Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.03 Программирование в компьютерных системах

**ОТЧЕТ**

**ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

**(преддипломной)**

Выполнил:

обучающийся учебной группы № 4ПКС-219

Смирнов Е. А.

*(И.О. Фамилия)*

Проверил:

руководитель практики от колледжа:

Сибирев И. В.

*(И.О. Фамилия)*

**Москва**

**2023**

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc135089303)

[ГЛАВА 1 ПРЕДПРОЕКТНОЕ ИССЛЕДОВАНИЕ 6](#_Toc135089304)

[1.1 Описание предметной области 6](#_Toc135089305)

[1.2 Сравнительный анализ программ-аналогов 7](#_Toc135089306)

[1.3 Постановка задачи 10](#_Toc135089307)

[1.4 Характеристика инструментальных средств разработки 10](#_Toc135089308)

[ГЛАВА 2 ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ ПРОГРАММЫ 14](#_Toc135089309)

[2.1 Анализ требований и разработка спецификаций 14](#_Toc135089310)

[2.2 Проектирование программного обеспечения 15](#_Toc135089311)

[2.3 Разработка программного обеспечения 16](#_Toc135089312)

[2.4 Отладка и тестирование программного обеспечения 22](#_Toc135089313)

[2.5 Руководство по использованию программы 23](#_Toc135089314)

[ЗАКЛЮЧЕНИЕ 27](#_Toc135089315)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 29](#_Toc135089316)

# ВВЕДЕНИЕ

В наше время индустрия видеоигр твёрдо удерживает позицию в списке популярных видов развлечений. Это и не удивительно, ведь с помощью игр можно не только лишь интересно провести своё свободное время, но и также получать новые знания. Игры позволяют человеку развить его реакцию, мышление, а также учат действовать в различных непредвиденных ситуациях.

Жанр action представляет собой один из популярных жанров среди всех видеоигр. Такую популярность он обрёл из-за своих отличительных особенностей, которые вместе создают захватывающий и динамичный геймплей. Главной чертой такого жанра, как раз является динамичность – у игрока просто не будет времени заскучать, так как почти постоянно на экране будут происходить разные события, начиная от неожиданных препятствий и заканчивая различными противниками, стремящимися помешать пользователю продвинуться дальше.

Чаще всего разработчики выпускают продукт, который содержит в себе несколько разных жанров, такое решение помогает разнообразить игровой процесс и не дать человеку быстро заскучать. Одним из таких жанров является survival horror.

Жанр survival horror (выживание в ужасе) — это жанр компьютерных игр, направленный на создание атмосферы страха и напряжения. В этих играх игрокам приходится сражаться за выживание во мрачных и угрожающих местах, где на каждом шагу подстерегают опасности и ужасы.

Герой игры должен умело использовать доступные ему средства и ресурсы для борьбы с врагами и решения головоломок, чтобы достичь своей цели. Среди общих черт жанра можно выделить наличие опасных и устрашающих противников, присутствие элементов экшена. Центральными концепциями этого жанра являются выживание и экология ресурсов – игроку необходимо бережно обращаться с оставшимися у него ресурсами и использовать их по мере необходимости, ведь они могут стать не только ценными для выживания, но и единственным способом обезвредить опасных противников. Таким образом, жанр survival horror предназначен для тех, кто ищет игры с мрачной и неприветливой атмосферой, интригой и опасностями на каждом шагу, и для тех, кто готов сражаться за свою выживаемость и постоянно находить новые способы оставаться в живых.

Тема исследования «Разработка 3D игры «Secret things» в жанре adventure с элементами «survival horror» актуальна в связи с популярностью компьютерных игр этого жанра.

Объект исследования: 3D игра.

Предмет исследования: технологии разработки приложений на Unity.

Главной целью проекта является создание игры, которая позволит человеку научиться быстро действовать и принимать решения в разных ситуациях.

Задачи ВКР: описать предметную область; произвести сравнительный анализ аналогов; охарактеризовать инструментальные средства разработки; разработать ландшафт; спроектировать и анимировать 3d модели; разработать скрипты; спроектировать и протестировать игру; разработать документацию.

Для предпроектного исследования будут применены следующие методы: анализ предметной области, сравнение существующих решений и моделирование нового решения.

В качестве источниковой базы исследования используем полное руководство по языку программирования С# и учебное пособие по методам и алгоритмам обработки данных.

Для разработки будут использоваться язык программирования C# следующие инструментальные средства: интегрированная среда разработки программного обеспечения Visual Studio.

Движок Unity. Unity 3D — это кроссплатформенный движок для разработки 3D и 2D игр, виртуальной реальности и приложений, поддерживающий множество платформ, включая Windows, Mac, Linux, Android, iOS, PlayStation, Xbox и другие. Unity 3D предоставляет инструменты для создания игровых моделей, текстур, анимации, эффектов, управления физикой объектов и многое другое. С помощью этого движка можно создавать как однопользовательские, так и сетевые игры. Движок используется в игровой индустрии, обучении, симуляциях и других областях, связанных с разработкой интерактивных приложений.

Программа MagicaVoxel. MagicaVoxel — это бесплатный кроссплатформенный редактор вокселей (3D пикселей). Он используется для создания визуальных объектов и мира в играх, анимации, CG-фильмах и других областях 3D-графики. MagicaVoxel позволяет создавать и редактировать воксельные модели 3D, а также давать им текстуры, освещение, эффекты и анимацию. Это мощный инструмент для создания прототипов игр или качественных визуализаций технических моделей, а также предлагает возможность экспорта в различные форматы, такие как vox, obj, ply и другие.

Программа Blender 3D. Blender 3D — это бесплатный и открытый 3D-редактор, используемый для создания анимации, визуализации, игр, мультфильмов и других цифровых проектов. Он доступен на различных платформах, включая Windows, Mac и Linux. Blender 3D предоставляет множество инструментов для работы с 3D-моделями, включая моделирование, текстурирование, освещение, рендеринг, анимацию и многое другое. Также с помощью Blender 3D можно создавать визуализации архитектурных проектов, редактировать видео, создавать спецэффекты и многое другое. Blender 3D является одним из самых популярных 3D-редакторов и имеет широкое сообщество пользователей, которые создают и обмениваются своим опытом и моделями.

Пользователями данной игры могут быть подростки.

# ГЛАВА 1 ПРЕДПРОЕКТНОЕ ИССЛЕДОВАНИЕ

1.1 Описание предметной области

Темой выпускной квалификационной работы является разработка 3D игры «Secret things» в жанре action с элементами survival horror.

В основе сюжета разрабатываемой игры лежит популярная в наше время серия игр FNAF (Five Nights at Freddy’s). Сюжет этой игры повествует об охраннике, устроившимся работать в пиццерию, где специальные роботы (аниматроники) развлекают детей. Но в ночное время эти роботы по неизвестной причине становятся агрессивными и начинают преследовать охранника. Самому игроку необходимо пережить 5 ночей, каждая из которых длится с 12 часов ночи и до 6 часов утра. Так как пользователь сидит в офисе, ему даётся возможность использовать различные функции для защиты себя в этой комнате (закрытие и открытие дверей, освещение дверных проходов, система видеонаблюдения). Сложность заключается в том, что энергия ограничена, поэтому держать двери всегда закрытыми не получится. Это делает игровой процесс интересным, так как приходится думать над всеми действиями и продумывать тактику, которая приведёт к победе.

Для создания игрового окружения будет использоваться идея «Закулисья» (ориг. «Backrooms»). Эта история рассказывает о различных случаях попадания людей в неизвестные пространства, напоминающие собой бесконечную сеть лабиринтов из офисных помещений. Чаще всего в этих помещения очень пусто и единственное, что можно встретить это некие существа, стремящиеся догнать человека. Существует множество разных уровней этого самого «закулисья», у всех них есть одна общая черта – они похожи на лабиринты и относятся к лиминальным пространствам. Лиминальное пространство (англ. Liminal space), — интернет-эстетика, запечатлевающая места, которые кажутся переходными, жуткими и зачастую сюрреалистичными.

Так, сюжет разрабатываемой игры повествует об охраннике в специальном комплексе по расследованию паранормальных явлений и объектов, которому необходимо продержаться в своей комнате до 6 часов утра пока компьютер обрабатывает все необходимые данные о имеющихся в этом здании неизвестных сущностях. Но дело в том, что в этом комплексе в ночное время весь персонал уходит в специальные безопасные зоны, так как населяющие данное пространство существа становятся активными и пробуждаются. Поэтому игроку будет необходимо защищаться с помощью специальной системы защиты, в которую входят: камеры видеонаблюдения, управление дверьми в своём офисе, дрон для получения доступа к удалённым участкам комплекса и для взаимодействия с определёнными объектами. Важным упомянуть, что энергия в комплексе ограничена, поэтому использовать систему защиту нужно аккуратно и расчётливо.

1.2 Сравнительный анализ программ-аналогов

Для сравнительного анализа были найдены игры, сравнение которых показано в таблице 1.

Таблица 1. Сравнительный анализ игр

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название | Тип графики | Наличие разных механик | Наличие различных NPC | Сюжет |
| The Complex: Found Footage | 3D | нет | нет | Сюжет отсутствует |
| FNAF 1 | 2D/3D | Есть возможность вести наблюдение через камеры, а также закрывать и открывать двери | Существует 3 персонажа | Сюжет рассказывается через звонки и другие элементы повествования. |

Изображение выглядит как текст, снимок экрана, Цифровая сборка, Компьютерная игра

Автоматически созданное описание

Рисунок 1. Игра Five Nights at Freddy's

Вот основные механики игры FNAF 1:

- Игрок должен следить за камерами видеонаблюдения, чтобы увидеть, где находятся куклы, и незамедлительно закрывать двери, если кто-то приблизится.

- Игрок должен экономно использовать источник энергии на охранной панели, чтобы не допустить слишком быстрого разряда и остаться в темноте.

- Если куклы окажутся достаточно близко к игроку, они могут напасть и убить его.

- В каждую из пяти ночей в игре, новые роботы добавляются в сводке на камере, и уровень сложности возрастает.

- Игрок всегда находится внутри офиса, и должен оценивать ситуацию с помощью камер видеонаблюдения и звуковых эффектов.

Из минусов можно выделить отсутствие возможности передвижения и отсутствие каких-либо активностей за пределами главного офиса игрока.

Изображение выглядит как стена, в помещении, потолок, освещение

Автоматически созданное описание

Рисунок 2. Игра The Complex: Found Footage

The Complex: Found Footage – это игра-симулятор, где мир воспринимается через VHS-камеру. Игрок исследует огромный, неприветливый и загадочный мир, раскрывая тайну, которая скрыта внутри.

Несмотря на красивую графику и большую игровую территорию, игра не обладает какими-либо механиками кроме возможности свободного передвижения и использования фонарика.

Подводя итоги анализа, можно сказать, что имеющиеся аналоги либо не обладают интересным и продуманным геймплеем, либо устарели. Поэтому в разрабатываемой игре необходимо уделить особенное внимание самому игровому процессу, а именно придумать и продумать разнообразные игровые механики, чтобы игрок всё время чем-либо занимался и не успевал заскучать. Также стоит наполнить проект звуками, способствующими погружению в атмосферу игры.

1.3 Постановка задачи

Основной задачей проекта является создание игры со своей атмосферой страха и напряжения. Это может быть достигнуто с помощью разных механик игры, таких как наблюдение за опасными существами, затруднительные задачи, требующие внимания, ограниченные ресурсы и. В целом, задача состоит в том, чтобы создать мир, который заставит игрока сидеть на краю своего стула и держать дыхание, не зная, что произойдет дальше.

Также важно создать интересные механики для большего взаимодействия с окружающим игровым миром: система управления камерами, открытие и закрытие дверей в главной комнате, использование дрона для взаимодействия с отдалёнными объектами на игровой карте.

Стоит отметить и о необходимости в создании умного искусственного интеллекта, которым можно было бы заполнить игровое пространство и сделать прохождение игры ещё более интересным и напряжённым.

Функциональные требования к программе:

* высокая скорость работы программы;
* работа в трёхмерном пространстве;
* интересный и продуманный игровой процесс
* умный искусственный интеллект
* необычные механики

Минимальные системные требования для запуска игры:

- ОС: Windows® 7/Vista/XP и выше;

- Процессор: Intel® Core™ 2 Duo E6600 или AMD Phenom™ X3 8750;

- Оперативная память: 2 GB ОЗУ;

- Видеокарта: Видеокарта с 256 МБ памяти или больше, совместимая с DirectX 9 с поддержкой Pixel Shader 3.0;

- DirectX: Версии 9.0c;

- Место на диске: 4 GB.

1.4 Характеристика инструментальных средств разработки

Для разработки проекта был выбран язык программирования C# (произносится как "си шарп") - современный объектно-ориентированный и типобезопасный язык программирования. C# позволяет разработчикам создавать разные типы безопасных и надежных приложений, выполняющихся в .NET. C# относится к широко известному семейству языков C, и покажется хорошо знакомым любому, кто работал с C, C++, Java или JavaScript. Здесь представлен обзор основных компонентов языка C# 8 и более ранних версий. C# предоставляет языковые конструкции для непосредственной поддержки такой концепции работы. Благодаря этому C# подходит для создания и применения программных компонентов. С момента создания язык C# обогатился функциями для поддержки новых рабочих нагрузок и современными рекомендациями по разработке ПО. В основном C# - объектно-ориентированный язык.

Для написания программного кода будет использоваться интегрированная среда разработки Visual Studio которая используется для создания программного обеспечения для операционных систем Windows, Android, iOS и веб-приложений. Visual Studio предоставляет разработчикам возможности для написания кода, отладки приложений, создания пользовательских интерфейсов и управления проектами. В IDE также включены средства для тестирования, профилирования производительности, развертывания приложений и интеграции с другими инструментами разработки.

В создании трёхмерного пространства поможет межплатформенная среда разработки компьютерных игр Unity. Unity позволяет создавать приложения, работающие на более чем 25 различных платформах, включающих персональные компьютеры, игровые консоли, мобильные устройства, интернет-приложения и другие. 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.

С помощью программы MagicaVoxel сделаем воксельные модели и элементы игрового окружения для всей игры.

ПО Blender 3D — это бесплатный, открытый исходный код программного обеспечения для трехмерного моделирования, анимации и создания визуализаций. Blender был создан как инструмент для создания анимации, но в настоящее время он используется в различных отраслях, включая разработку игр, архитектурное проектирование, разработку визуальных эффектов для фильмов и другое. Blender содержит множество инструментов для создания моделей, включая возможность создания мешей (трехмерных форм), текстурирования, анимации и визуализации. Blender также имеет возможности для композитинга и редактирования видео, а также позволяет создавать интерактивные приложения разных типов.

Для создания и оформления пояснительной записки используем Microsoft Word (часто - MS Word, WinWord или просто Word) – программа обработки текстов, которая является частью пакета Microsoft Office и используется для создания, редактирования и форматирования документов любого типа, включая письма, резюме, научные статьи, отчеты и другие текстовые документы. В Microsoft Word есть множество инструментов форматирования, таких как использование различных шрифтов и размеров, стилей форматирования, таблиц и списков, поддержка изображений и таблиц, возможность создания гиперссылок и многое другое. Word также поддерживает режимы чтения и просмотра документов и позволяет работать в режиме совместной работы, если несколько пользователей работают над одним и тем же документом.

Для описания инструментов для построения диаграмм и схем задействуем бесплатное кроссплатформенное программное обеспечение для рисования графиков Diagrams.net (ранее известный как Draw.io) – это онлайн-приложение для создания различных типов диаграмм, включая блок-схемы, организационные схемы, макеты сетей, электрические схемы, диаграммы потоков и другие. Это бесплатное и открытое программное обеспечение, которое не требует установки на компьютер. Пользователи могут создавать диаграммы из набора предварительно созданных форм, а также импортировать изображения, текст и другие типы файлов. diagrams.net также поддерживает совместную работу над диаграммами с другими пользователями, добавление комментариев и многое другое.

# ГЛАВА 2 ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ ПРОГРАММЫ

## 2.1 Анализ требований и разработка спецификаций

Для разработки программного продукта, нужно определить спецификации и построить следующие диаграммы:

– диаграмма вариантов использования;

– диаграмма Ганта;

– структурная диаграмма;

– диаграмма переходов состояний;

– функциональная схема.

**Изображение выглядит как линия, диаграмма, График, Шрифт

Автоматически созданное описание**На рисунке 3 изображена диаграмма переходов и состояний.

Рисунок 3. Диаграмма переходов и состояний

Данная диаграмма наглядно показывает, как программа переходит из одного состояния в другое. Сначала игроку нужно смотреть по камерам за вражеским ИИ. После некоторого времени происходит встреча с искусственным интеллектом, а пользователю необходимо защитить себя от него. Далее пользователю необходимо следить за показаниями некоторых объектов, а также оставшимся запасом энергии.

На рисунке 4 показана диаграмма вариантов использования. На этой диаграмме отображены все возможные действия игрока. Он может: настраивать параметры игры, запускать игру и выходить из неё, взаимодействовать с камерами, открывать и закрывать двери, подсвечивать дверные проходы, свободно перемещаться по игровому миру с помощью дрона, взаимодействовать с различными предметами.

Изображение выглядит как текст, диаграмма, круг, снимок экрана

Автоматически созданное описание

Рисунок 4. Диаграмма вариантов использования

## 2.2. Проектирование программного обеспечения

Изображение выглядит как текст, Шрифт, снимок экрана, дизайн

Автоматически созданное описаниеНа рисунке 5 продемонстрирована диаграмма Ганта.

Рисунок 5. Диграмма Ганта

На диаграмме показано расписание, следуя которому можно эффективно распределить время и завершить проект в поставленные сроки.

На рисунке 6 изображена функциональная диаграмма.

Изображение выглядит как диаграмма, текст, линия, План

Автоматически созданное описание

Рисунок 6. Функциональная диаграмма

На диаграмме показано, как будет работать программа при взаимодействии с пользователем.

## 2.3 Разработка программного обеспечения

Разработка игры началась с проектирования пользовательского интерфейса.

На рисунке 7 представлено главное меню.

Изображение выглядит как текст, снимок экрана, лифт, дизайн

Автоматически созданное описание

Рисунок 7. Главное меню

В качестве основного шрифта был выбран VCR OSD Mono. Данный шрифт имеет «хорошую читабельность», а также за счёт своего необычного внешнего вида позволит игроку уже из меню начать погружаться в атмосферу игры. На фоне представлена локация из игры.

Представленный ниже код отвечает за переключение между сценами в игре. Переключение между главным меню и его подразделами обеспечивается UI-системой, которая скрывает ненужные элементы.

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.SceneManagement;

public class MainMenu : MonoBehaviour

{

public void LoadBackrooms()

{

SceneManager.LoadScene("SampleScene");

}

public void LoadMenu()

{

SceneManager.LoadScene("MainMenu");

}

public void QuitGame()

{

Application.Quit();

}

}

Рисунок 8 содержит в себе меню с настройками.

Изображение выглядит как снимок экрана, текст, линия, лифт

Автоматически созданное описание

Рисунок 8. Меню с настройками

В данном меню с помощью ползунка можно настроить громкость звука в игре. Также есть возможность изменить разрешение экрана, учтены три варианта: 1920х1080, 1600х900, 1280х1024. Снизу расположена кнопка для возвращения обратно в главное меню.

На рисунке 9 можно увидеть меню с повествованием.

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

Рисунок 9. Меню с повествованием

Данное меню содержит в себе краткое описание сюжета игры, чтобы игрок мог понять, что примерно его ожидает и, что от него может требоваться.

Рисунок 10 демонстрирует параметры меню в Unity.

Изображение выглядит как снимок экрана, текст, Мультимедийное программное обеспечение, программное обеспечение

Автоматически созданное описание

Рисунок 10. Настройки при создании меню

Чтобы меню и его объекты были адаптивными и подстраивались под различные экраны, в параметре UI Scale Mode ставим значение Scale With Screen Size. А в параметре Reference Resolution записываем значения для наиболее популярного разрешения Full HD (1920×1080).

Далее перейдём непосредственно к игровому процессу.

На рисунке 11 показан интерфейс во время игры.

Изображение выглядит как снимок экрана, Программное обеспечение для видеоигр, Компьютерная игра, 3D-моделирование

Автоматически созданное описание

Рисунок 11. Игровой интерфейс

Игровой интерфейс содержит в себе шкалу с запасом оставшейся энергии (если вся энергия закончится, то все системы безопасности отключатся и к игроку в комнату придут опасные сущности). Снизу по центру расположена кнопка для переключения в режим наблюдения через камеры. Также на экране расположены параметры, помогающие определить, какая сейчас ночь (всего их будет около 5) и сколько часов ещё осталось продержаться до финала.

Показанный ниже код отвечает за расход энергии на все элементы системы защиты. Структура программы позволяет менять определённые параметры, чтобы сделать игровой процесс сложнее и увеличить трату энергии.

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Battery : MonoBehaviour

{

public float energy = 100;

public float discharge = 0.2f;

public GameObject[] segments;

public TabController tablet;

public Door door1;

public Door door2;

public LightButton light1;

public LightButton light2;

private void Awake()

{

InvokeRepeating("Discharging", 1f, 1f);

}

private void Update()

{

SetDischarge();

ViewEnergy();

}

private void Discharging()

{

energy -= discharge;

}

private void ViewEnergy()

{

if (energy < 75)

segments[3].SetActive(false);

if (energy < 50)

segments[2].SetActive(false);

if (energy < 25)

segments[1].SetActive(false);

if (energy < 0)

segments[0].SetActive(false);

}

private void SetDischarge()

{

float tabletDC;

float doorsDC;

float lightDC;

//разрядка от планшета

if (tablet.minimap.activeSelf)

tabletDC = 0.1f;

else

tabletDC = 0f;

//разрядка от дверей

if (door1.isOpen == false && door2.isOpen == false)

doorsDC = 0.2f;

else if (door1.isOpen == false && door2.isOpen == true)

doorsDC = 0.1f;

else if (door1.isOpen == true && door2.isOpen == false)

doorsDC = 0.1f;

else

doorsDC = 0f;

//разрядка от света

if (light1.doorLight.activeSelf == true || light2.doorLight.activeSelf == true)

lightDC = 0.1f;

else

lightDC = 0f;

//суммарный разряд

discharge = 0.2f + tabletDC + doorsDC + lightDC;

}

}

На рисунке 12 изображена механика использования камер видеонаблюдения.

Изображение выглядит как снимок экрана, Мультимедийное программное обеспечение, программное обеспечение, Графическое программное обеспечение

Автоматически созданное описание

Рисунок 12. Система видеонаблюдения

На игровой территории расположены камеры видеонаблюдения, по которым можно отследить расположение вражеского искусственного интеллекта, а также узнать состояние генераторов.

На рисунке 13 можно увидеть использование дрона.

Изображение выглядит как снимок экрана, стена, в помещении, искусство

Автоматически созданное описание

Рисунок 13. Использование дрона

Дрон используется для передвижения по игровой карте и взаимодействия с объектами. Так игроку необходимо перезагружать два специальных генератора, каждый из которых отвечает за разные функции.

На этом изображении также можно заметить некое существо. Именно оно и является одним из двух вражеских ИИ, которые будут мешать игроку победить.

Логика одного из ИИ действует за счёт данного кода, расположенного ниже.

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.AI;

public class AiMovement : MonoBehaviour

{

public Transform target;

public NavMeshAgent agent;

public Animator anim;

public GameObject entityDoor;

private void Start()

{

agent.destination = target.position;

}

private void Update()

{

if (entityDoor.activeSelf == true)

{

agent.speed = 0;

anim.SetBool("IsMove", false);

}

else

{

agent.speed = 20;

if (agent.velocity.x != 0 | agent.velocity.z != 0)

{

anim.SetBool("IsMove", true);

}

else

{

anim.SetBool("IsMove", false);

}

}

}

}

## 

## 2.4 Отладка и тестирование программного обеспечения

Во время ручного тестирования не было обнаружено ошибок. Все функции работают исправно.

В таблице 2 представлены результаты отладки и тестирования приложения.

Таблица 2. Результаты тестирования и отладки приложения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № теста | Действие со стороны пользователя | Ожидаемая  реакция программы | Фактическая реакция программы | Ошибка выявлена |
| 1 | Использование дрона для прохождения сквозь элементы игрового окружения | Дрон упирается в игровые объекты | Благодаря реализации физики для всех объектов дрон не проходит сквозь другие объекты | Нет |
| 2 | Переход в режим наблюдения | Основной интерфейс выключается и заменяется на другой | Происходит замена интерфейса на нужный | Нет |
| 3 | Переход в режим дрона | Переключение между камерой игрока и камерой дрона | Переключение происходит, и игрок может управлять дроном | Нет |

## 2.5 Руководство по использованию программы

2.5.1 Руководство системного администратора

Для начала работы необходимо установить программу. Для этого был собран проект и создан исполняемый файл. На рисунках 14–16 продемонстрирован этот процесс.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 14. Выбор необходимых сцен и других параметров

Изображение выглядит как снимок экрана, текст, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 15. Создание и выбор папки, в которой будет храниться собранный проект и исполняемый файл

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Автоматически созданное описание

Рисунок 16. Результат

2.5.2 Руководство программиста

На рисунках 17–19 представлена структура проекта.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 17. Структура проекта

На рисунке 18 можно увидеть все панели и элементы, которые отвечают за переключение между разными меню.

Изображение выглядит как текст, Шрифт, снимок экрана, дизайн

Автоматически созданное описание

Рисунок 18. Содержание сцены «MainMenu»

На рисунке 19 показана структура содержания скриптов.

Изображение выглядит как снимок экрана, текст, пианино, Шрифт

Автоматически созданное описание

Рисунок 19.

Почти все скрипты в проекте расположены в определённых папках для того, чтобы в дальнейшем было проще ориентироваться среди них

2.5.3 Руководство оператора

Для начала работы с приложением его необходимо скачать (если не скачано) и запустить .exe файл.

Изображение выглядит как текст, Шрифт, снимок экрана, алгебра

Автоматически созданное описание

Рисунок 20. Системные требования

После запуска приложения пользователя встретит главное меню. Сначала нужно настроить все необходимые параметры (уровень звука и разрешение экрана). После можно зайти в повествовательный раздел и ознакомиться с историей игры. Далее запускаем игру, в начале которой игрока встретит аудио сообщение, обучающее игрока основам игры.

# ЗАКЛЮЧЕНИЕ

Целью проекта является создание 3D игры «Secret things» в жанре «Adventure» с элементами Survival horror.

Для достижения этой цели шаг за шагом, были выполнены следующие задачи: разработка игры на языке С#; разработка и создание дизайна интерфейса; реализация интерфейса приложения; тестирование и отладка игры.

Пригодились знания языка C#, полученные во время учебного процесса.Для построения диаграмм и схем была изучена среда разработки draw.io (diagrams.net). Был разработан график проведения работ, который был реализован с помощью диаграммы Ганта.

Также была изучена платформа для разработки игр и приложений Unity. Использование данного движка значительно облегчило процесс разработки за счёт своего хорошего игрового редактора и инструментария.

По итогу разработки приложения отметим, что:

- было разработано приложение, ориентированное на целевую аудиторию в возрасте 16–25 лет;

- был реализован удобный и интуитивно понятный интерфейс;

- были разработаны различные механики делающие геймплей занимательным и в меру трудным и страшным.

Необходимо отметить, что данное приложение имеет высокий потенциал для его развития.

В будущем планируется добавить больше игровых активностей, чтобы сделать игровой процесс интереснее. Интересным будет добавление дополнительных ночей для выживания, каждая из которых будет отличаться от предыдущей значительно увечившимся уровнем сложности. Этого можно достигнуть с помощью повышения расхода энергии, большим количеством игровых активностей и дополнительным искусственным интеллектом.

Подводя итог всему вышесказанному, можно сделать вывод о том, что главная цель и все поставленные задачи ВКР были успешно достигнуты.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

*Законодательные и нормативные акты:*

1. ГОСТ Р 7.0.12-2011 Библиографическая запись. Сокращение слов и словосочетаний на русском языке. Общие требования и правила. – М.: Стандартинформ, 2012. – 61 с.
2. ГОСТ 7.1-2003 Библиографическая запись. Библиографическое описание. Общие требования и правила составления. – М.: Стандартинформ, 2010. – 92 с.
3. ГОСТ 7.32-2017 Отчет о научно-исследовательской работе. Структура и правила оформления. – М.: Стандартинформ, 2017. – 47 с.
4. ГОСТ 7.82-2001 Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила составления. – М.: ИПК Издательство стандартов, 2001. – 39 с.
5. ГОСТ Р 7.0.100-2018 Библиографическая запись. Библиографическое описание. Общие требования и правила составления. – М.: Стандартинформ, 2018. – 122 с.
6. ГОСТ Р 7.0.5-2008 Библиографическая ссылка. Общие требования и правила составления. – М.: Стандартинформ, 2008. – 32 с.
7. Единая система программной документации. – М.: Стандартинформ, 2005. – 128 с.

*Учебная и научная литература:*

1. Павловская, Т.А. C#. Программирование на языке высокого уровня: учебник для студентов вузов. – СПб: Питер, 2020. – 432 с.

*Интернет-документы*

1. Алгоритмы генерации игр. – [Электронный ресурс]. – URL: <https://ru.wikipedia.org/wiki/Алгоритм_Эллера#:~:text=Алгоритм%20Эллера%20—%20математический%20генератор,лабиринты%20с%20неограниченным%20количеством%20рядов> (дата обращения: 25.09.2022)
2. Алгоритмы генерации игр. – [Электронный ресурс]. – URL: <https://tproger.ru/articles/maze-generators/> (дата обращения: 25.09.2022)
3. Добро пожаловать в интегрированную среду разработки Visual Studio. – [Электронный ресурс]. – URL: <https://learn.microsoft.com/ru-ru/visualstudio/get-started/visual-studio-ide?view=vs-2019> (дата обращения: 25.09.2022)
4. Как создавать в играх бесконечные миры при помощи процедурной генерации. – [Электронный ресурс]. – URL: <https://habr.com/ru/company/piter/blog/593237/> (дата обращения: 25.09.2022)
5. Краткий обзор языка C#. – [Электронный ресурс]. – URL: <https://learn.microsoft.com/ru-ru/dotnet/csharp/tour-of-csharp/> (дата обращения: 25.09.2022)
6. Обзор систем процедурной генерации игр. – [Электронный ресурс]. – URL: <https://cyberleninka.ru/article/n/obzor-sistem-protsedurnoy-generatsii-igr/viewer> (дата обращения: 25.09.2022)
7. Microsoft Word. – [Электронный ресурс]. – URL: <https://microsoft.fandom.com/ru/wiki/Microsoft_Word> (дата обращения: 25.09.2022)
8. Unity (игровой движок). – [Электронный ресурс]. – URL: <https://ru.wikipedia.org/wiki/Unity_(игровой_движок)> (дата обращения: 25.09.2022)