



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ
ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Экономики и управления на предприятии природопользования»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(бакалаврская работа)
по направлению подготовки 09.03.03 Прикладная информатика
(квалификация – бакалавр)

На тему «Проектирование компьютерной игры в жанре Point&Click»

Исполнитель Саакян Давид Ашотович

Руководитель к.т.н. Попов Николай Николаевич

«К защите допускаю»

и.о.заведующий кафедрой _____

кандидат экономических наук, доцент

Шутов Василий Васильевич

« ____ » _____ 2023 г.

Туапсе
2023

ОГЛАВЛЕНИЕ

Введение.....	3
1 Анализ объекта и предмета исследования.....	6
1.1 Общее представление о разработке игр.....	6
1.2 Сравнительный анализ игр жанра «Point & Click».....	9
1.3 Общий алгоритм реализации проекта.....	15
2 Программные средства и технологии создания игр	17
2.1 Анализ игровых движков	17
2.2 Анализ графических редакторов	27
3 Проектирование «2D платформера жанра Point & Click»	29
3.1 Требования к игре	29
3.2 Характеристика оборудования разработки	30
3.3 Проектирование графического оформления	31
3.4 Проектирование сюжета игры	51
Заключение	53
Список литературы	55

Введение

Разработка игр – это очень интересное и прогрессивное направление в области ИТ. Ведь благодаря игровой индустрии многие специалисты из различных областей могут объединиться, для создания одного, масштабного проекта. Благодаря игровой индустрии, различные производители компьютерного железа каждый год выпускают всё более мощное оборудование, что позже эволюционируют в аппаратные средства для серверных оборудований или мейнфреймы [19, с.221]. На текущий момент, общество неразрывно связано с системами геоинформационных ресурсов или социальными сетями, мессенджерами, системами получения государственных услуг и т.д. Все ранее перечисленное существенно зависит от мощностей аппаратной составляющей серверного оборудования. С каждым годом, компьютерное железо становится всё более отказоустойчивее и производительнее, что положительно сказывается для пользовательского опыта Систем типа «клиент – сервер».

Закончив лирическое отступление и вернувшись к тебе проектирования игры, можно отметить, что современные игры, для различных платформ, необычайно сложны. Особенно, самый сложный и ответственный этап разработки – является именно планирование проекта. Ведь, проектирование включает в себя анализ рынка компьютерной и игровой индустрии, отбор оптимального игрового движка, разработку сюжета и визуальной составляющей, оценку экономической эффективности и планирование сроков реализации.

Будучи студентом, изучающим информатику, можно изучать разработку игр, потому что эта область требует некоторых навыков, которые уже есть, и в то же время вы можете их улучшить.

В рамках текущего проекта, будет проектироваться компьютерная игра в жанре «Point & Click», так как, это очень интересный жанр в игровой индустрии и для потребителя требуется минимальный порог вхождения, не

требующий особых знаний и навыков для комфортного игрового процесса, не как в играх серии Dota или Counter-Strike, где необходимо знать очень много тактик, наизусть знать локации и зоны игровых сражений, идеальное умение в использовании основных персонажей или типа игрового вооружения и снаряжения. Для простых пользователей – это может вызвать затруднение и дискомфорт во время игрового сеанса, что оставит негативные эмоции и усилит усталость. Ведь игры, в первую очередь, должны помочь человеку расслабиться и погрузиться в игровой мир, для получения положительных эмоций и хорошего отдыха.

Сегодня с вычислительными устройствами человек сталкивается практически постоянно. Если еще полвека лет назад компьютеры можно было встретить только в научно-исследовательских центрах, институтах и на крупных предприятиях, то сейчас они проникли почти во все сферы жизни. К одной из таких сфер относится индустрия развлечений. Практически с самого своего зарождения, разработка компьютерных игр стала одной из самых трудоемких задач для различных видов специалистов из сферы информационных технологий и смежных областей. Однако, как и разработка промышленного и другого программного обеспечения, создание компьютерных игр основывается на тех же самых циклах разработки, где нужен поэтапный подход, включающий сюда прогнозирование, работу с целевой аудиторией, экономические оценки, постановку задачи, разработку инструментария и самого продукта, создание мультимедийных ресурсов, контроль качества и прочее, которые в настоящее время не доведены до цикличности и структуризации. Данный аспект определил актуальность вопроса исследования выпускной квалификационной работы.

Объектом исследования является проектирование компьютерных игр.

Предмет исследования – технологии разработки компьютерной 2D игры жанра «Point & Click».

Цель работы - спроектировать концепт компьютерной игры жанра «Point & Click».

Для достижения поставленной цели были определены следующие задачи:

1. Проанализировать предметную область проекта
2. Проанализировать программные средства и технологии создания игр
3. Сформировать требования к игровому ПО
4. Спроектировать эскиз визуальной составляющей игры

Методы решения главной задачи работы – экспериментальный, методы системного анализа, моделирования, нормализации данных, теоретический.

Теоретико-методологической основой исследования послужили работы отечественных и зарубежных ученых в области ИТ – проектирования в сфере разработки инструментария, учебные пособия, монографии, научные статьи, ресурсы сети Интернет.

1 Анализ объекта и предмета исследования

1.1 Общее представление о разработке игр

Компьютерные игры в основном состоят из программного обеспечения, которое можно разделить на две широкие функции: игровой движок, управляющий ходом игры, и системы, поддерживающие живую работу. Некоторые организации используют собственные игровые движки, сделанные на заказ, в то время как другие используют готовые движки, такие как Unity и Unreal.

В то время как небольшие независимые игры требуют всего понемногу, крупные игровые студии, как правило, имеют масштабы для поддержки собственных команд инженеров-программистов, которые разрабатывают различные элементы игры с нуля. В такой команде часто есть три основные роли:

- Программист геймплея – инженеры в этой роли сосредоточены на том, как игра выглядит с точки зрения пользователя, что включает в себя добавление элементов дизайна и обеспечение удобства системы

- Графический программист – инженеры в этой роли отвечают за создание эффективного механизма рендеринга или настройку технических функций, чтобы игра хорошо работала на разных системах. Программисты графики смогут решить сложные проблемы оптимизации и производительности, обычно используя C++

- Инженер-программист внутренних систем – инженеры в этой роли пишут компьютерный код для систем игры, которые игроки не видят напрямую; например, логика на стороне сервера и функции входа или выставления счетов в крупной онлайн-игре.

Помимо этих ролей программирования, есть должности, которые поддерживают живые операции, такие как инженеры по сетям, безопасности и эксплуатации (они относятся к системному администрированию), а также различные члены художественных команд, включая аниматоров, создателей

моделей персонажей и художников по окружению. Благодаря развитию рынка инди-игр, многие разработчики компьютерных игр получили возможность работать над своими игровыми проектами без финансовых и юридических обязательств перед издательскими компаниями. Инди-игры (англ. Indie games, от англ. independent video games - «независимые компьютерные игры») - это компьютерные игры, созданные отдельными разработчиками или небольшими коллективами без финансовой поддержки издателя компьютерных игр. Распространение осуществляется посредством каналов цифровой дистрибуции. Масштаб явлений, связанных с инди-играми, ощутимо возрастает со второй половины 2000-х годов, в основном ввиду развития новых способов онлайн-дистрибуции и средств разработки.

Компьютерные игры оказали настолько значительное влияние на общество, что в последнее время в информационных технологиях наблюдается устойчивая тенденция к геймификации неигрового прикладного ПО. Так, в некоторых европейских школах для обучения стали использовать известную игру Minecraft, а для нужд армий стали создавать специальные программные тренажеры для обучения солдат. Компьютерные игры так же с 2011 года официально признаны правительством США и американским Национальным фондом отдельным видом искусства, наряду с театром и кино, а в России киберспорт официально был причислен к видам спорта [8]. Из всего этого следует, что компьютерные игры плотно влились в нашу нынешнюю жизнь. Более того, сфера их использования за последние 10 лет сильно выросла, теперь игры используют не только для развлечения и отдыха, но и для обучения людей и проведения научных исследований.

Компьютерные игры классифицируют по нескольким основным признакам:

- жанр
- количество игроков
- визуальное представление
- платформа

Жанр игры определяется целью и основной механикой игры. Основные жанры представлены в таблице 1.

Таблица 1 - Жанровая классификация [5]

Жанр	Описание
Аркада (Arcade)	игры с нарочно примитивным игровым процессом
Приключенческая игра (Adventure)	главной частью игры является история
Ролевая игра (RPG)	главная особенность игры, игрок практически не ограничен в выборе игровых предметов, напарников и диалогов
Экшен (Action)	игра, характеризующая частым и активным нажатием кнопок управления
Стратегическая игра (Strategy)	основа игры – необходимость игроку делать нетривиальный выбор
Компьютерный симулятор (Simulator)	игрок делает множество упражнений и оттачивает свою технику
Головоломка (Puzzle)	требует аналитического мышления
Обучающая игра (Educational)	игрок обучается во время выполнения каких-либо действий в игре
Игрушки (Toys)	программы, взаимодействуя с которыми, игрок получает удовольствие

Представленные здесь жанры нельзя назвать полными, так как в последнее время стали появляться игры собственных жанров, которые можно отнести как к одному из представленных здесь жанров, так и к самостоятельному отдельно от них. Игры делятся на следующие виды.

По количеству игроков:

- однопользовательские;
- многопользовательские.

По визуальному представлению:

- текстовые – минимальное графическое представление, общение с игроком проходит с помощью текста;
- 2D – все элементы отрисованы в виде двумерной графики (спрайтов);
- 3D – все элементы отрисованы в виде трехмерной графики (3D –

- модели).

По типу платформы:

- персональные компьютеры;
- игровые приставки/консоли;
- мобильные телефоны.

Основываясь на данной классификации компьютерных игр было принято решение разработать двумерный однопользовательский платформер для персональных компьютеров. Данное решение было принято потому, что платформер обладает простой и понятной механикой, что означает легкость в проектировании и разработке для начинающих игровых разработчиков [17].

Данный проект относится к инди-разработке и развивается только за счет средств его разработчиков. Благодаря развитию ПО разработки компьютерных игр, команде разработчиков не требуется тратить несколько лет на разработку игрового движка.

Это позволяет сразу же приступать к непосредственной работе над игровым проектом и значительно сокращает время его разработки.

1.2 Сравнительный анализ игр жанра «Point & Click»

Среди всех типов игр «Point & Click» – довольно распространенная категория, даже если в течение некоторого времени она сокращалась. Популярность мобильных игр обновила жанр и открыла перед дизайнерами новые возможности [3].

«Point & Click» [16] относится к типу приключенческой игры, в которой пользователь взаимодействует в основном с помощью мыши или любого указывающего устройства (будь то палец на мобильных телефонах). Этот тип игры требует минимальных знаний о компьютерах и очень прост в использовании даже для обычных игроков по сравнению с 3D-FPS или такой игрой, в которой вы будете использовать 100-кнопочную мышь, чтобы выжить. Чтобы сделать его максимально простым, взаимодействие обычно состоит из:

- нажатия;
- двойной щелчок;
- перетаскивания.

Вся игра построена вокруг взаимодействия с предметами мира. Наиболее классические взаимодействия:

- разговаривать;
- смотреть на что-то;
- взять что-нибудь;
- использовать предмет;
- комбинировать предметы.

Большая часть этапа игрового дизайна состоит в том, чтобы свободно использовать эти возможности, чтобы заставить игрока находить решения проблем. Решение их заставит приключение продолжиться. В некотором смысле игры типа «укажи и щелкни» похожи на игры-головоломки. Количество взаимодействий довольно ограничено, но их количество отличает point-and-click от визуальных новелл, в которых их меньше, и даже от исследовательской игры, в которой они могут вообще не быть головоломкой [30, с.210].

Игры LucasArt часто считаются наиболее важными в истории point-and-click с таким названием, как Maniac Mansion. Они были созданы с использованием языка сценариев под названием SCUMM. Этот язык носит описательный характер и был создан для того, чтобы геймдизайнеры, сценаристы, диалогисты и т. д. могли модифицировать игру без помощи программиста на каждом этапе [22, с.131].

В наши дни, благодаря популярности мобильных игр и их казуальному характеру, такие игры:

- состоит из истории и головоломок
- использует мало системных ресурсов
- не требует специальных устройств
- не требует специальных навыков исполнения

У этого жанра все больше и больше игроков, и довольные люди продолжают приходить к нему из других областей, таких как литература.

Самыми знаменитыми представителями жанра Point and Click можно назвать серии Syberia (Сибирь), The Curse of Monkey Island, Papers Please, Botanicula и другие.

Остановимся и подробнее рассмотрим некоторые игры данного жанра:

Syberia (дата релиза 2002 г.) – представляет собой point-and-click квест. Геймплей (игровой процесс) заключается в перемещении между локациями, поиске активных точек на экране и решении головоломок. Управление осуществляется при помощи мыши. При наведении на активную точку курсор меняет форму в зависимости от того, что может сделать Кейт (главная героиня): перейти на другую локацию, взять предмет, поговорить с персонажем и др.

В отличие от большинства игр этого жанра, инвентарь разделен на две части: в одной хранятся предметы, в другой – документы (которые в некоторых случаях приходится использовать как предметы).

Также в инвентаре постоянно присутствует телефон Кейт, по которому с ней регулярно связываются её начальник, мама, подруга Оливия, а также жених Дэн. Некоторые головоломки связаны с использованием телефона(рисунок 1-3).



Рисунок 1 – Скриншот из игры «Syberia»



Рисунок 2 – Скриншот из игры «Syberia»



Рисунок 3 – Скриншот из игры «Syberia»

The Curse of Monkey Island (дата релиза 1997 г.) – это третья часть квеста в жанре Point and Click с отличным сюжетом и головоломками, выпущенная в далёком 1997 году. Проект сейчас мало кого может привлечь, поскольку максимальное разрешение 640 x 480 точек, качество звука и интерфейс выглядят очень древними. Однако попробовав хотя бы одну часть на вкус, очень легко стать фанатом приключений на острове обезьян. Ведь здесь продуман каждый элемент приключения, а события вызывают бурю различных эмоций. И если вам всё же чужда подобная графика, то разработчики уже позаботились о ремастере (обновление графической составляющей игры, с

сохранением использованием исходного «движка» игры), благодаря которому познакомиться со знаменитой вселенной теперь может ещё больше людей (рисунок 4-6).



Рисунок 4 – Скриншот из игры «The Curse of Monkey Island»



Рисунок 5 – Скриншот из игры «The Curse of Monkey Island»



Рисунок 6 – Скриншот из игры «The Curse of Monkey Island»

Papers, Please (дата релиза 2013 г.) – это симулятор сотрудника миграционной службы, в котором игрокам придется проверять бумаги определенного числа людей, и или отказывать им в пребывании в стране, или же пропускать их. На рисунке 7 можно ознакомиться с визуальной частью игры.



Рисунок 7 – Скриншот из игры «Papers, Please»

В случае выявления несоответствия фотографии в паспорте с человеком, стоящим перед игроком, игрок может попросить мигранта сдать отпечатки пальцев для проверки личности пример пользовательского интерфейса можно увидеть на рисунке 8. Если пол человека, указанный в паспорте, вызывает сомнения, или вес персонажа на весах и в документах различаются, то игрок может попросить гражданина пройти сканирование, после которого игрок получит обнажённую карточку, на которой будет виден пол человека или наличие на теле сканируемого контрабанды. Нарушителя с поддельными документами или контрабандой можно арестовать, нажав на кнопку, появившуюся после выявления нарушения. После нажатия на неё в помещении заходят охранники и задерживают персонажа.



Рисунок 8 – Скриншот из игры «Papers, Please»

Исходя из вышеперечисленного, игры жанра Point & Click действительно разнообразны и по-своему интересны. Но, эти игры были выпущены в свет очень давно, соответственно, разработка происходила на актуальных программных обеспечениях и аппаратно-технических средств того же времени, вероятно, и более старых, так как разработка игр занимает довольно длительное время, в момент релиза проекта, уже на рынке игровой индустрии могут появиться более новое оборудование и программное обеспечение.

В рамках текущего проекта будет учитываться технологический скачок. В разработке будет использоваться современное программное обеспечение и более мощные аппаратно-технические средства.

1.3 Общий алгоритм реализации проекта

Создание ассетов на этапе производства – это базовый этап, на котором разработчики игр создают различные ассеты, а затем используют их на этапе реализации игры. На этапе производства первым шагом является создание ресурсов для игры.

Создание раскадровки – самый важный этап производства игры; он включает в себя разработку игровых сценариев для решений уровней и включение методов планирования искусственного интеллекта для представления различных особенностей игр с помощью традиционной белой доски или блок-схемы.

В исследованиях, отнесенных к категории выбора платформы разработки, предлагались различные типы платформ для разработки игр. Платформы разработки предоставляют готовую архитектуру для подключения сервер-клиент и помогают разработчикам быстро создавать игры.

Семантику игры можно отнести к формальному описанию языка для языков программирования; по этой классификации сообщалось только о двух исследованиях. Формальное языковое описание семантики игр дало возможность получить представление о дизайне языков программирования для

разработки игр. Сложность кода возрастает, особенно при разработке игр, из-за включения сложных модулей, методов искусственного интеллекта и разнообразного поведения. Наиболее распространенными языками программирования, используемыми в разработке игр, являются объектно-ориентированные структурированные языки, такие как Java, C и C++. Исследования, отнесенные к этой категории, изучали программный аспект разработки игр.

Исследование влияние объектно-ориентированной технологии на производительность, размер исполняемого файла и методы оптимизации для мобильных игр и предположил, что объектно-ориентированную технологию следует использовать с большой осторожностью, поскольку структурное программирование в разработке игр очень конкурентоспособно.

Игровой движок – это своего рода специальная программная среда, которая используется на этапе производства для создания и разработки игр. Игровые движки состоят в основном из комбинации основных функций, таких как звук, физический движок или обнаружение столкновений, ИИ, сценарии, анимация, работа в сети, управление памятью и графы сцен.

Основы теории игр используются при разработке игр, потому что это раздел теории принятия решений, описывающий взаимозависимые решения. В большинстве исследований этой категории описывались различные аспекты технологий реализации игр на разных типах платформ. Они рассматривали возможность улучшения навыков программирования, 2D/3D-анимации и графики, звукорежиссуры, управления проектами, логического дизайна, дизайна интерфейса для написания историй и методов искусственного интеллекта.

2 Программные средства и технологии создания игр

2.1 Анализ игровых движков

В настоящее время серьезные игры, а также разработка мобильных игр находятся на подъеме. День за днем настольные и консольные игры заменяются играми для мобильных телефонов и планшетов. Более того, люди, занимающиеся дизайном и разработкой серьезных игр, приходят из самых разных сфер. Например, педагоги и эксперты в предметной области с ограниченными навыками программирования, если таковые имеются, участвуют в проектировании и разработке серьезных игр. Таким образом, выбор разработки игры с использованием исключительно языка программирования, такого как C++, C# или Java, на самом деле возможен не для всех. Это сделало использование игровых движков еще более важным.

Игровые движки ускоряют процесс разработки игры с помощью существующих шаблонов и ресурсов, которые можно использовать повторно, сводя к минимуму или полностью устраняя необходимость иметь глубокие знания в области программирования. Более того, игровые движки дают возможность один раз разработать игру и экспортировать ее на различные платформы, включая мобильные устройства, внося всего несколько изменений в оригинальную версию. Однако существует множество игровых движков, которые имеют некоторые общие черты, но также имеют много различий. На самом деле разные игровые движки имеют разную философию разработки игр и нацелены на широкий спектр различных потребностей: движки, не требующие знаний программирования; движки, основанные на популярных веб-технологиях; движки с открытым исходным кодом, которые могут быть настроены/расширены опытными пользователями; и профессиональные игровые движки [27,с.47].

Сделать осознанный выбор игрового движка – это решение, основанное на нескольких критериях, и это непросто. Хотя существует несколько обзорных статей об игровых движках, в большинстве случаев эти статьи посвящены

определенному типу игровых движков (например, 3D-мобильным игровым движкам) или конкретной области (например, смоделированное хирургическое обучение). Предоставление обзора игровых движков, которые охватывают широкий спектр различных профилей и потребностей пользователей на основе расширенного списка функций, считается важным. Насколько мне известно, такого обзора нет, и это послужило поводом для написания этого раздела в выпускной квалифицированной работе. Следовательно, цель этого раздела – детально проанализировать и показать все функции, которые могут присутствовать в игровом движке, организованными в категории более высокого уровня; основные типы игровых движков; возможности, предлагаемые репрезентативными примерами игровых движков, попадающих в вышеупомянутые категории. Конечная цель – определится в выборе наиболее подходящего игрового движка, исходя из потребностей конкретного игрового проекта и профиля человека (лиц), которые собираются его реализовывать.

Краткое описание игровых движков.

GameMaker был впервые выпущен под названием Animo в 1991 году и подходил для создания 2D-игр. Хотя игровой движок подходит для 2D-игр, он позволяет пользователю добавлять 3D-графику и физику. GameMaker прост в использовании для начинающих пользователей, так как не требует знаний в области программирования. Однако функциональные возможности 3D-камеры по сравнению с 3D-графикой ограничены [4].

JMonkey был впервые выпущен в 2003 году. Это бесплатный игровой движок, который бесплатно экспортирует игры на все платформы, включая мобильные устройства. JMonkey поддерживает 3D-графику. Хотя необходимо иметь опыт программирования на Java, это движок с открытым исходным кодом, который позволяет пользователям расширять его и адаптировать к своим потребностям [6].

Первоначальная версия игрового движка Marmalade была выпущена в 2015 году. Этот движок был выбран, поскольку он бесплатно экспортирует игры на многие мобильные платформы и обеспечивает возможности

рендеринга 3D-графики. Кроме того, Marmalade способствует созданию настольных веб-приложений. Версия, которая анализируется в данном проекте – 8.5 [20,С.118].

Игровой движок OGRE 3D [9] был впервые выпущен в ноябре 2013 года. Ogre3D – это бесплатный движок с открытым исходным кодом. Он позволяет экспортировать игру на iOS, Android и Windows Phone 8 и рендерить 3D-графику. Версия, которая анализируется в данном проекте – 2.1.

Игровой движок Shiva [10] был впервые выпущен в июле 2007 года. Он обеспечивает визуализацию 3D-графики и использовался для разработки очень популярных игр, таких как Prince of Persia 2, воссозданная для мобильных устройств с использованием Shiva, и Babel Rising, был опубликован Ubisoft.

Игровой движок Sio2 был впервые выпущен в 2009 году и уже через год стал одним из 3-х самых популярных игровых движков, используемых в App Store. Sio2 – это игровой движок, который ранее был бесплатным и с открытым исходным кодом; однако теперь он доступен пользователям за плату [11].

Можно также получить доступ к его исходному коду, купив версию сертифицированного разработчика. Пробная версия предоставляется пользователям с большинством доступных функций, таких как мобильный экспорт.

Тем не менее, игровая симуляция предоставляется на различных устройствах iOS и является одним из самых популярных игровых движков для экспорта на платформы iOS, предоставляя расширенную 3D-графику.

Игровой движок Turbulenz был создан в апреле 2013 года и является единственным изучаемым нами движком, который позволяет экспортировать игры в мобильные браузеры на основе многообещающего подхода веб-технологий. В то же время Turbulenz является бесплатным, с открытым исходным кодом и поддерживает рендеринг 3D-графики [13].

Движок Unity Game был впервые публично анонсирован на Всемирной конференции разработчиков Apple в 2005 году и является одним из самых известных игровых движков. Unity бесплатно предоставляет расширенный

рендеринг 3D-графики и экспорт на мобильные устройства. Тем не менее, это один из самых популярных движков в игровой индустрии, с помощью которого было создано множество успешных игр, таких как Deus Ex: The Fall для мобильных телефонов или Assassin's Creed: Identity [2].

Первоначальный выпуск Unreal Engine состоялся в 1998 году, а Unreal Engine 4 впервые был выпущен в мае 2012 года.

Этот движок подходит для рендеринга 3D-графики, является бесплатным и открытым исходным кодом, а также является одним из самых популярных игровых движков, как и многие игры. были созданы с его помощью и преуспели в такой отрасли, как Absolver.

В таблице 2 представлены результаты сравнительного анализа признаков, попадающих в категорию аудиовизуализации.

Таблица 2 – Сравнительный анализ игровых движков в области аудиовизуализации

Особенности/Игровые движки	Game Maker	JMonkey	Marmalade	Ogre3D	Shiva	Sio2	Turbulenz	Unity	Unreal Engine 4
1. Визуализация									
1.1 Текстура									
1.1.1 Основные	√	√	√	√	√	√	√	√	√
1.1.2 Мульти-текстурный	x	√	√	√	√	√	√	x	√
1.1.3 Процедура	x	√	√	√	√	x?	√	√	√
1.2 Освещение									
1.2.1 Для каждой вершины	√	√	√	√	√	√	√	√	√
1.2.2 Попиксельно	x	x	x?	√	√	√	√	√	√
1.2.3 Картирование освещения	x	√	√	√	√	√	x?	√	√
1.2.4 Карты блеска/зеркальности	x	√	√	x?	√	x?	√	√	√

Продолжение таблицы 2

1.2.5 Анизотропное отражение	x	√	x?	x?	√	√	√	√	√
1.3 Тени									
1.3.1 Объем теней	x	√	x?	√	x	x?	x?	x	√
1.3.2 Картирование теней	x	√	√	√	√	√	√	√	√
1.3.3 Прогноз	x	x	x?	√	√	√	x?	x	√
1.3.4 Планарная проекция	x	x	x?	√	x	x?	x?	√	√
1.4 Специальные эффекты									
1.4.1 Карты окружающей среды	√	√	x?	√	√	√	√	√	√
1.4.2 Система частиц	√	√	√	√	√	√	√	√	√
1.4.3 Рекламный щит	√	√	√	√	√	√	√	√	√
1.4.4 Блик объектива	x	√	x?	√	√	√	√	√	√
2. Анимация									
2.1 Прямая кинематика	x	x	x?	x	√	x	x?	√	√
2.2 Инверсная кинематика	x	x	x?	√	√	x	x?	√	√
2.3 Анимация ключевых кадров	x	√	√	x	√	√	√	√	√
2.4 Скелетная анимация	√	√	√	√	√	√	√	√	√
2.5 Морфинг анимации	√	x	√	√	x?	x	x?	√	√
2.6 Смешивание	x	√	x?	√	√	x?	√	√	√
3. Звук									
3.1 2D Звук	√	√	√	√	√	x	x?	√	√
3.2 3D Звук	√	√	√	√	√	√	√	√	√

Продолжение таблицы 2

3.3 Потокное звучание	√	x	√	√	√	√	√	√	√
4. 3D-графика для мобильных устройств									
4.1 Картирование									
4.1.1 Редактор карт									
4.1.1.1 Рельефное отображение	x	√	√	√	√	x	x?	√	√
4.1.1.2 Отображение параллакса	x	√	x	√	√	x	x?	√	√
4.1.1.3 Отображение нормалей	x	√	√	√	√	x	√	√	√
4.2 Глобальное освещение									
4.2.1 Трассировка лучей	x?	x?	x?	√	√	x	√	√	√
4.2.2 Окклюзия окружающей среды	x	√	x	√	√	x	x?	√	√
4.2.3 Редактор кат-сцен	x	√	x	x?	x	x	x?	√	√
5. Редактор сцен									
5.1 Отсечение окклюзии	x	x	x?	√	√	√	√	√	√

Легенда к таблице 2:

– «√» – функция, обнаруженная на официальном веб-сайте движка и в некоторых случаях подтвержденная в литературе;

– «x» – функция не поддерживается на основании данных официального веб-сайта движка и в некоторых случаях подтверждена литературой;

– «x?» – функция не обнаружена ни в информации, доступной на официальном сайте, ни в литературе. Однако мы не можем быть уверены, что эта функция не поддерживается.

Game Maker – наименее развитый игровой движок с точки зрения аудиовизуализации, поскольку единственная категория, в которой он поддерживает все функции, – это звук.

JMonkey – один из лучших игровых движков в категории статического глобального освещения вместе с Unreal Engine 4, который даже лучше, и Ogre3D, но он не поддерживает попиксельное освещение, потоковое аудио и отсечение окклюзии.

Marmalade также не поддерживает попиксельное освещение и окклюзию окружающей среды, анизотропное отражение, картографирование окружающей среды, блики и анимацию смешивания.

Ogre3D имеет преимущество в категории теней наряду с Unreal Engine.

Shiva – это движок, который имеет как преимущество, так и недостаток в особенностях анимации, и конкретно прямая анимация является сильной стороной в отличие от анимации морфинга. В результате анализа было замечено, что Sio2 имеет большой недостаток в 3D-графике для мобильных устройств и отображении редактора, поскольку он не поддерживает ни одну из соответствующих функций, которые изучали в этих категориях. В то же время он не поддерживает 2D-звук, анимацию морфинга, анимацию смешивания, карты блеска/зеркала и процедурную текстуру. Turbulenz не поддерживает наложение света, трансформирующуюся анимацию, смешение, 2D-звук и рельефное наложение. Unity не поддерживает только несколько текстур, в то время как, с другой стороны, он поддерживает все остальные функции и, кажется, превосходит остальные движки в анимации вместе с Unreal Engine 4. Unreal Engine 4, кажется, превосходит аудиовизуальную точность, поскольку поддерживает все функции и имеет преимущество в анимации (наряду с Unity, как упоминалось), тенях и глобальном освещении. Разработка в Unreal Engine 4 очень проста для начинающих. С помощью системы визуального создания скриптов Blueprints Visual Scripting можно создавать готовые игры, не написав ни строчки кода. В сочетании с удобным интерфейсом это позволяет быстро изготавливать рабочие прототипы [24, с.47].

Таблица 3 – Сравнительный анализ игровых движков в области функциональности

Особенности/ Игровые движки	Game Maker	JMonkey	Marmalade	Ogre3D	Shiva	Sio2	Turbulenz	Unity	Unreal Engine 4
1. Сценарий									
1.1 Язык программирования	GameMaker Language	Java	C, C++	C++	Lua (+ Shiva)	C, C++, Lua	JavaScript, HTML	C#, JavaScript, Boo	C++, Blueprint
1.2 Редактор программ	√	√	x	√	√	x	√	√	√
1.3 Редактор графического интерфейса	x	√	√	√	√	x	x	√	√
1.4 Отладчик	√	√	√	√	√	√	√	√	√
1.5 Профайлер	√	√	x	√	x	√	√	√	√
2. Искусственный интеллект									
2.1 Редактор ИИ	x	√	x	x	√	√	x?	√	√
2.2 Поиск пути	√	√	x	√	√	√	x	√	√
2.3 Принятие решений	x	√	x	x?	√	x?	x?	√	√
2.4 Обнаружение столкновений	√	√	x	√	√	√	√	√	√
3. Физика									
3.1 Физические инструменты									
3.1.1 Редактор анимации	√	x?	√	√	√	x	x?	√	√
3.1.2 Редактор материалов	x	√	√	√	√	x	x?	√	√
3.1.3 Редактор эффектов частиц	x	√	x	√	√	x	x?	√	√
3.1.4 Редактор 3D-аудио	√	x?	√	x	√	x	x?	√	√
3.2 Базовая физика	√	√	√	√	√	√	√	√	√
3.3 Жесткое тело	√	√	√	√	√	√	√	√	√

Продолжение таблицы 3

3.4 Динамика автомобиля	x	√	x?	x	√	√	x?	√	√
3.5 SDK для 3D-физики	Bullet	Bullet	ODE	x?	ODE	Bullet	x?	PhysX	PhysX

Легенда к таблице 3:

– «√» – функция, обнаруженная на официальном веб-сайте движка и в некоторых случаях подтвержденная в литературе;

– «x» – функция не поддерживается на основании данных официального веб-сайта движка и в некоторых случаях подтверждена литературой;

– «x?» – функция не обнаружена ни в информации, доступной на официальном сайте, ни в литературе. Однако мы не можем быть уверены, что эта функция не поддерживается.

Как видно из анализа, Unity и Unreal Engine 4 поддерживают все основные функции и имеют поддержку PhysX SDK для физического моделирования. Так же, видно, что игровой движок Turbulenz демонстрирует больше всего недостатков в области искусственного интеллекта и физики.

По результатам сравнительного анализа можно сделать несколько важных выводов относительно эффективности каждого игрового движка для категорий характеристик, исследованных в данном исследовании:

Было замечено, что Game Maker не подходит для разработки игр с 3D-графикой. Тем не менее, он идеально подходит для простого создания игр с 2D-графикой, обеспечивая удобную среду и возможность разрабатывать игры без какого-либо опыта программирования.

JMonkey не справляется с некоторыми аспектами аудиовизуальной точности, такими как отсутствие отбраковки окклюзии или анимации и инструментов 3D-звука. Более того, он не может экспортировать игры на

консоли. С другой стороны, это игровой движок с открытым исходным кодом, позволяющий пользователям расширять его функциональность.

Marmalade – это движок, которому не хватает нескольких функций в каждой категории, но у него есть преимущество в категории платформ развертывания, поскольку он может экспортировать игры на большинство игровых платформ.

Ogre3D, как и JMonkey, не поддерживает многие функции категории аудиовизуальной и функциональной точности. С другой стороны, Ogre3D также является игровым движком с открытым исходным кодом, поэтому его можно адаптировать к потребностям опытного пользователя.

Кроме того, Ogre3D идеально подходит для создания теней и подходит для начинающих, предоставляя неплохую поддержку. Шива не так доступна для пользователей, потому что она не бесплатна, а также не подходит для неопытных пользователей. Тем не менее, он содержит почти все аудиовизуальные функции, и в то же время превосходит в физическом подклассе функциональной точности, поддерживая все функции. Sio2 подходит для создания приложений в программном обеспечении iOS, так как его ядро основано на этой операционной системе. Ему не хватает многих функций аудиовизуальной и функциональной точности. Он также не так доступен для пользователей, потому что, хотя раньше он был бесплатным и с открытым исходным кодом, сейчас это самый дорогой игровой движок.

Turbulenz – это игровой движок, которому не хватает нескольких функций в категории функциональной точности. Однако он имеет открытый исходный код и идеально подходит для создания приложений для браузеров с использованием графического API WebGL.

Unity и Unreal Engine 4 – самые мощные игровые движки, которые поддерживают почти все функции, включенные в используемый фреймворк. Unreal Engine 4 обеспечивает лучшую аудиовизуальную производительность, чем Unity, поскольку он поддерживает больше функций и в то же время является движком с открытым исходным кодом [29, с.347].

Эмпирическое исследование Unity и Unreal Engine 4 были выбраны обзор и бенчмаркинг для дальнейшего сравнения, потому что они явно являются двумя наиболее развитыми игровыми движками.

В контексте эмпирического исследования были разработаны два похожих настольных шутера на двух игровых движках с помощью руководств на их официальных веб-сайтах, чтобы наилучшим образом использовать различные функции двух движков [10].

В частности, игра Survival Shooter была разработана в Unity, а Twin Stick Shooter – в Unreal Engine 4. Затем игры были обработаны для запуска на мобильных устройствах и экспортированы для тестирования на мобильном устройстве Android для получения результатов.

В ходе эмпирического исследования изучались такие характеристики, как удобство использования движка, кривая обучения, процесс экспорта для мобильных устройств, а также качество игры и размер приложения.

Исходя из всего вышеперечисленного, оптимальным выбором для проектирования игры, будет Unity, так как имеет поддержку языка программирования C#.

2.2 Анализ графических редакторов

Для создания игры недостаточно определиться с выбором игрового движка, так же, как и недостаточно заливки фундамента для постройки жилого дома. В играх присутствуют очень много объектов: персонажи, объекты декора, что в совокупности создают общий план игровой зоны, элементы интерфейса, объекты взаимодействия (иными словами, ключевыми предметами). Это могут быть, как 2D или 3D модели. Обычно, их можно реализовать в разных графических редакторах и экспортировать в созданный проект игрового движка.

Для наглядности, составим таблицу 4 и посмотрим, какие игровые движки совместимы с экспортом объектов из графических редакторов.

Таблица 4 – Сравнительный анализ игровых движков в области импорта/экспорта объектов из графических систем

Особенности/Игровые движки	Game Maker	JMonkey	Marmalade	Ogre3D	Shiva	Sio2	Turbulenz	Unity	Unreal Engine 4
1. Импорт/Экспорт контента									
1.1 MAYA	x	√	√	√	√	√	√	√	√
1.2 3D Studio MAX	x	√	√	√	√	√	√	√	√
1.3 Blender	x	√	√	√	√	√	√	√	√

Легенда к таблице 4:

– «√» – функция, обнаруженная на официальном веб-сайте движка и в некоторых случаях подтвержденная в литературе;

– «x» – функция не поддерживается на основании данных официального веб-сайта движка и в некоторых случаях подтверждена литературой;

– «x?» – функция не обнаружена ни в информации, доступной на официальном сайте, ни в литературе. Однако мы не можем быть уверены, что эта функция не поддерживается.

Для создания эскизов, можно использовать графический редактор Adobe Photoshop, в моём арсенале было ещё приложение Procreate на iPad со стилусом Apple Pencil, что упростило возможность реализации визуального планирования интерфейса будущей игры.

3 Проектирование «2D платформера жанра Point & Click»

3.1 Требования к игре

В рамках текущего проекта, жанром игры является «Point & Click», соответственно весь игровой процесс должен сопровождаться управлением с помощью компьютерной мыши, включая основные аспекты: передвижение главного героя по локации, взаимодействие с окружающими предметами и не игровыми персонажами (НПС).

Так же, после запуска игры, должен отображаться стартовый экран (иначе говоря, «Главное меню»), на котором расположены кнопки: «Загрузить игру», «Новая игра», «Настройки», «Об авторе» и «Выход из игры». Эта небольшая навигация имеет большое влияние на общее впечатление о готовом продукте, так как данный экран – это первое, что видит пользователь после запуска игры.

Пункт меню «Загрузить игру» должен считать файлы сохранения и после выбора нужного сохранения, перенести пользователя на сохраненный этап игрового процесса.

Пункт меню «Новая игра» должен предупредить пользователя, о утрате заработанного прогресса и, в случае соглашения с предупреждением, создается новый файл сохранения с нулевым прогрессом.

Пункт «Настройки» должен содержать данные-константы, для управления графическими или звуковыми значениями игрового процесса, например, повысить или понизить уровень громкости звука в игре.

Пункт «Выход» должен предупредить пользователя о выходе из программы. В случае соглашения с предупреждением, произвести завершение работы игры. Так же, во время игрового процесса, у пользователя должна быть возможность открытия небольшого меню с пунктами навигации «Продолжить», «Настройки» и «Выход в главное меню».

Пункт «Продолжить» должен закрыть экран «Меню».

Пункт «Настройки» аналогичен по функционалу с пунктом «Настройки» из главного меню программы.

Пункт «Выход в главное меню» должен сохранить игровой процесс пользователя и произвести открытие экрана «Главное меню».

3.2 Характеристика оборудования разработки

Для реализации игры требуется достаточно мощная конфигурация ПК.

На самом деле, всё зависит от направления разработчика, у программистов и

Характеристики аппаратной составляющей ПК варьируются в зависимости от направления разработчика и инструментария, которым он пользуется. Например, требования к железу у программистов графики, художников и 3D-модельеров разные.

В основном, требуется мощный многоядерный процессор, для быстрого компилирования проекта, а также для работы с 3D моделями (к примеру, при компиляции 3D-объекта, вся нагрузка перекладывается на процессор.

Оперативная память тоже очень важна, особенно при редактировании многополигональных объектов или при работе нескольких программ одновременно. Для хранения данных, нужно использовать HDD или SSD больших объем накопителей (HDD будет дешевле, но SSD намного быстрее). Например, на игровом клиенте игры могут достигать (но не ограничиваться) – 30-70 Гб, но у разработчиков, исходные данные игры, могут и не уместиться на диск объемом в 1Тб. Использование SSD обусловлена высокой скоростью работы, но ограничена циклами записи, со временем SSD приходят в негодность, чего не сказать про HDD (они тоже со временем приходят в негодность, но не по причине износа чипов памяти, а из-за «старости»). Для 3D графики важна видеокарта, поэтому в тяжелых проектах более комфортную работу обеспечат современные видеокарты. Оптимальная сборка оборудования: 32 Гб ОЗУ, процессор на 8 ядер 16 потоков intel/amd, видеокарта серии NVidia GeForce GTX1060/RTX2060. Для хранения исходных данных проекта, рекомендуется использовать HDD объёмом от 512 гб [25, с.44]

3.3 Проектирование графического оформления

В качестве тематики и визуального оформления игрового пространства было определено футуристичное оформление, а если быть точнее – стиль киберпанка. Практически большинство современных игр, стремятся к созданию сверхреалистичных текстур, физики управления персонажами/транспортом, физики поведения окружающего мира и т.д. Но, в данном проекте, будет рассматриваться другое направление визуального оформления – Pixel Art. Все объекты будут иметь лесенки и заметные, крупные пиксели. Это позволит сохранить баланс, между приятной графикой в игре, нагрузку на компьютер пользователя во время игрового сеанса, а также уменьшит размер игры на жестком диске.

По традиции, после каждого запуска игры, происходит его загрузка, а позже пользователь видит главное меню. Во время загрузки игры, пользователь видит статичную или анимированную картинку. За этим намного приятнее наблюдать, нежели на черный фон и гадать, «игра грузится или зависла?». Для начала, необходимо составить логотип игры по её названию, чтобы придать ей уникальность. Игра будет называться «Endless Night».

Проектирование логотипа будет происходить в графическом ПО Adobe Photoshop. Для начала, необходимо добавить квадрат (рисунок 9), а после закруглить углы (рисунок 10).

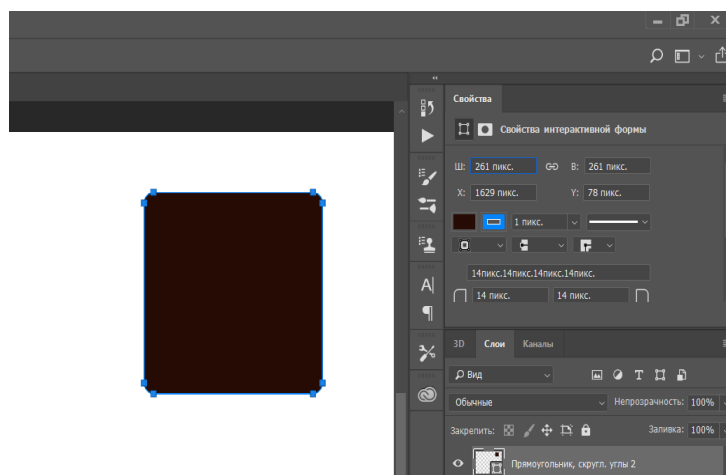


Рисунок 9 – Иконка игры

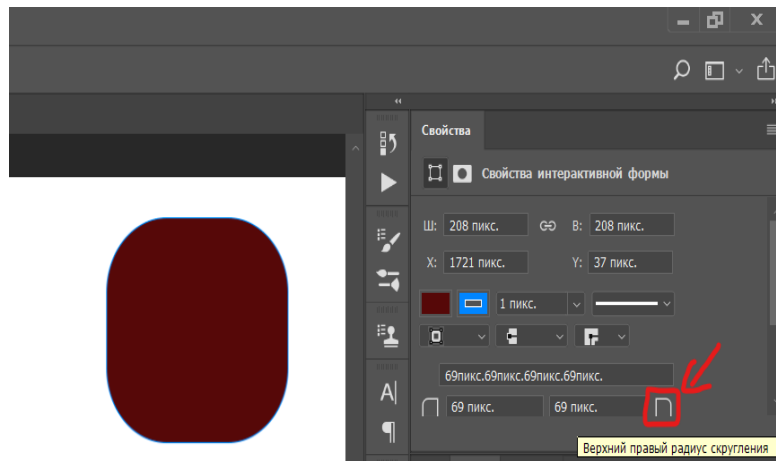


Рисунок 10 – Иконка игры

Далее, можно настроить градиентную заливку фона логотипа. Для этого, необходимо выбрать Градиент (рисунок 11), установить диапазон цветов и настроить уровень градиентной заливки (рисунок 12).

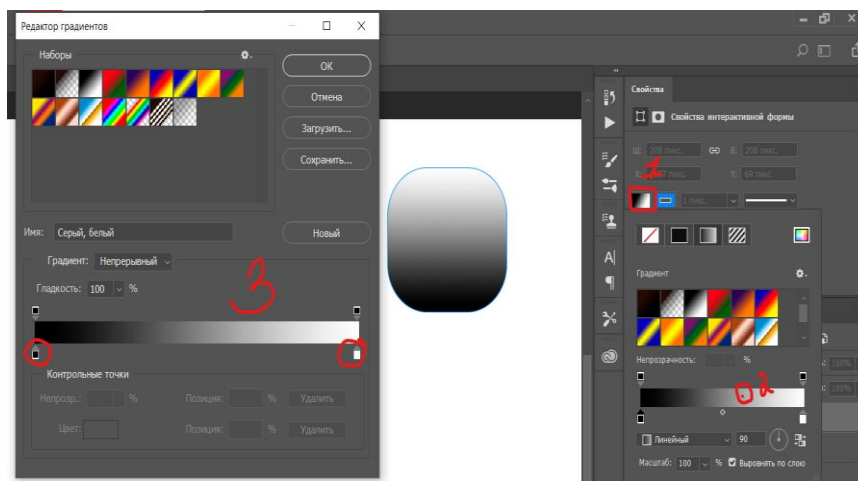


Рисунок 11 – Иконка игры

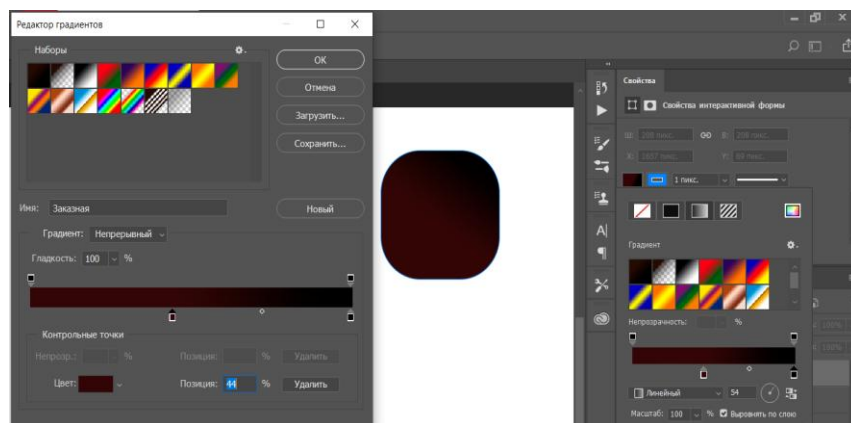


Рисунок 12 – Иконка игры

После вышеописанных действий, формируется готовый шаблон для иконки. В разных проектах, иконки содержат инициалы игры, небольшой логотип игры или картинку во всю иконку (рисунок 13).

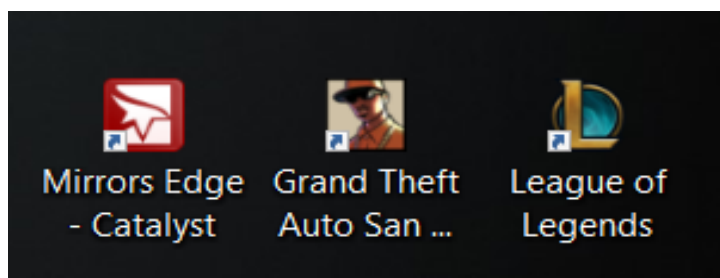


Рисунок 13 – Пример иконок других игр

Для минимализма, иконка данного проекта будет содержать инициалы игры «E» и «N». В качестве уникальности, буква «E» будет зеркально отражена и буквы будут разного цвета (рисунок 14).

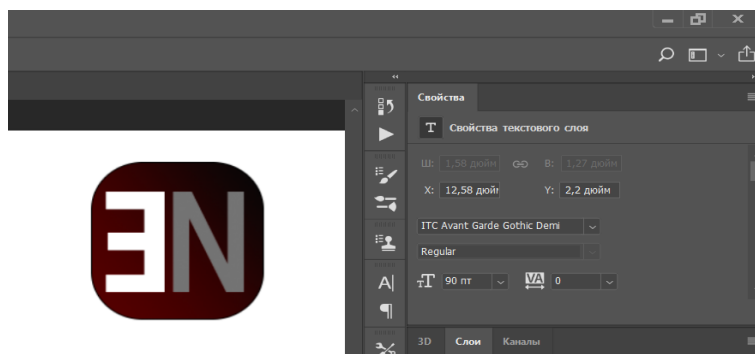


Рисунок 14 – Иконка игры

На рисунке 15 представлена вариация логотипа с названием игры, так, как бы это выглядело на рабочем столе ПК или ноутбука.



Рисунок 15 – Иконка игры

Готовая иконка представлена на рисунке 16.



Рисунок 16 – Итоговая иконка игры

Ниже, на рисунке 17 представлена примерная визуализация, как иконка отражалась бы в панели задач Windows.



Рисунок 17 – Отображение иконки в панели задач

После каждого запуска игры, должна произойти его загрузка, во время загрузки, обычно, на экране ничего нет, лишь чёрный фон. Чтобы избавиться от «скучного» и «отталкивающего» чёрного фона, надо реализовать загрузочный экран, с красивым логотипом и фоном загрузки.

Для реализации загрузочного экрана, было использовано графическое ПО Procreate, работающее на планшете Apple iPad (только на операционной системе iPadOS) в связке со стилусом Apple Pencil.

На начальном этапе необходимо подготовить фон загрузочного экрана. Оптимальным вариантом, будет серый оттенок (рисунок 18).

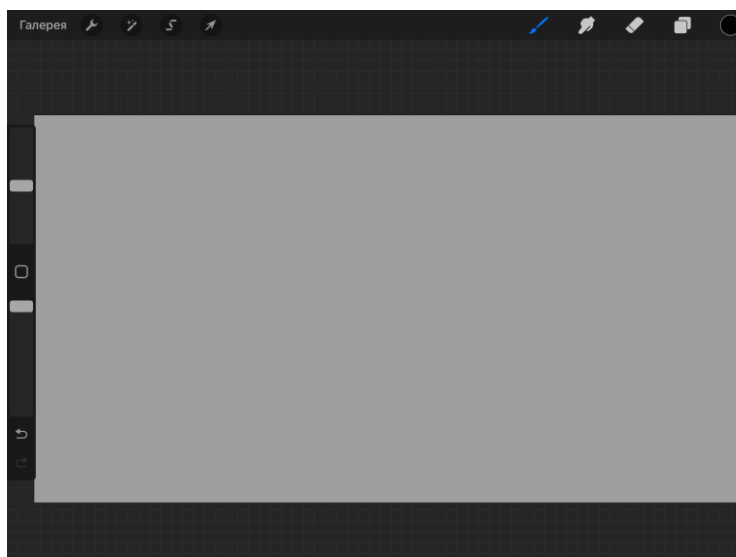


Рисунок 18 – Создание фона в Procreate

Далее, нужно добавить текст, название игры (рисунок 19-20). Для этого, нужно коснуться кнопки гаечного ключа в левом верхнем углу экрана, чтобы открыть меню «Действия», затем «Добавить» > «Добавить текст».

Текстовое поле будет добавлено на холст. После добавления текстового поля, можно его переместить на любую область на холсте (слое) или сразу же начать печатать. Для визуального оформления, текст будет размещен по центру слоя.

Шрифт ITC Avant Garde Gothic Demi. В данном проекте использовался размер шрифта 90 пт.

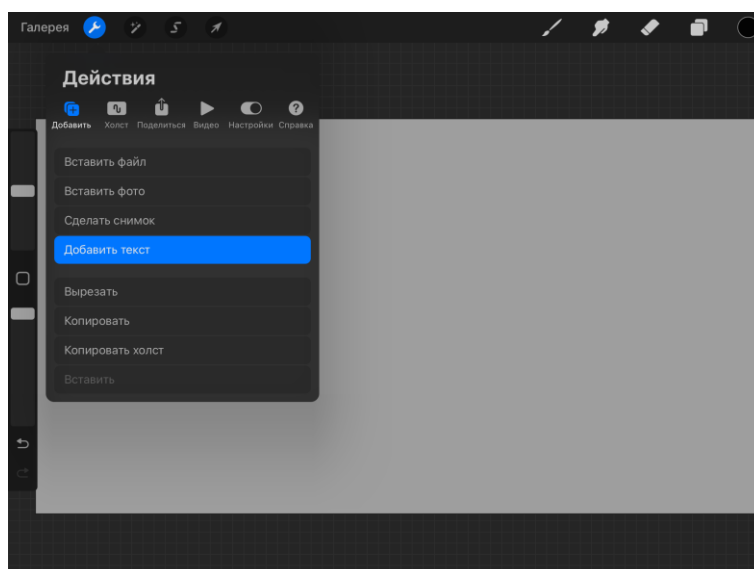


Рисунок 19 – Создание фона в Procreate

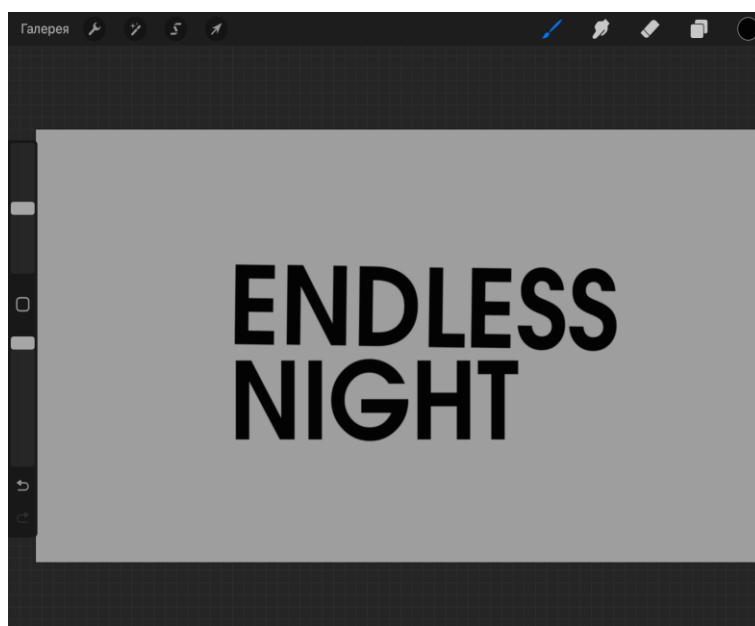


Рисунок 20 – Создание фона в Procreate

После добавления текста, необходимо как-то украсить его. Поэтому, используя стилус Apple Pencil, добавляем небольшие узоры (рисунок 21). Для оформления, добавлялись узоры в начале и в конце текста, а после, под буквами «SS» была закрашена пустая область. После чего, необходимо добавить новый слой, расположить его ниже слоя с текстом, чтобы добавить новые элементы декорации для текста [28, с.429].

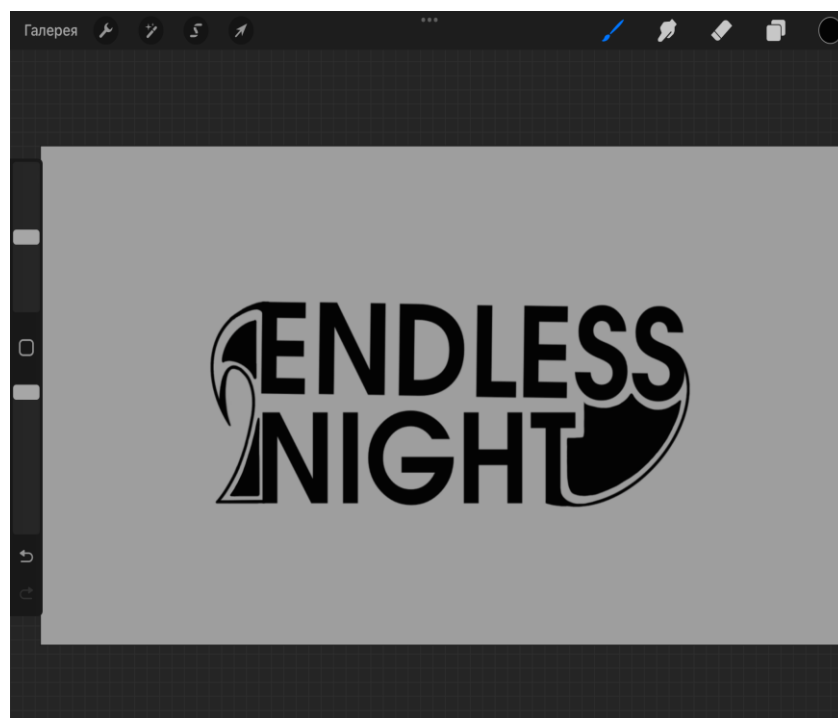


Рисунок 21 – Создание фона в Procreate

В Procreate есть 18 дефолтных наборов кистей, которые уже встроены в программу. В зависимости от своих характеристик, кисти разбиты на наборы. Некоторые кисти реалистичные, некоторые - фантастические. Идея этих кистей состоит в том, чтобы у пользователя данной программы был набор разных инструментов [12]:

- **Наброски.** Кисти предназначены для практического рисования, планирования, черчения и быстрого обучения. Имитируют графитовые карандаши, технические карандаши, пастель и мелки для художников

- **Чернила.** Кисти идеально подходят для рисования в стилях, где используются чернила, например, sumi-e. Включают разные стили чернил и кистей, технические ручки, маркеры, гелевые ручки и эффекты сухих чернил

- **Рисунок.** Набор кистей для художественного рисования - универсальное сочетание влажных и сухих сред, идеально подходит для быстрой учебы и рисования

- **Краски.** Набор содержит кисти от акриловых до штукатурных и масляных, а также разные варианты поведения кисти, включая перетаскивание старой кисти и мазок масляной кисти, заполненной скипидаром

- **Художественные.** Кисти умеют накапливать или удалять цвет, чтобы показать основную текстуру бумаги или холста. Набор содержит широкий спектр эффектов от водянистых до комковатых

- **Каллиграфия.** Кисти разных текстур с высокой чувствительностью к давлению, и высоким Streamline. Помогают создавать плавные и каллиграфические формы

- **Аэрограф.** Набор аэрографов, позволяет художникам быстро создавать чистые цвета и плавные градиенты. У аэрографа для моделизма одинарного действия (single action) кнопка открывает только подачу воздуха, краска из резервуара подается в точно определенном, установленном заводскими параметрами количестве

- **Текстуры.** Универсальный набор для закрашивания текстурой больших площадей. Здесь есть как реалистичные текстуры, имитирующие

поверхности, так и цифровые узоры: викторианские узоры, математические сетки, мозаики для использования в дизайне. В Artec Studio реализовано два режима выделения, которые отличаются от обычных кистей тем, как выбирается область для удаления

- Абстракция. Игровой набор для демонстрации необычных возможностей кисти в Procreate. Содержит чувствительные к давлению кисти, которые меняют цвет, становятся светлее, создают дымчатый эффект или хаотичные формы

- Угольки. Реалистичные угольные кисти, идеально подходят для обучения, портретов и реалистичного рисования

- Элементы. Эффекты, которые мгновенно создают дым, пламя, воду, облака и многое другое

- Аэрозоли. Разнообразные брызги и эффекты распыления

- Подкраска. Тонкие штрихи, такие как волосы, щетина, шум и несколько текстур кожи, чтобы оживить нарисованный концепт-арт

- Ретро. С помощью этого винтажного набора кистей можно добавить ретро-текстур и различных элементов в дизайн

- Освещенность. Кисти использует возможности цифрового рисования для создания целого ряда ярких световых эффектов от импульсов до галактик, блесток и боке

- Индастриал. Набор разработан для быстрого добавления реалистичных гранж-текстур. Эффекты бетона, металла, камня и ржавчины

- Органика. Используется для быстрого добавления травы, листьев и текстур коры. Также содержит кисти, имитирующие штрихи, созданные натуральными материалами: чернила с тростником, рисование бамбуком, рисование кистью соболя

- Вода. Набор содержит несколько реалистичных акварельных кистей, а также кисти брызг, клякс, пятен и капель

В данном случае, будет использоваться кисть «Клякса» из набора «Аэрозоли» (рисунок 22). Цвет кисти будет красным

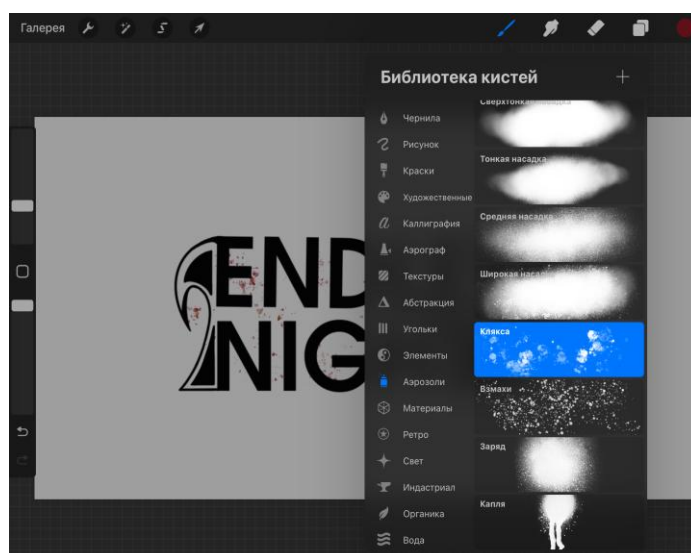


Рисунок 22 – Создание фона в Procreate

Далее, для акцентирования внимания логотипа, можно добавить ограничивающие полосы технологичного стиля. Это придаст более привлекательный вид загрузочному экрану и «психологически» сконцентрирует внимание пользователя на середину этих линий, иначе говоря – на текст [18, с.428]. Результат можно наблюдать на рисунке 23.

После, на новом слое, добавим третий элемент декора текста. Для этого будет задействоваться кисть «Иглы», набора «Абстракция». На рисунке 24 можно увидеть добавленные элементы на слое.

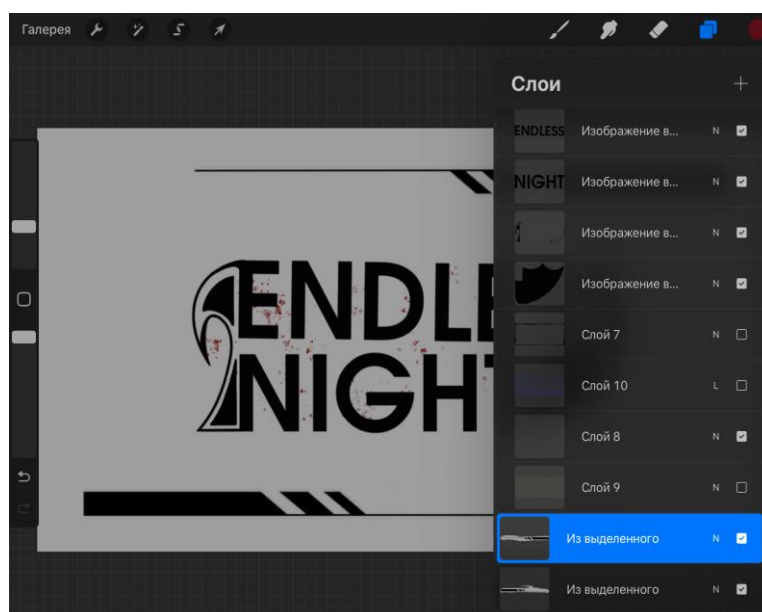


Рисунок 23 – Создание фона в Procreate

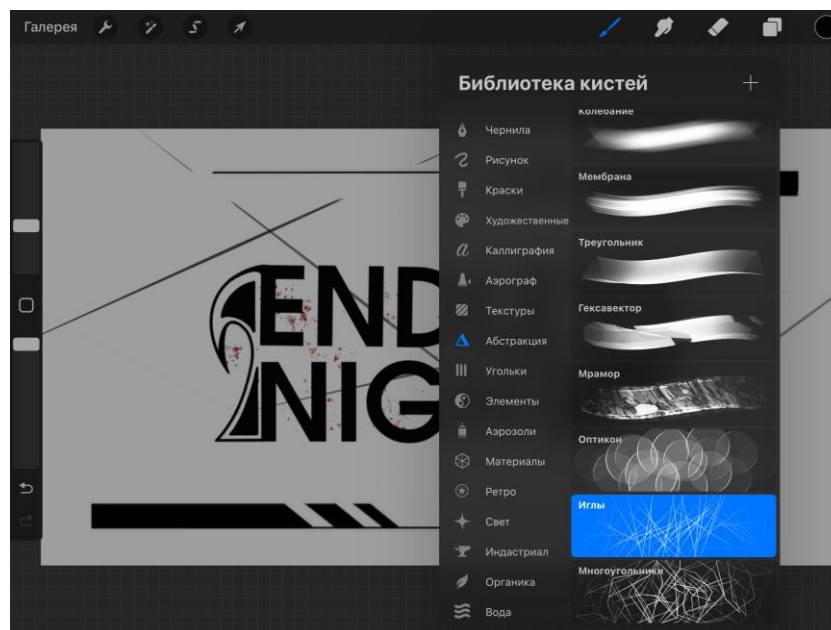


Рисунок 24 – Создание фона в Procreate

Последним элементом декора, будет заполнение пустого пространства на заднем фоне текста. Для этого, можно выбрать кисть «Кубики», набора кистей «Текстуры». Цвет кисти синий. На рисунке 25 отображено заполнение заднего фона, с помощью данного инструментария набора кистей.

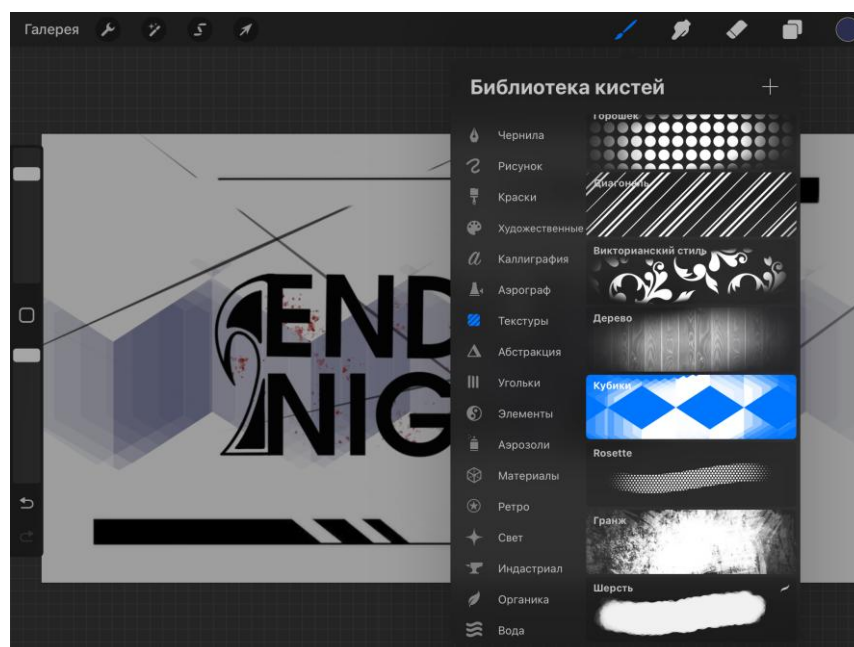


Рисунок 25 – Создание фона в Procreate

Для гибкого редактирования загрузочного фона, весь холст со слоями из Procreate был экспортирован в программу Adobe Photoshop [1]. После чего,

была добавлена дополнительная надпись «Загрузка», шрифтом Comic Sans MS, Regular с размером 48 пт (рисунок 26).

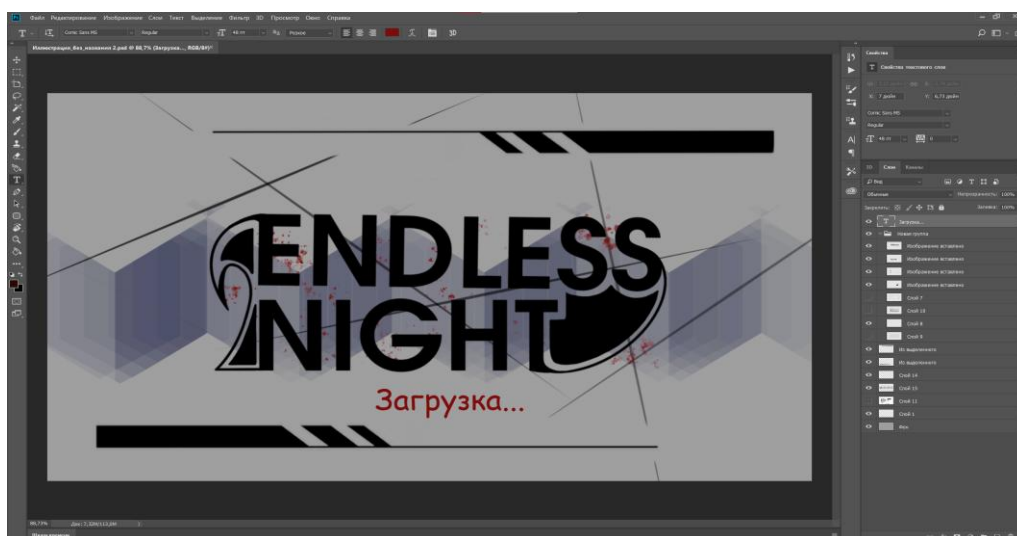


Рисунок 26 – Создание фона в Procreate

Готовый загрузочный экран представлен на рисунке 27.



Рисунок 27 – Название и логотип игры

Так же, помимо загрузочного фона игры, необходимо реализовать внутриигровой логотип. В игровой индустрии, это обязательная практика [21, с.20]. За основу внутриигрового логотипа, можно взять части уже существующих элементов загрузочного экрана. Рисование мини-логотипа будет продолжено в Procreate. Для этого, в параметрах слоев нужно скрыть все слои, кроме текста (рисунок 28).



Рисунок 28 – Создание логотипа в Procreate

С помощью инструмента «выделение», можно ограничить область, в которой будут применяться изменения. Например, в заштрихованной зоне рисовать не получится, внутри заштрихованной зоны – можно.

На рисунке 29 отчетливо видно примененное ограничение области слоя. Это нужно для рисования рамки вокруг текста. Поэтому, выбираем любую кисть, типа «Аэрограф» и обводим контур ограничивающего выделения. Результат можно увидеть на рисунке 30.



Рисунок 29 – Создание логотипа в Procreate



Рисунок 30 – Создание логотипа в Procreate

В слоях холста нужно включить отображения слоя с красными кляксами, для получения результата, как на рисунке 31.

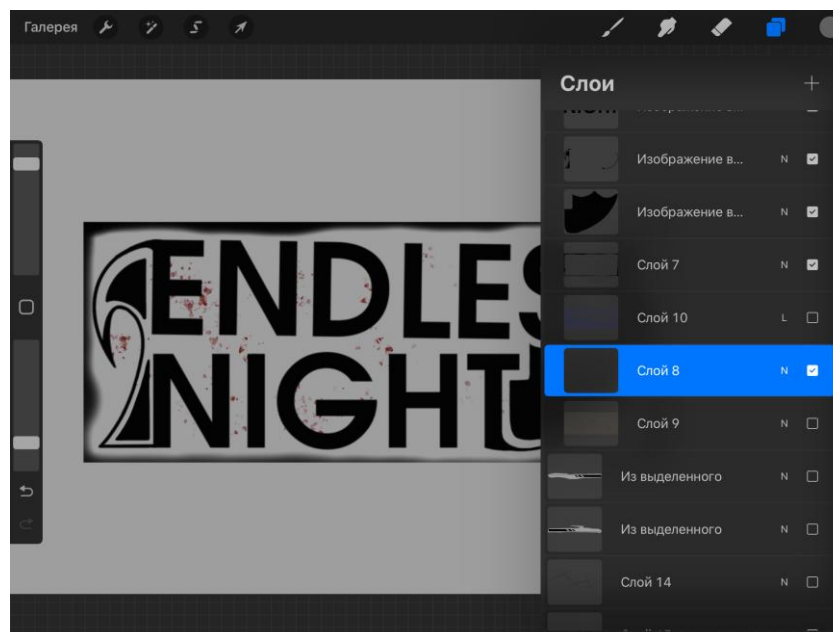


Рисунок 31 – Создание логотипа в Procreate

Для удобства оформления внутриигровых окон, дальнейшее проектирование будет продолжено в программе Adobe Photoshop. В качестве упрощения, по управлению слоями в программе, нужно разделить слои на группы. Так как в игре будет много диалоговых окон, локаций и персонажей,

нужно создать группы «Диалоговое окно», «Герой» и «Фон». Так же, стоит учесть текст и оформления загрузочных экранов. В связи с этим, нужно создать дополнительно группу слоев «Текст» с подгруппами «Загрузочный», «Текст меню» и «Текст главное меню» (рисунок 32).

Таким образом, это позволит «включать» и «отключать» определенные слои для построения интерфейса. Более подробные примеры применения данного подхода проектирования, будут описаны дальше.

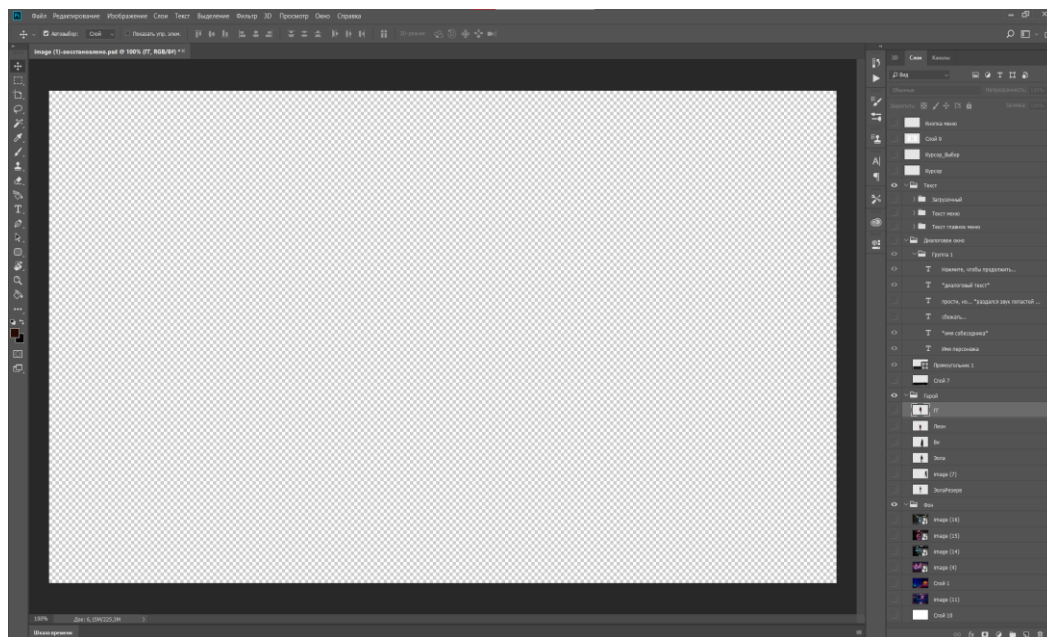


Рисунок 32 – Проектирование интерфейса в Adobe Photoshop

После загрузки, должно появляться «Главное меню».

Внешний вид экрана «Главное меню» будет состоять из стиля футуризма, для соответствия основному стилю игрового процесса, но в тоже время и не слишком строгим, чтобы не вызвать дискомфорт у будущих пользователей.

Для этого, нужно добавить элементы в следующие группы:

- в подгруппу «Текст главное меню» добавляем текстовые слои «Главное меню», «Загрузить в игру», «Новая игра», «Настройки», «Об авторах», «Выход», слой с мини-логотипом игры и слой с полупрозрачным красным прямоугольником (для выделения выбранного текста в главном меню)
- в группу «Фон» добавляем картинку, которая будет использоваться в качестве заднего фона главного меню»

Результат оформления главного меню можно наблюдать на рисунке 33.

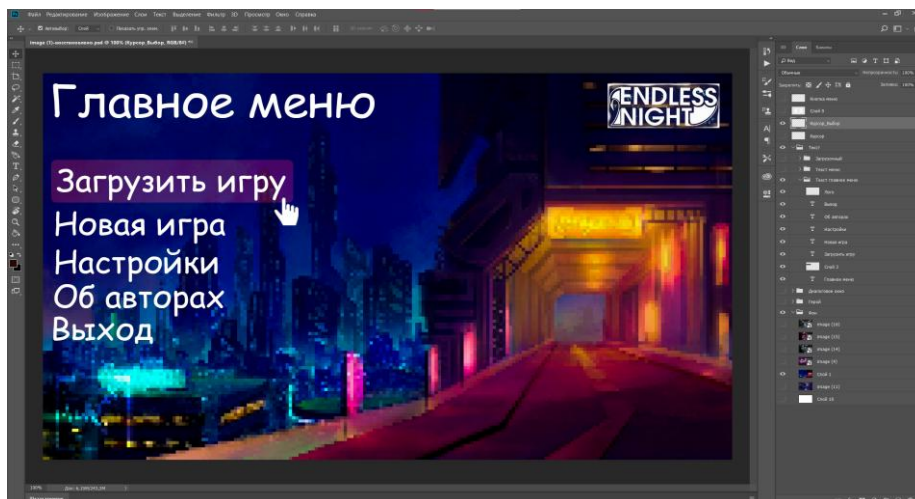


Рисунок 33 – Проектирование интерфейса в Adobe Photoshop

После нажатия «Загрузить игру», произойдет загрузка сохранения. В этот момент, должно отобразиться загрузочное окно. В проекте программы нужно выключить отображения всех слоев (при отключении видимости группы слоев, автоматически отключается видимость всех слоев внутри группы). Добавляем подгруппу «Загрузочный» фон экрана (чтобы по щелчку видимости группы, сразу включать весь загрузочный экран, а не только текст), полупрозрачный треугольник красного оттенка (для создания эффекта тонированного стекла, через который открывается взор на город) и текстовый слой «Загрузка...» в правом нижнем углу экрана [23, с.174].

Результат данного комбинирования можно увидеть на рисунке 34.



Рисунок 34 – Проектирование интерфейса в Adobe Photoshop

В проекте программы нужно выключить отображения всех слоев. После чего, необходимо добавить в группу «Фон» картинки остальных локаций (рисунок 35-39).



Рисунок 35 – Фоны игровых локаций

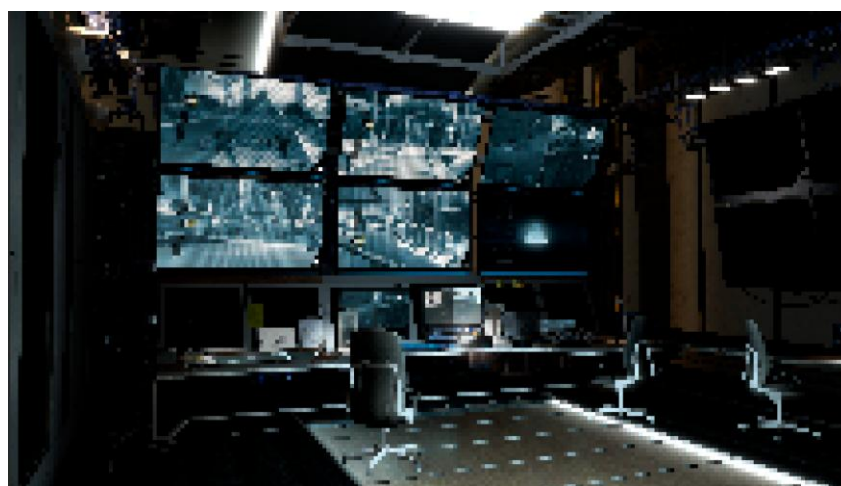


Рисунок 36 – Фоны игровых локаций

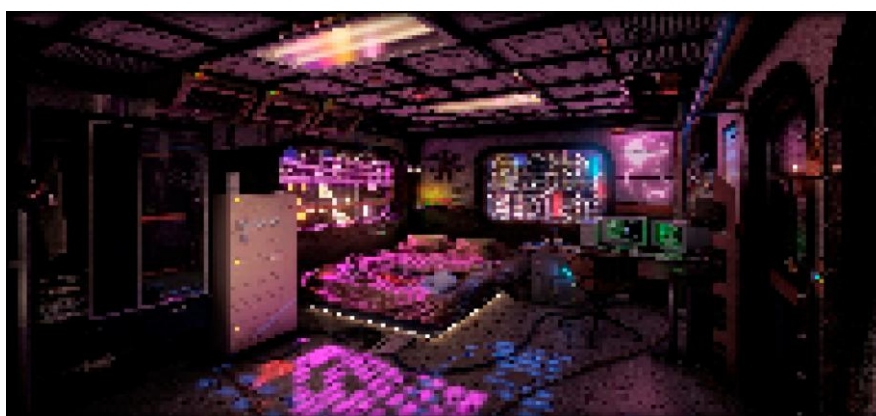


Рисунок 37 – Фоны игровых локаций



Рисунок 38 – Фоны игровых локаций



Рисунок 39 – Фоны игровых локаций

Далее, необходимо добавить персонажей в группу «Герой» (рисунок 40).



Рисунок 40 – Основные персонажи игры

На рисунке 41 отображена одна из игровых зон. В этой области достаточно места, для расположения главного героя и достаточно объектов для взаимодействия.

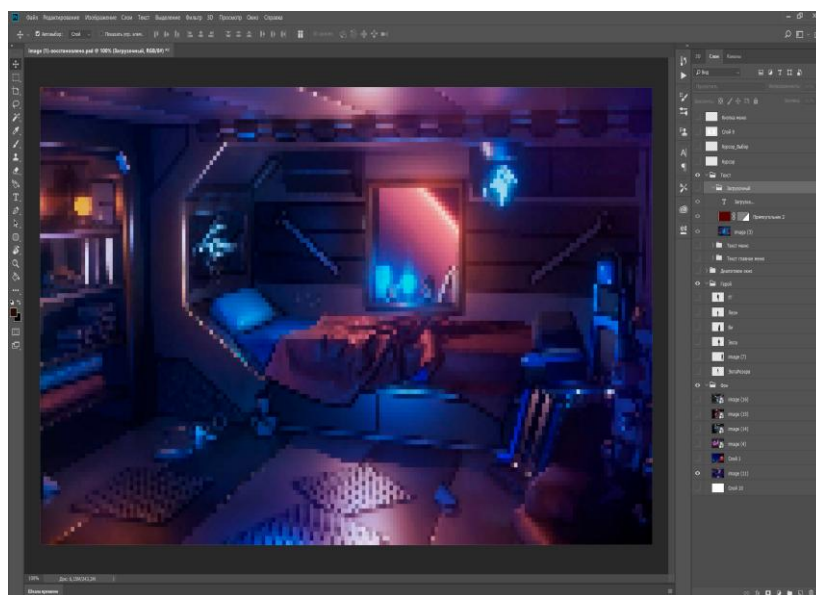


Рисунок 41 – Проектирование интерфейса в Adobe Photoshop

Далее, необходимо включить игрового персонажа в общую область (рисунок 42).

После, необходимо расставить элементы интерфейса, например, кнопку «меню», с помощью которой можно сохранить прогресс, настроить громкость звука или выйти из игры (рисунок 43).

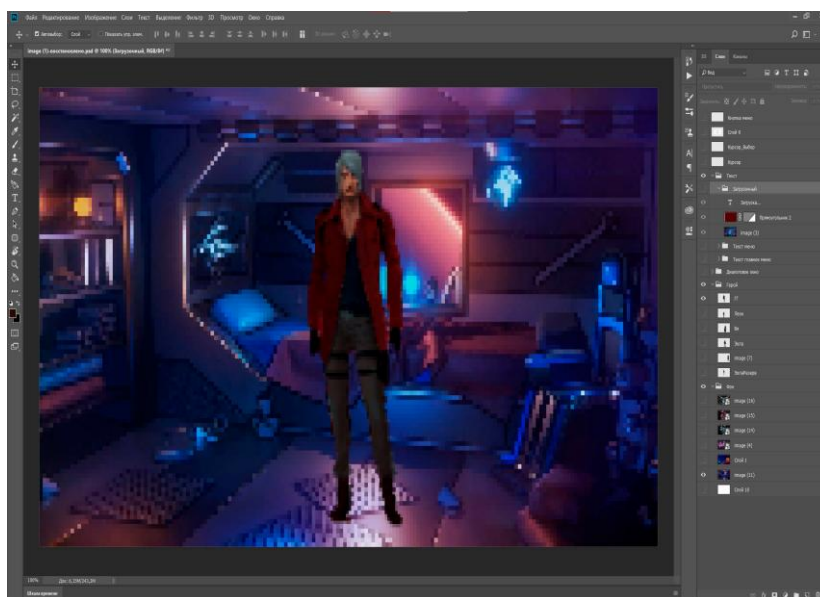


Рисунок 42 – Проектирование интерфейса в Adobe Photoshop

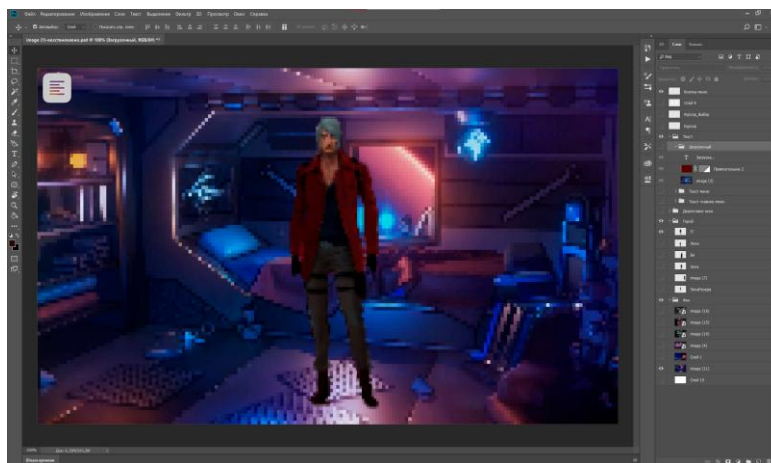


Рисунок 43 – Проектирование интерфейса в Adobe Photoshop

В игре, планируется, реализовать диалоги между главным героем и неигровыми персонажами [9].

Либо, в определенных ситуациях, главный герой будет свидетелем общения между группой неигровых персонажей. Диалоговые окна будет иметь эффект полупрозрачности с чёрным фоном и белым оформлением текста.

Пример диалогового окна можно увидеть на рисунке 44.

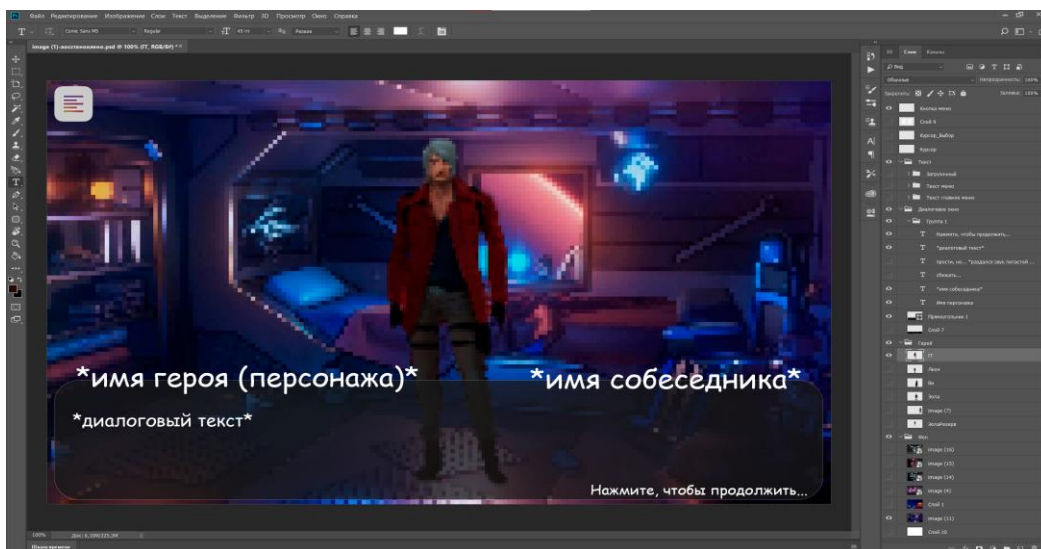


Рисунок 44 – Проектирование интерфейса в Adobe Photoshop

В любой момент, игру можно поставить на паузу, нажав на кнопку «Меню». Экран меню должен составлять пункты «Продолжить», «Сохранить», «Настройки» и «Выход». Для контраста, можно затемнить фон меню, используя эффект полупрозрачности [26, с.211].

Визуальное оформление экрана меню представлено на рисунке 45.

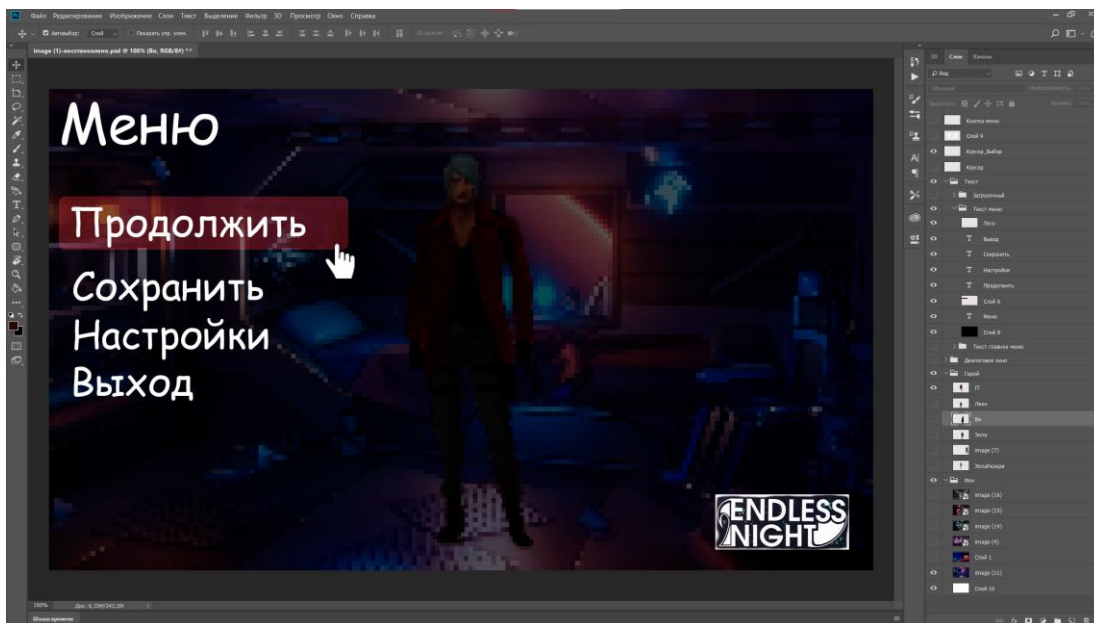


Рисунок 45 – Проектирование интерфейса в Adobe Photoshop

Так же, игра не ограничивается только этим персонажем, на текущем этапе их 4 и все они являются ключевыми, соответственно, с ними связан сюжет и игровые диалоги, пример остальных персонажей. На рисунке 46 представлен внешний вид и характер персонажа – Леон Уайт. На рисунке 47 представлен внешний вид и характер персонажа – Эола Уайт. На рисунке 48 представлен внешний вид и характер персонажа – Адам Ви.

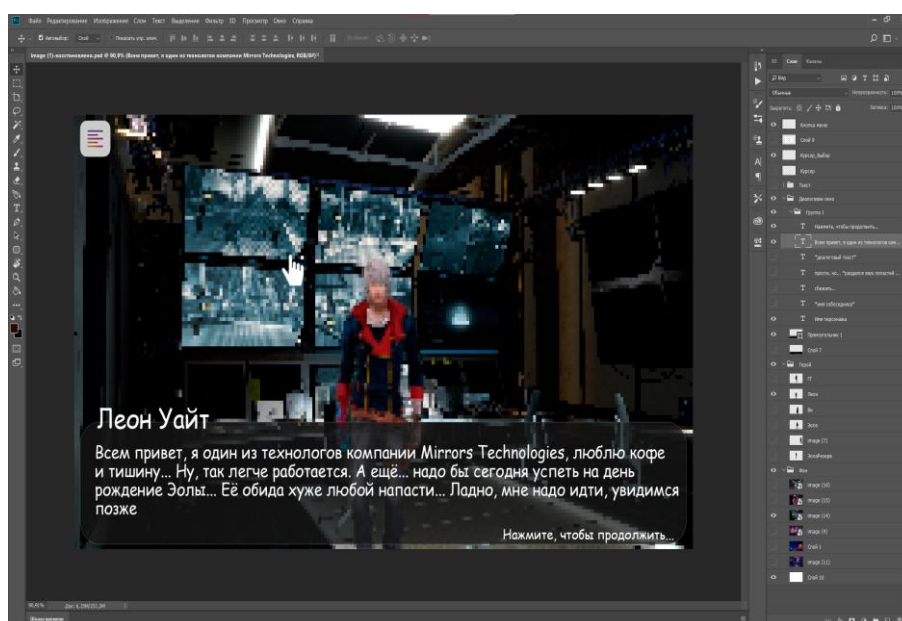


Рисунок 46 – Внешний вид персонажа Леон Уайт

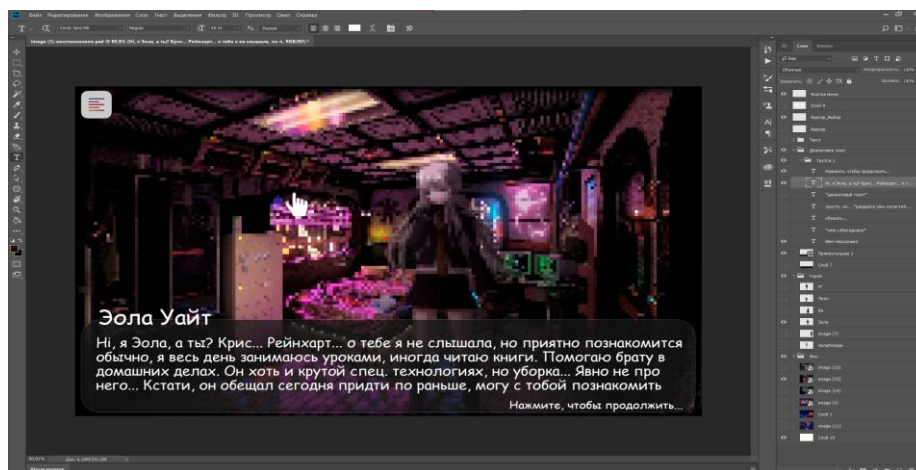


Рисунок 47 – Внешний вид персонажа Эола Уайт

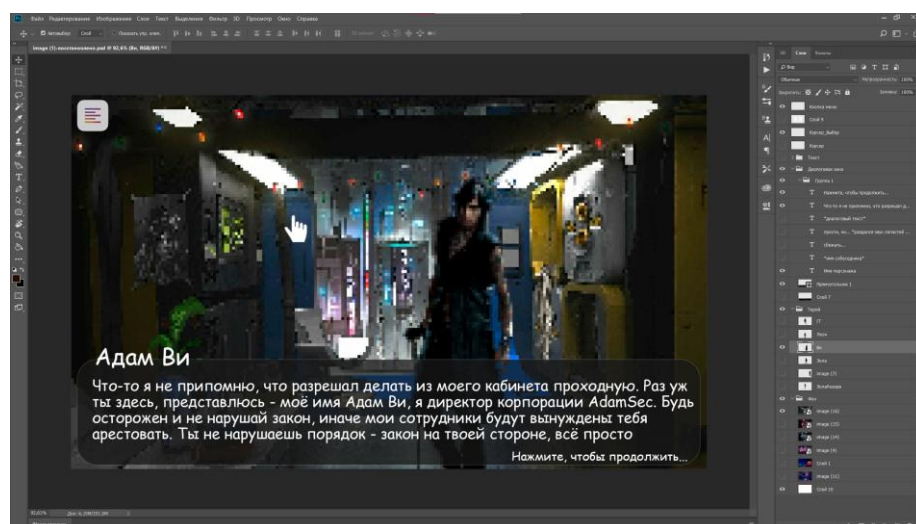


Рисунок 48 – Внешний вид персонажа Адам Ви

3.4 Проектирование сюжета игры

В этом разделе, хотелось несколько рассказать о сюжете игры. Название «Endless Night», было выбрано не просто так. В игровом мире, действия оборачиваются возле города «SkyCity». Так сложилось, что даже в дневное время, над городом царит ночь, но, несмотря на это, жизнь в городе процветает и ничуть не хуже в классических городах 2077 года. Главного героя, как уже было показано в предыдущих разделах, зовут Крис Рэйнхарт. Он работает детективом и, так как преступность в городе снизилась до минимума, благодаря корпорации AdamSec, за текущий год не было ни одного крупного

преступления или какого-нибудь разбоя с летальным исходом. В простанароде даже появился термин «Осторожно, Адам следит за тобой». Поэтому, Крису ничего не остается, как расследовать мелкие преступления, либо заняться поиском пропавших людей или предметов. Живёт он один, на окраине города, в одном из небоскребов. С окна его комнаты, открывается красивый вид на город. Порой может показаться, что весь город – это один, большой торт, глубоко спрятанный в тёмной комнате и единственное, что освещается всё в округе – это бесчисленное множество горящих свечей. В один день, Крис вышел на очередное дело о пропавшем человеке. Придя на место расследования, он нашёл загадочное письмо с координатами и странной надписью: «Этот город таит главную ложь – день, всё же реален, ибо тьма не может править вечно». Начиная с этого дня, жизнь главного героя начнет меняться и начнется самое крупное расследование, которое было в его жизни – найти ответ на вопрос «Когда Солнце вновь воссияет над нами?». Исследуя городские тайны, он встретит множество персонажей, некоторые из них станут его друзьями, а некоторые обратят на него свой взор, но...к добру ли это?

На данный момент, в игре присутствуют несколько ключевых персон – это Адам Ви, Леон и Эола Уайт. Так же, не исключены и сюжетные повороты, в которых герой попадает в очень неприятную ситуацию (рисунок 49). В конце истории, правда выходит в свет и, спустя много десятилетий, город увидел лучи света.



Рисунок 49 – Сцена из игры

Заключение

Были проанализированы популярные средства разработки. В ходе анализа, было проведено их сравнение и выбраны наиболее актуальные средства разработки для начинающих разработчиков. Выбор приоритетных средств разработки проходил по двум критериям: доступность и функциональность.

В ходе анализа стало ясно, что при разработке компьютерной игры с простой игровой механикой, стоит обратить внимание на дополнительные элементы игры, такие как сюжет и графическое оформление. Это нужно для того, что бы удержать потенциального игрока и продлить жизненный цикл разработки.

Целью данного проекта было проектирование концепта казуальной компьютерной игры жанра «Point & Click» в стиле футуризма с элементами обзора 2D платформера. Данная цель была декомпозирована на задачи:

1. Проанализировать предметную область проекта
2. Проанализировать программные средства и технологии создания игр
3. Сформировать требования к игровому ПО
4. Спроектировать эскиз визуальной составляющей игры

В проекте были рассмотрены и проведены анализы требований к программному продукту, проектированию.

Для этих целей были использованы различные методы и средства, зарекомендовавшие себя в мировой практике разработки программных продуктов.

Основываясь на всей полученной в ходе исследования информации, было решено спроектировать прототип двумерного платформера для одного игрока на игровом движке Unity. Такое решение было принято по нескольким причинам:

1. Двумерная графика, в отличие от трехмерной легче в создании
2. По игровой механике, игра жанра платформер проще реализуется

3. Игровой движок Unity распространяется бесплатно и позволяет разрабатывать приложения на языке программирования C#

Освоение среды разработки Unity несет не маловажный характер, так как в современном мире индустрия разработки игр все сильнее распространяется в нашем обществе. Игры перестали быть лишь предметом для развлечений, и теперь используются и в других областях, например, в науке или в обучении пользователей. Поэтому развитие в данном направлении можно считать одним из самых важных в современном обществе. Задачи, в рамках текущей работы, были выполнены в полном объеме.

Список литературы

1. ADOBE PHOTOSHOP ELEMENTSHELP [Электронный ресурс]. Режим доступа . URL:https://helpx.adobe.com/pdf/photoshop_reference.pdf. (дата обращения: 25.11.2022).
2. bibliofond. ru_ Game [Электронный ресурс]. Режим доступа. URL: <https://www.bibliofond.ru/view.784285#text> (дата обращения: 03.10.2022).
3. fr. flossmanuals. net [Электронный ресурс]. Режим доступа. URL: <https://fr.flossmanuals.net/creating-point-and-click-games-point-and-click-games/> (дата обращения: 05.12.2022).
4. GameMaker.ru [Электронный ресурс]. Режим доступа. URL: <https://www.yooyogames.com/> (дата обращения: 08.12.2022).
5. Gamesisart.ru [Электронный ресурс]. Режим доступа. URL: https://gamesisart.ru/game_.html#Game_Create_1 (дата обращения: 15.11.2022).
6. JMonkey [Электронный ресурс]. Режим доступа. URL: <https://jmonkeyengine.org/> (дата обращения: 08.12.2022).
7. Marmalade. madewith [Электронный ресурс]. Режим доступа. URL: <https://www.madewithmarmalade.com/> (дата обращения: 08.12.2022).
8. Mmo13.ru - Game [Электронный ресурс]. Режим доступа. URL: <https://mmo13.ru/games/feature/-click> (дата обращения: 20.11.2022)
9. OGRE 3D [Электронный ресурс]. Режим доступа. URL: <http://www.ogre3d.org/> (дата обращения: 09.12.2022).
10. Shiva/Game. [Электронный ресурс]. Режим доступа. URL: <http://www.shiva-engine.com/> (дата обращения: 08.12.2022).
11. SIO2.Game [Электронный ресурс]. Режим доступа. URL: <http://www.sio2interactive.com/index.php> (дата обращения: 08.12.2022).
12. Teletype.in [Электронный ресурс]. Режим доступа. URL: <https://teletype.in/@karaponich/4yLXgха-5> (дата обращения: 02.11.2022).
13. Turbulenz [Электронный ресурс]. URL: <http://biz.turbulenz.com/home> (дата обращения: 09.12.2022).

14. Unity3d [Электронный ресурс]. Режим доступа. URL: <https://unity3d.com/> (дата обращения: 10.12.2022).
15. Unreal Engine [Электронный ресурс]. Режим доступа. URL: <https://www.unrealengine.com> (дата обращения: 01.12.2022).
16. Vgtimes.ru [Электронный ресурс]. Режим доступа. URL: <https://vgtimes.ru/games/genres /point-and- click/windows- pc/sort-rating/> (дата обращения: 17.11.2022).
17. Workspay.ru [Электронный ресурс]. Режим доступа. URL: <https://workspay.ru/work/30171/> (дата обращения: 20.11.2022).
18. Алим, С., Капретц, Л.Ф., Ахмед, Ф. Критические факторы успеха для улучшения процесса разработки игр с точки зрения разработчика. - *Comput Sci Technol* 31 (5), 2019. - 950 с.
19. Ахмед, Ф., Капретц, Л.Ф. Модель бизнес-зрелости разработки линейки программных продуктов. Информационные системы. - *Frontiers, Springer*, 2018. - 560с.
20. Воробкалов, П.Н., Катаев, А.В., Шабалина, О.А. Обучение разработчиков программного обеспечения: применение компьютерных игр и процесса их разработки. - *Известия ВолгГТУ*, 2020. - № 11. - С. 117-124с.
21. Вулф, Т. Кодзима – гений (Легендарные компьютерные игры). – М.: ЭКСМО, 2018. - 309 с.
22. Гвоздева, Т.В. Проектирование информационных систем. Стандартизация: учеб. пособие. - СПб.:Лань, 2019. – 252 с.
23. Дыбская, В.В. Проектирование системы распределения в логистике: Монография / В.В. Дыбская. - М.: Инфра - М, 2019. - 277с.
24. Касихин, В.В. Как стать создателем компьютерных игр. Краткое руководство. - М.: Вильямс, 2019. – 208 с.
25. Майк Гейг. Разработка игр на unity 2018 за 24 часа. – М.: ЭКСМО, 2020. – 466 с.
26. Петров, М. Н. Компьютерная графика: учеб. для вузов. - Спб.: Изд-во «Питер», 2021. – 544 с.

27. Старикова, Ю.С. Основы дизайна. Конспект лекций. - М.: Изд-во «А-Приор», 2020. – 112 с.
28. Финни, К. 3D-игры. Все о разработке. - М.: Изд-во «Бином. Лаборатория знаний», 2019. – 973 с.
29. Шарп, Джон. Microsoft Visual C#. Подробное руководство. 8-е изд. – СПб.: Питер, 2018 – 848 с.
30. Шелл, Д. Геймдизайн. – М.: «Альпина Диджитал», 2019. – 550 с.