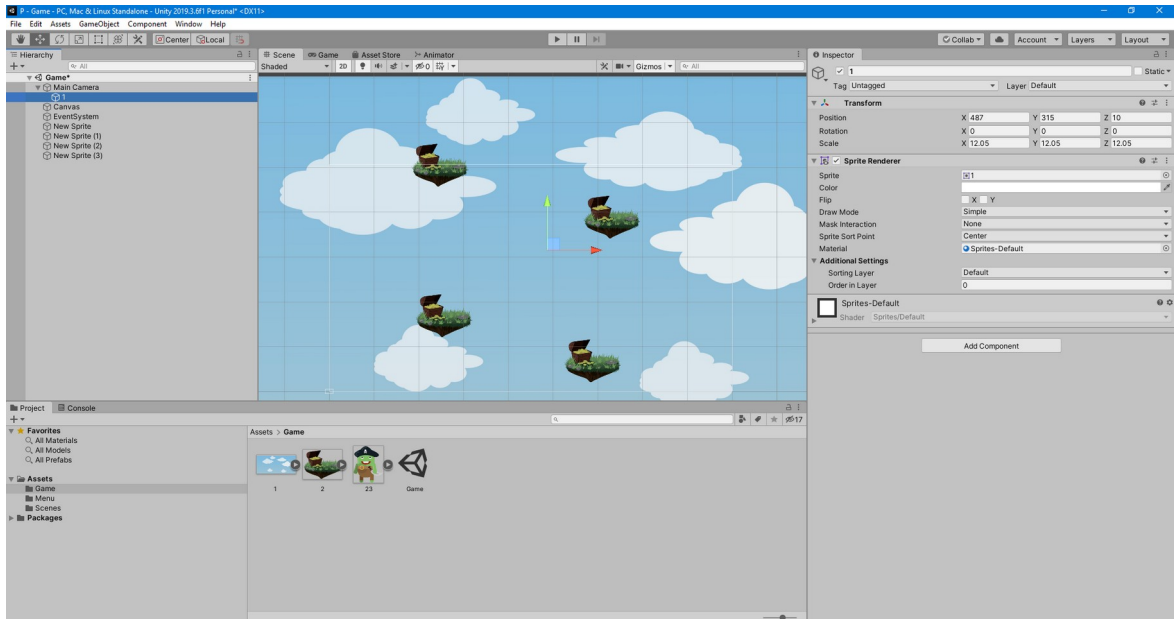


Рисунок 23 – Добавление заднего фона для игры

Далее создаём спрайт, это будет наш персонаж, для этого нажимаем на левую сторону меню правой кнопкой мыши и нажимаем 2D, выберем Sprite и дадим ему название Player, далее переносим изображение с персонажем на этот объект в кладку Sprite в правом меню и настроим его размер рисунок 24.

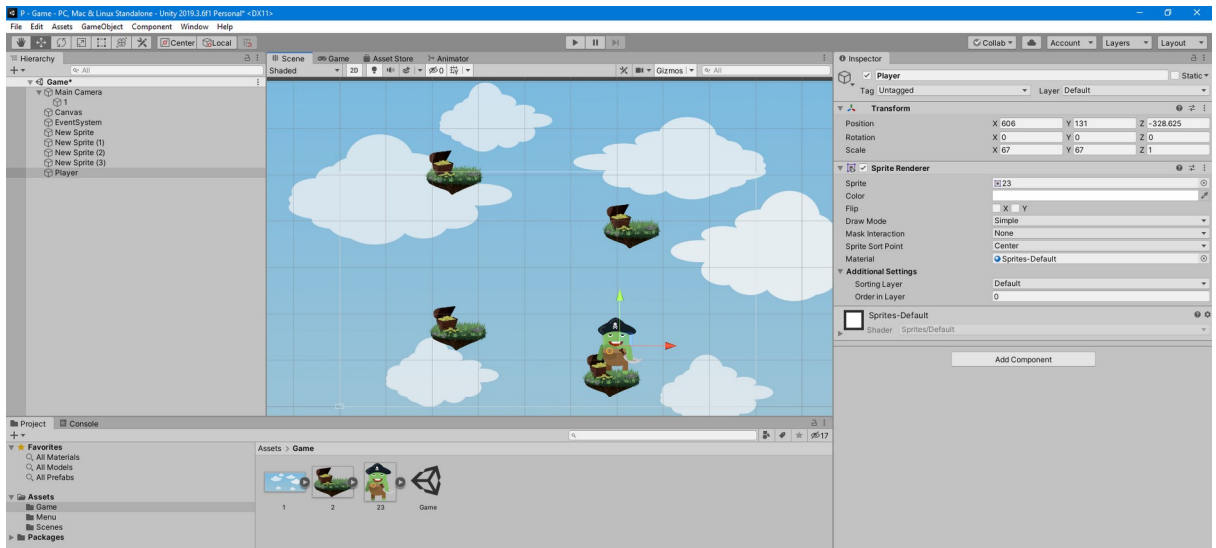


Рисунок 24 – Добавление персонажа на сцену

Теперь нужно персонажу дать физические свойства, для этого добавляем ему компоненты, такие как Box Collider, RigidBody 2D, Animator, и создаем скрипт для его передвижения, прикрепив его к компонентам рисунок 25.

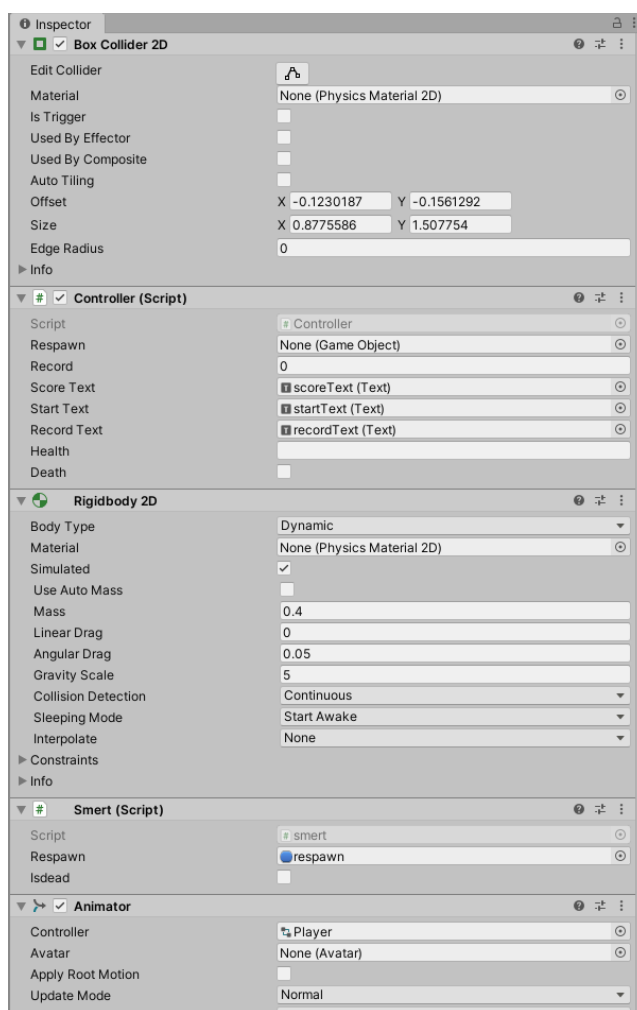


Рисунок 25 – Добавление персонажу физические свойства

Далее нужно создать зоны, за которые если персонаж выйдет, то мы проиграем и будем начинать заново уровень, для этого в левом меню нажимаем правой кнопкой мыши и создаём объект Sprite, присваиваем объекту компоненты, такие как Box collider 2D и скрипт со окончанием игры рисунок 26.

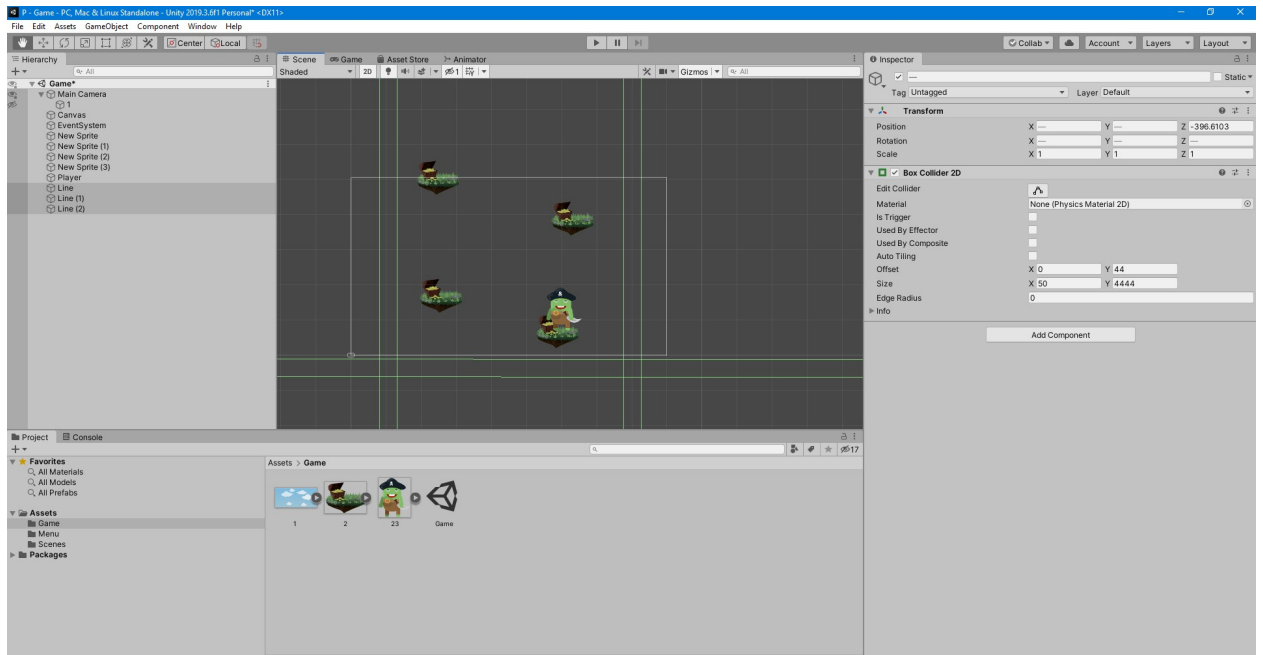


Рисунок 26 – Добавление зоны окончания игры

Создаем скрипт, при котором, если игрок попадёт в данные зелёные линии, то мы проиграем игру и начнём с заново, для этого нажимаем правой кнопкой мыши в папке Game и выбираем создать скрипт C# , после его написания, нужно его перенести в эти линии под компоненты, как показано на рисунке ниже рисунок 27.

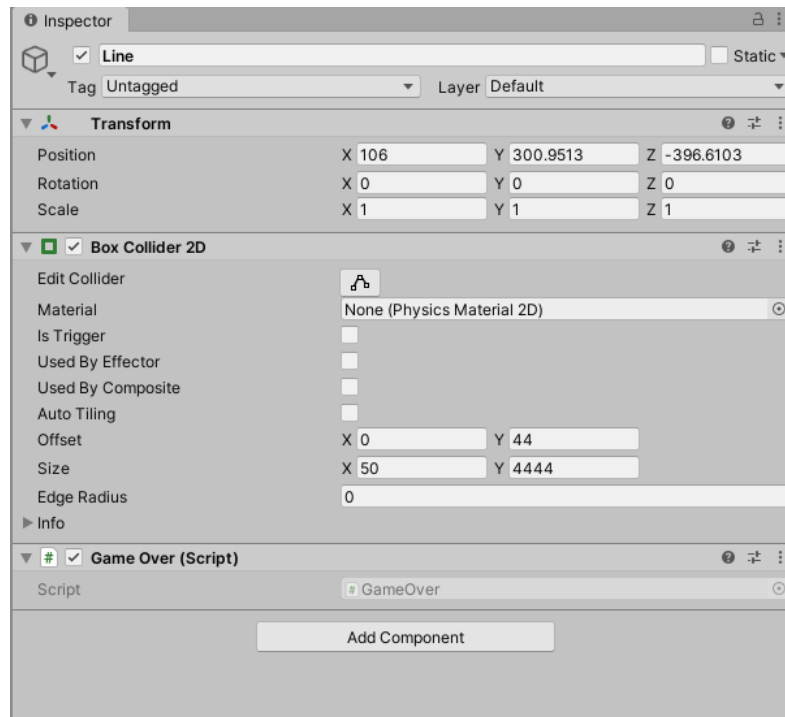


Рисунок 27 – Добавление скрипта для линий

Далее нужно создать текст по середине экрана для оповещения игрока о том, что, чтобы начать игру, нужно нажать любую клавишу на клавиатуре, для этого в левом меню нажимаем правой кнопкой мыши, ищем в разделе UI и там выбираем Text , далее открываем текст в меню, редактируем его в левой части и пишем, что для начала игры необходимо нажать кнопку, помещаем его по середине экрана для того, чтобы игрок увидел рисунок 28.



Рисунок 28 – Добавление надписи

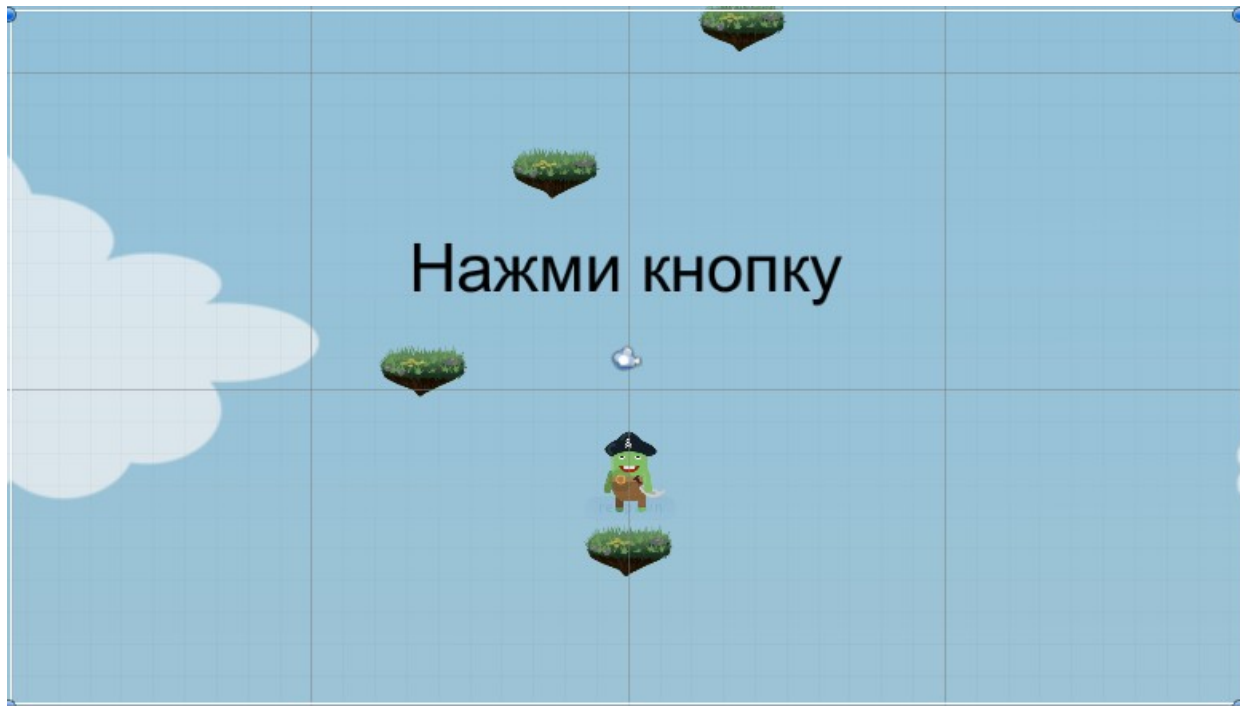


Рисунок 29 – Добавление надписи

Далее нужно создать две надписи, в которых будет показано сколько у нас сейчас очков, очки будут отображаться в монетах, и какой максимальной рекорд был поставлен, для этого нажимаем левой кнопкой мыши в левом меню и создаем два объекта с текстом, называем их Монет, Рекорд, и располагаем в двух углах монитора рисунок 29.

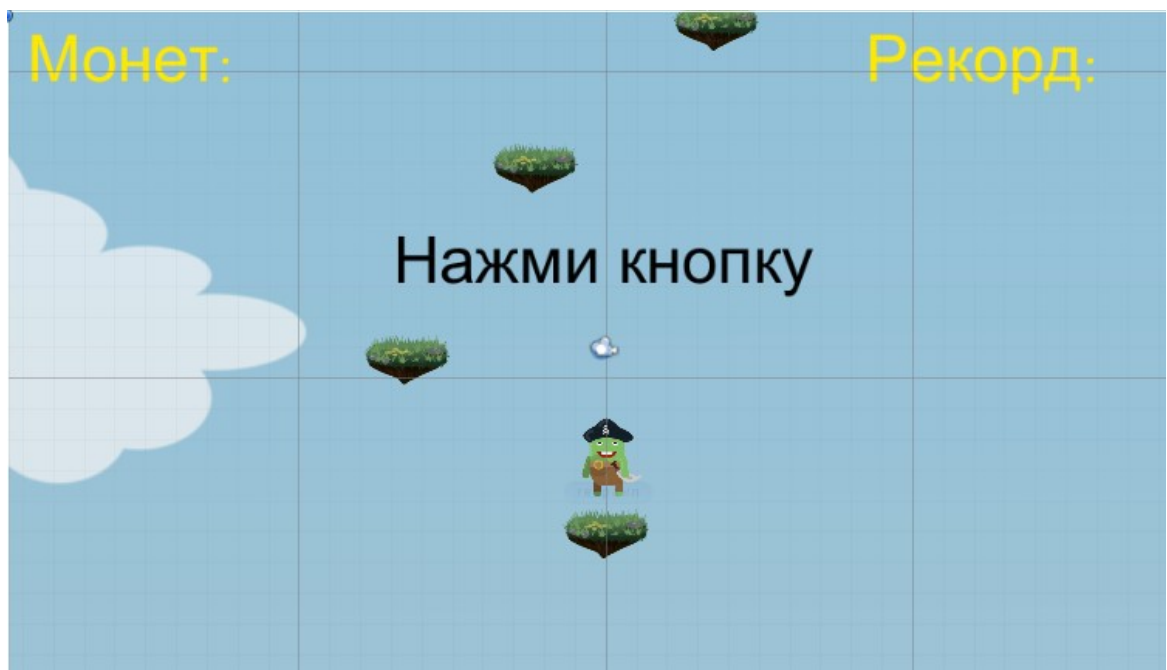


Рисунок 30 – Добавление надписи

Так выглядит основная сцена игры в Unity 5 рисунок 30.

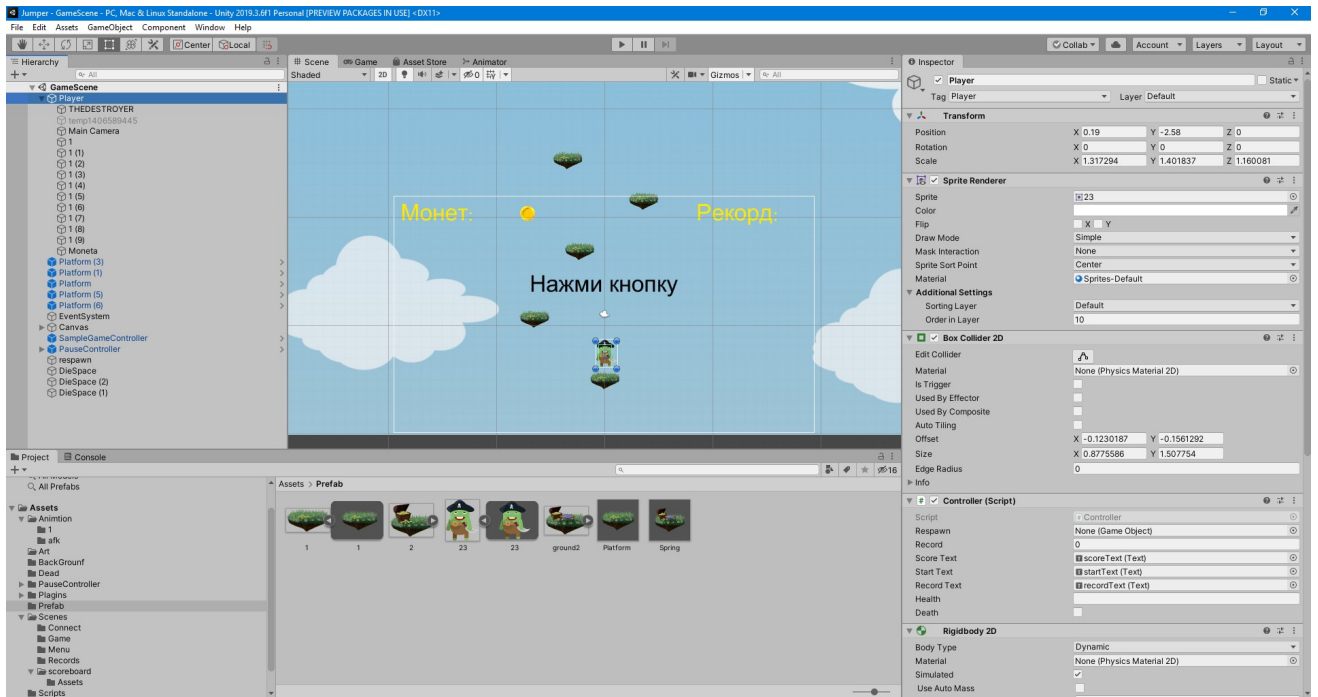


Рисунок 31 – Основная сцена игры

Далее нужно создать последнюю сцену, сцену рекордов, для этого создаем новую папку и сцену под название Records, в которую добавляем заранее подготовленные рисунки, которые необходимы для данной сцены и создаем три скрипта такие как Button, DataReader, UpdateScore рисунок 31.

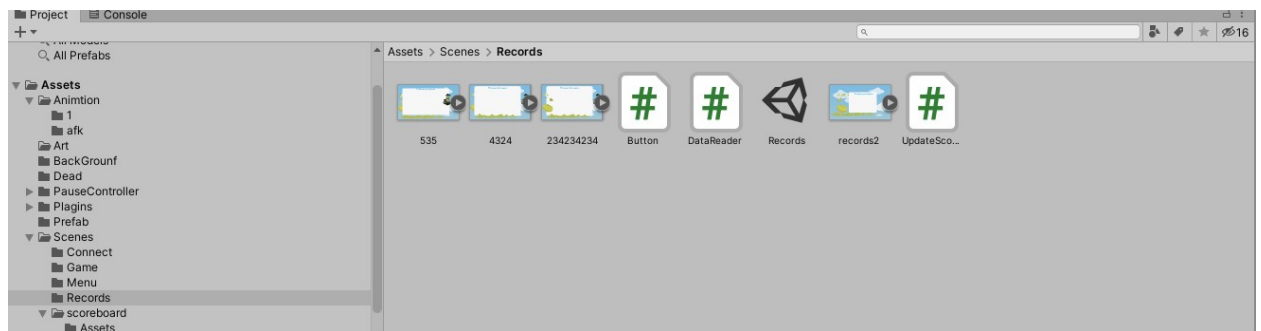


Рисунок 32 – Папка с файлами Records

Далее нужно создать Canvas, для этого нажимаем правой кнопкой мыши по левому меню, пункт UI и выбираем Canvas, которому присваиваем задний фон рисунок 33.

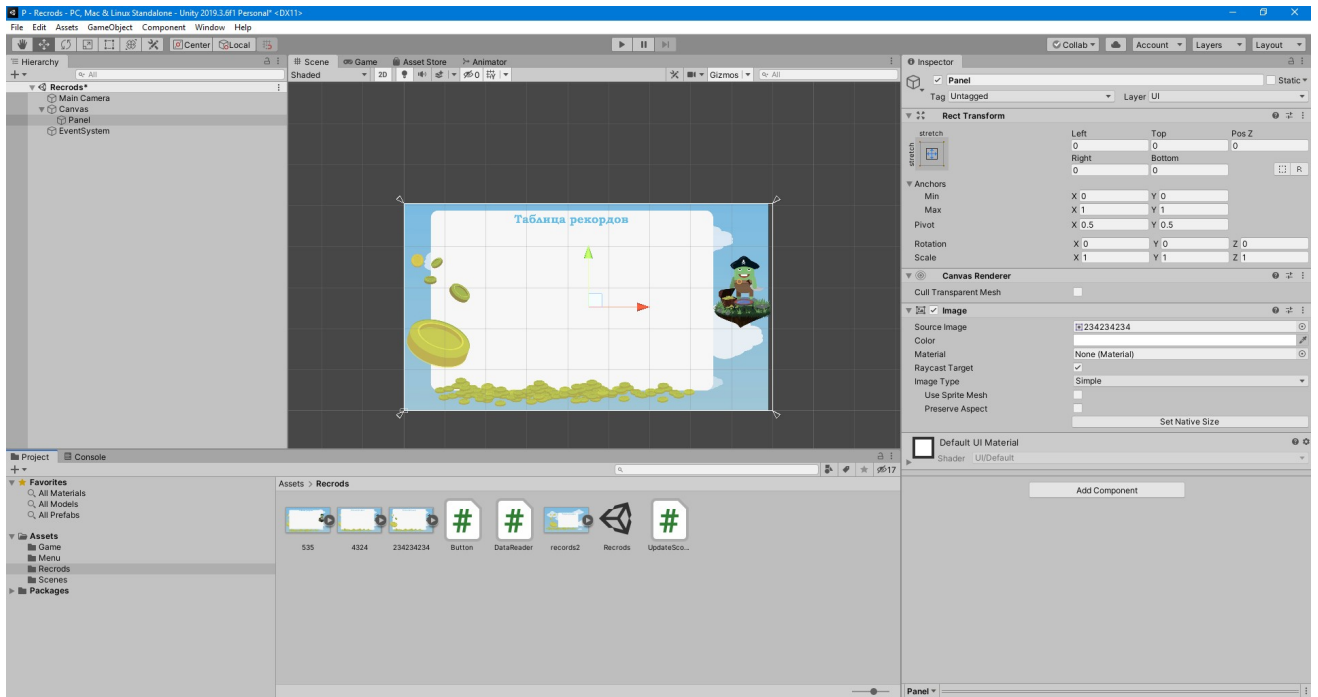


Рисунок 33 – Настройка заднего фона для сцены Records

Далее создаем объект Scroll View, для этого нажимаем правой кнопкой мыши на левое меню, выбираем пункт UI и ищем Scroll View, и так же добавляем три текста Место, имя, очки рисунок 34.

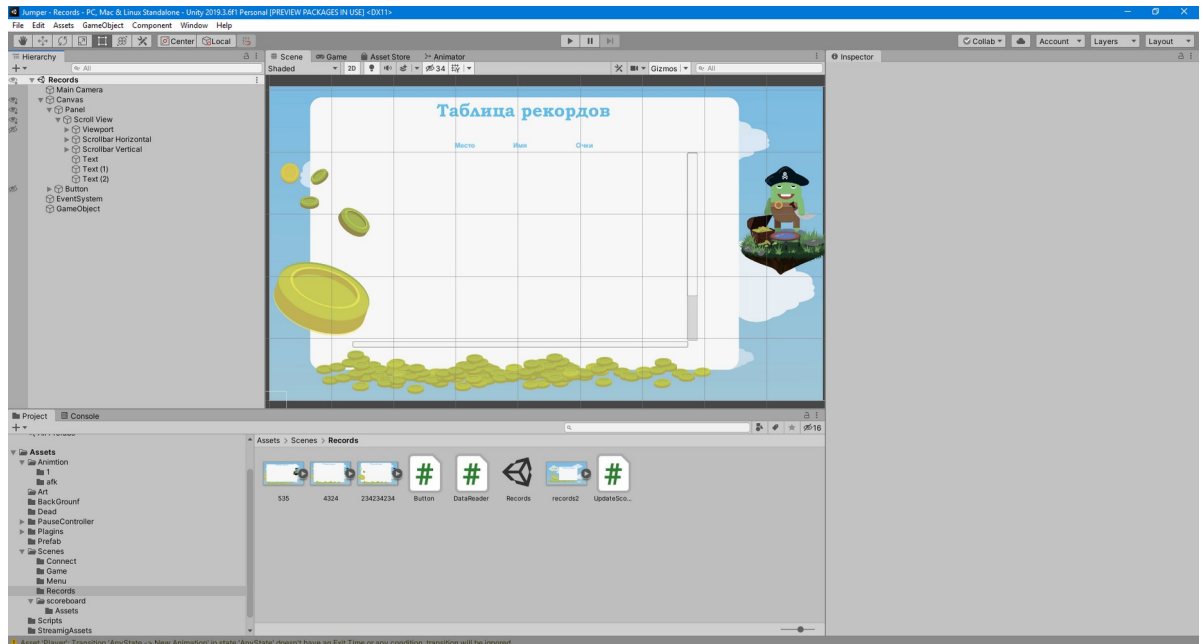


Рисунок 34 – Настройка таблицы Records

Далее создаем объект Viewport он отображает наши данные с таблицы SQLite, для этого нажимаем правой кнопкой мыши по левому меню, выбираем пункт UI и ищем Viewport и создаем, далее настраиваем его, нужно создать 10

строк с текстом и выставить как в таблице, в этом тексте будут писаться данные с нашей таблицы рисунок 35.

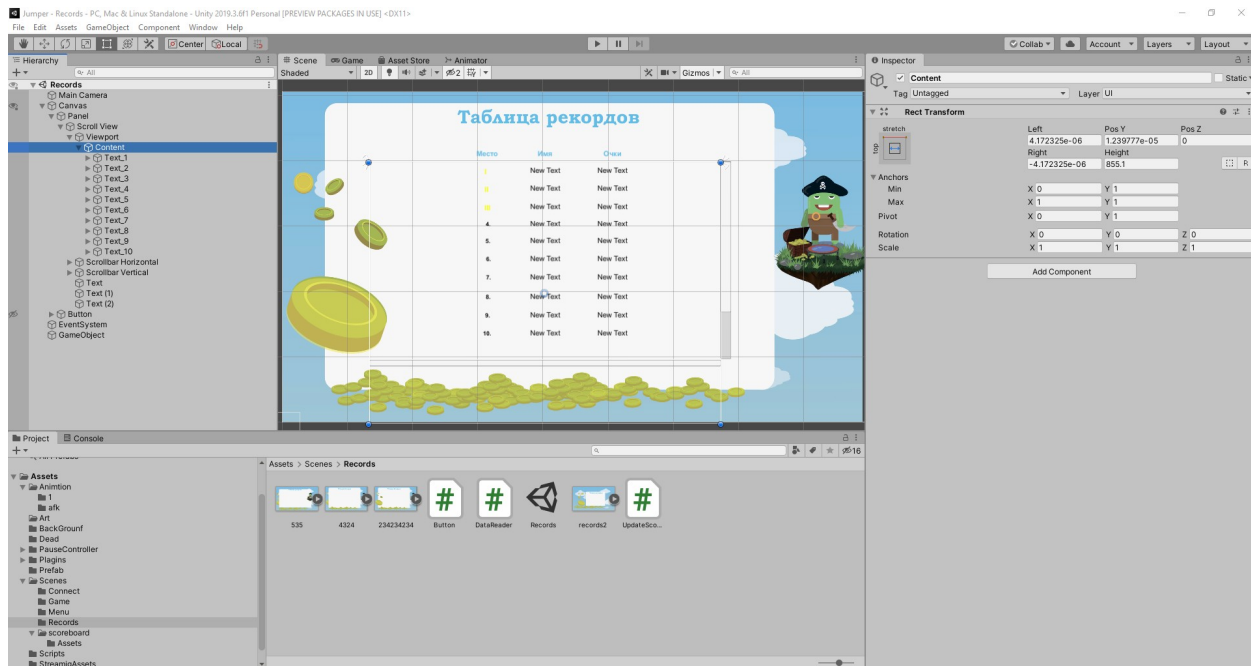


Рисунок 35 – Настройка таблицы Records

Далее нужно написать код для работы с нашей таблицей из SQLite и этот скрипт присвоить объекту Panel для того, чтобы наша таблица работала рисунок 36.

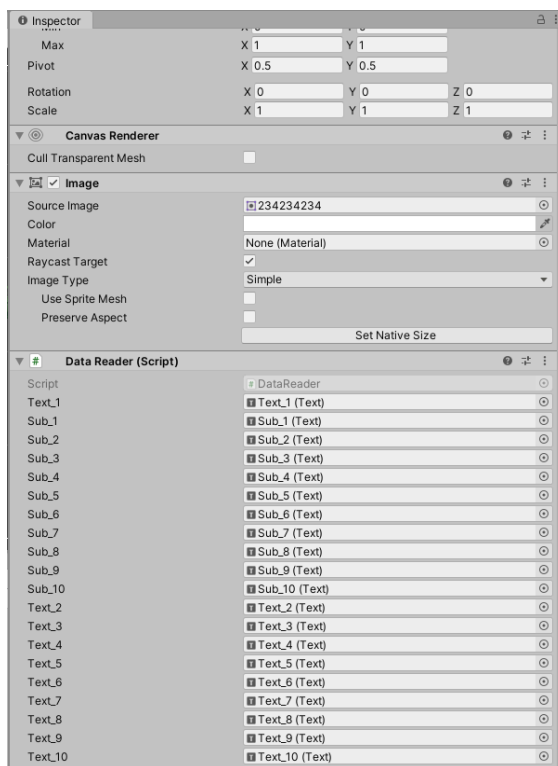


Рисунок 36 – Настройка объекта Panel

Далее нужно создать кнопку для выхода обратно в главное меню, для этого нажимаем правой кнопкой мыши по левому меню, пункт UI, ищем Button и создаем, пишем код для работы кнопки и прикрепляем его на кнопку рисунок 37.

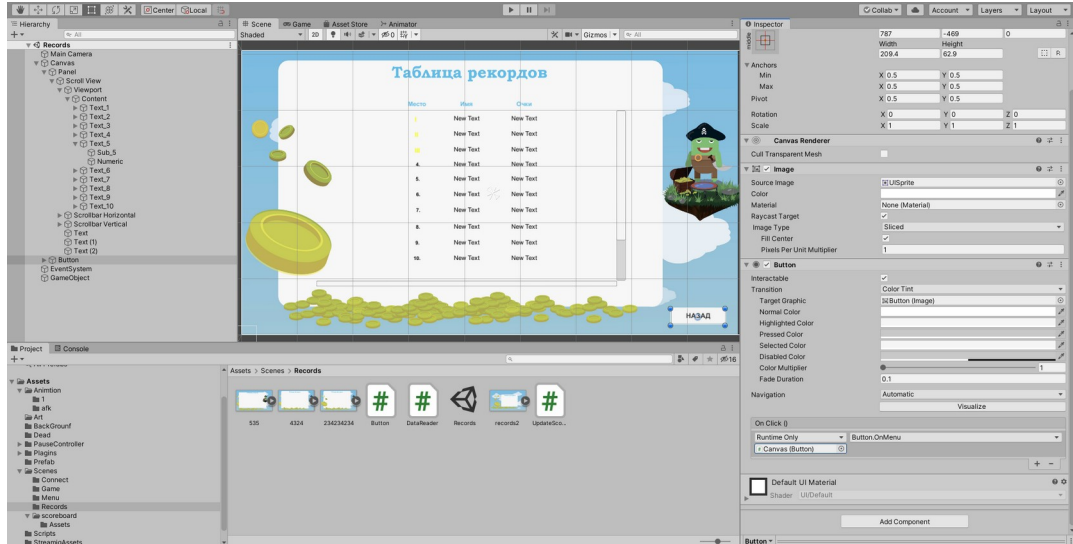


Рисунок 37 – Настройка кнопки в главное меню

Запускаем игру в самом движке и видим, что наша таблица успешно отображается и готова к работе рисунок 38.

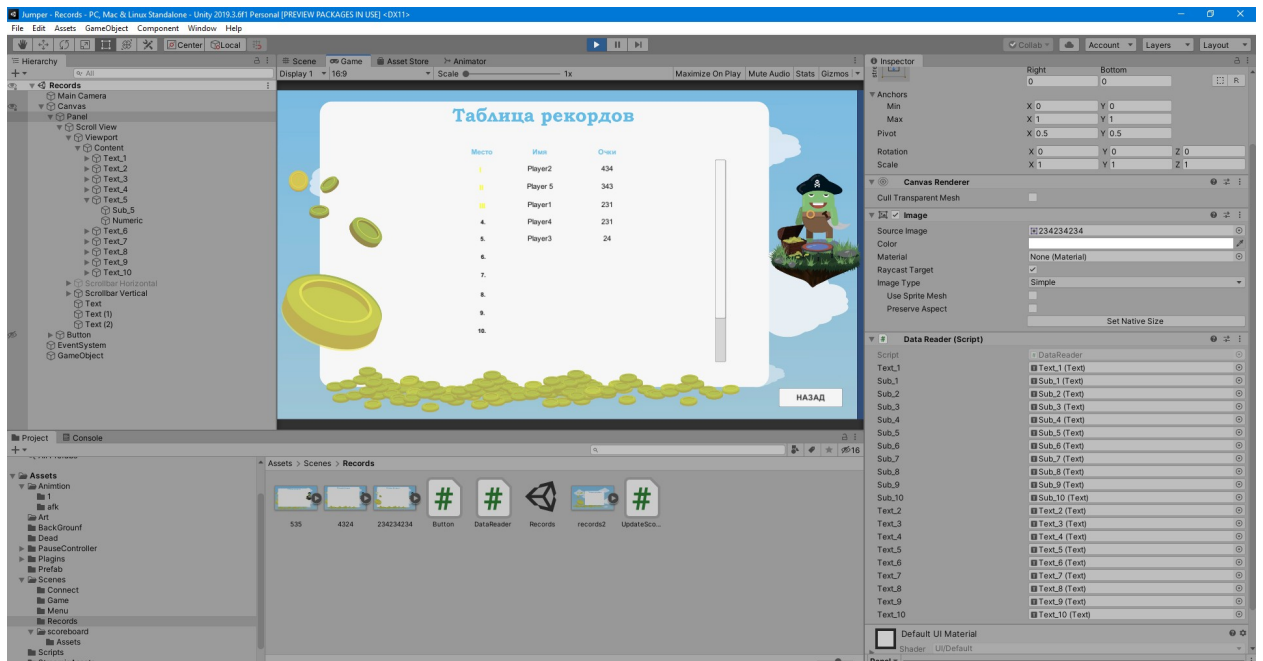


Рисунок 38 – Таблица рекордов

На рисунке рисунок 39, показана главная сцена меню игрового приложения.

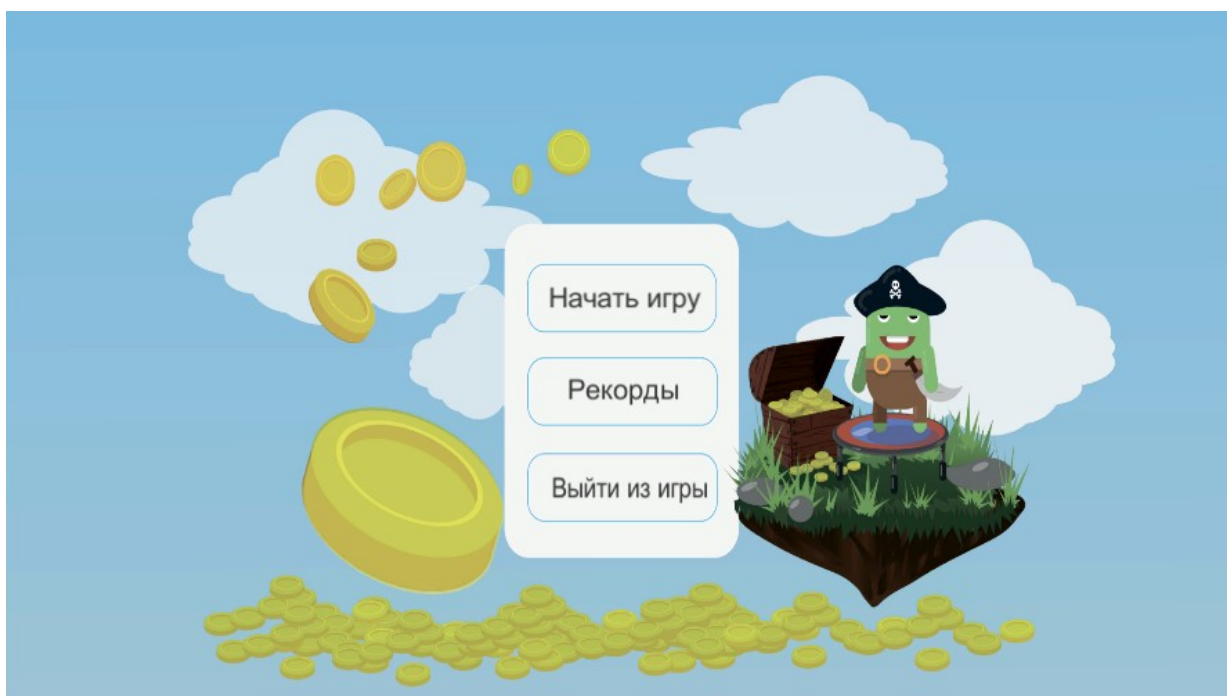


Рисунок 39 – Сцена меню

Рисунок 40 демонстрирует сцену авторизации для игрока.

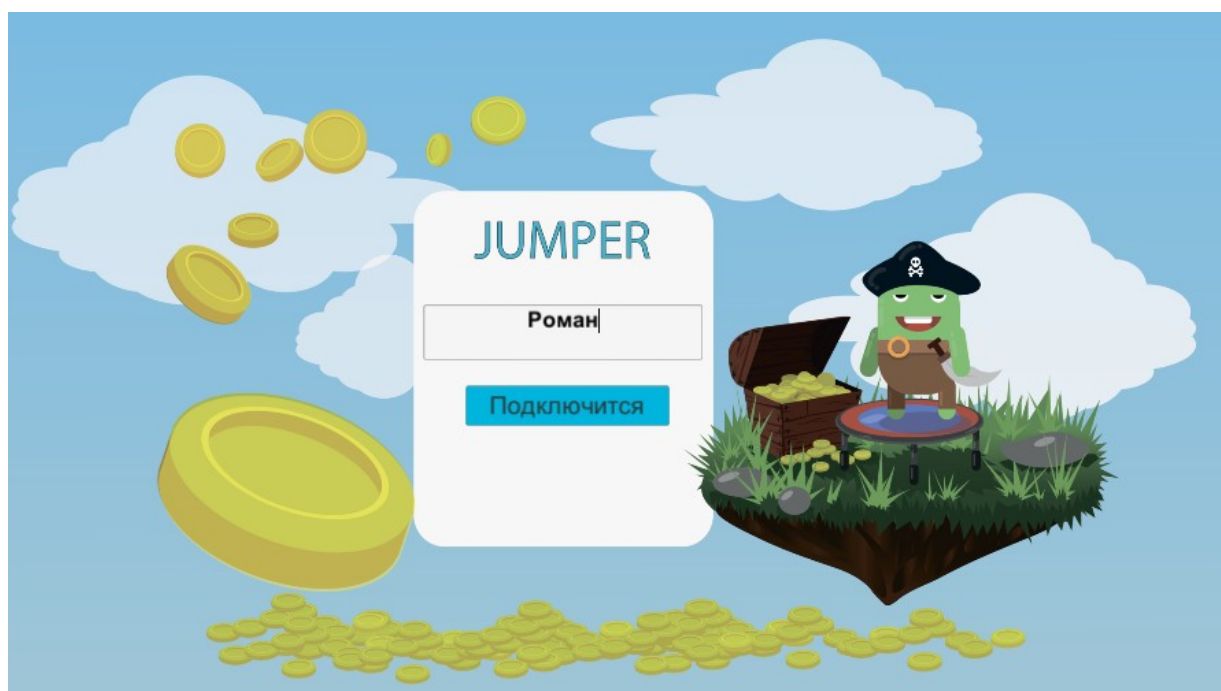


Рисунок 40 – Сцена авторизация

После введенных данных пользователем мы можем увидеть игровое поле, на котором игрок управляет героем (Рисунок 41).

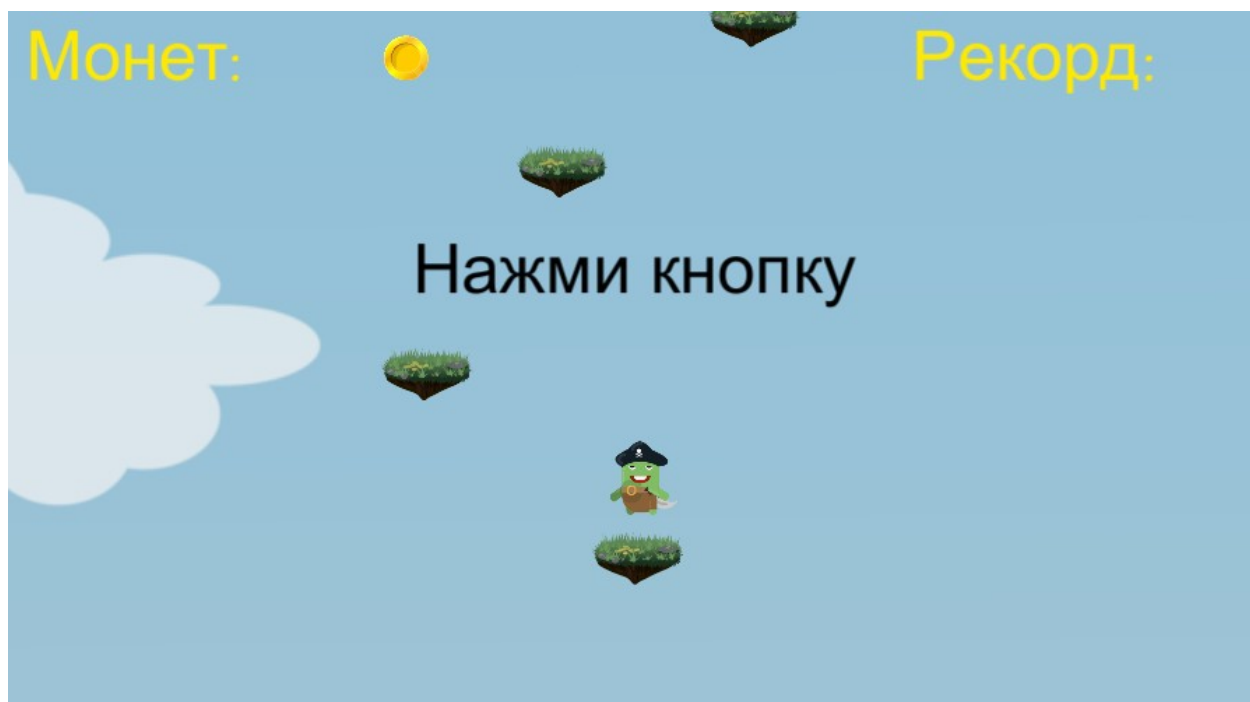


Рисунок 41 – Сцена игра

Когда игра завершается на дисплеи экрана показывается сцена таблицы рекордов, как показано на рисунке 42.

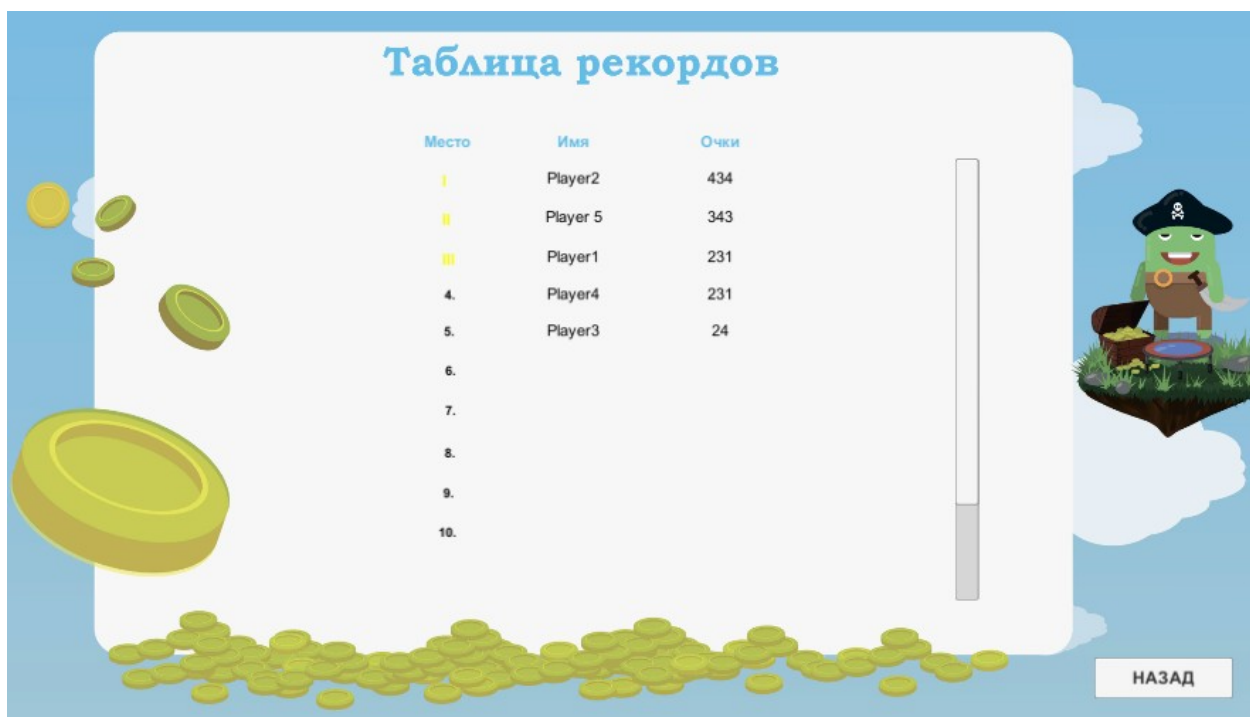


Рисунок 42 – Сцена таблица рекордов

Во второй главе выполнено проектирование средствами диаграмм, которые показывают игровую логику 2D игры. Так же представлена поэтапная разработка в среде «Unity 5»

ЗАКЛЮЧЕНИЕ

В данном проекте было необходимо создать 2D игру на игровом движке «Unity 5», актуальность которой определена тем, что игровая индустрия очень быстро набирает обороты во всем мире, и приносит большие доходы как разработчикам, так и государству в виде налогов, и так же игры пользуются очень большим спросом во всем мире, любого возраста. В данном проекте было создано 2D игра на платформе «Unity 5», она была выбрана, потому что является бесплатной и проста в процессе разработке и не требует больших финансовых ресурсов.

В первой главе был проведён анализ предметной области и было произведено описание технологий и методов реализации программного продукта. Где рассказывается почему была выбрана платформа «Unity 5», какие ещё плюсы и минусы, так же было выбрано на каком языке программирование был написан код, и так же была выбрана СУБД SQLite Studio. Так же был произведен анализ системных требований для платформы «Unity 5».

Во второй главе было рассмотрено проектирование диаграммой игровой логики для 2D игры на «Unity 5». В данной главе было рассмотрена игровая логика данного проекта, как работают сцены между собой, и были нарисованы все сцены проекта.

Во втором пункте второй главы была разработка самой 2D игры на игровом движке «Unity 5». Где было показано как создавался весь проекта с нуля по этапам, и в конце была показан результат.

Таким образом, мы получили программный продукт 2D игру, обладающую игровой логикой, где пользователь может авторизоваться, указывая Ник, после чего переходит на игровое поле, на котором при помощи клавиши пробела заставляет героя прыгать вверх и стрелками клавиатуры задавать траекторию полета, при касании с объектом бонуса «Сундук» происходит активация двойного перемещения вверх. Игровой процесс завершается при попадании игрока в пропасть при этом появляется таблица

рекордов с рейтингом очков, набранных в ходе игры. Поэтому можно с уверенностью сказать, все поставленные задачи выполнены.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Бонд Джереми Гибсон. Unity и C#. Геймдев от идеи до реализации, 2019. .- 470 с.
2. Вагнер Билл. Наиболее эффективное программирование на C#. 50 способов улучшения кода, 2017. .- 190 с.
3. Васильев Алексей Николаевич. Программирование на C# для начинающих. Особенности языка, 2018. .- 130 с.
4. Гейг Майк. Разработка игр на Unity 2018 за 24 часа, 2018. .- 250 с.
5. Герберт Шилдт. Программирование книга по C#, 2019. .- 310 с.
6. Гибсон Бонд Гибсон Бонд Джереми. Unity и C#. Геймдев от идеи до реализации, 2019. .- 480 с.
7. Джефффри Рихтер. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#, 2018. .- 730 с.
8. Дикинсон Крис. Оптимизация игр в Unity 5, 2017. .- 750 с.
9. Торн Алан. Основы анимации в Unity, 2019.- 430с.
10. Троелсен Эндрю, Джепикс Филипп. Язык программирования C# 7 и платформы .NET и .NET Core, 2018. .- 220 с.
11. Тучкевич Евгения Ивановна. Самоучитель Adobe Illustrator CS 2018, 2018. .- 560 с.
12. Хокинг Джозеф. Unity в действии. Мультиплатформенная разработка на C#, 2018. .- 610 с.
13. Шелл Джесси. Геймдизайн. Как создать игру, в которую будут играть все, 2020. .- 110 с
14. Jennifer Smith, AGI Team Creative. Adobe Illustrator CS6 Digital Classroom, 2019. .- 170 с.
15. Will Goldstone. Основы разработки игр на Unity, 2018.- 150 с.
16. Документы. Знакомство с документами // <https://er.nau.edu.ua/> Руководство по SQLite. URL: [//er.nau.edu.ua/bitstream/NAU/10100/6/SQLite.Allow.0.90.pdf/](https://er.nau.edu.ua/bitstream/NAU/10100/6/SQLite.Allow.0.90.pdf) (02.05.2020).

17. Документы. Знакомство с документами // Unity5.com/ Научитесь программированию в Unity. URL: <https://unity.com/ru/how-to/beginner-gaming-coding-resources> (15.03.2020).

18. Документы. Знакомство с документами // Unity5.com/ Начало программирования Unity. URL: <https://unity.com/ru/learn/get-started> (04.05.2020).

19. Документы. Знакомство с документами // Unity5.com/ Как пользоваться Unity. URL: <https://docs.unity3d.com/Manual/index.html/> (05.05.2020).

20. Справочники. Знакомство со справочниками // unity3d.fandom.com / Wiki Unity5. URL: <https://unity3d.fandom.com/ru/wiki/> (12.05.2020).