

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет
вычислительной техники

Кафедра
«Вычислительная техника»

Направление подготовки: 09.03.01 «Информатика и вычислительная техника»
Профиль подготовки: «Программное обеспечение средств вычислительной техники и автоматизированных систем»

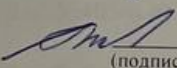
БАКАЛАВРСКАЯ РАБОТА

на тему

ПРИЛОЖЕНИЕ В ЖАНРЕ ИЗОМЕТРИЧЕСКАЯ
ГОЛОВОЛОМКА. ГРАФИЧЕСКАЯ СОСТАВЛЯЮЩАЯ

Студент  Литвинов Андрей Юрьевич
(подпись, дата) 20.06.2024 (ФИО полностью)

Руководитель  Юрова О.В.
(подпись, дата) 20.06.2024 (фамилия, инициалы)

Нормоконтролёр  Бычков А. С.
(подпись, дата) 20.06.2024 (фамилия, инициалы)

Работа допущена к защите (протокол зас. кафедры от 27.05.2024 г. № 11)

Заведующий кафедрой  М.А. Митрохин
(подпись) (фамилия, инициалы)

Работа защищена с отметкой хорошо (протокол заседания ГЭК от
24.05.2024 № 36)

Секретарь ГЭК  Е.И. Гурин
(подпись) (фамилия, инициалы)

Пенза 2024

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет
вычислительной техники

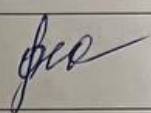
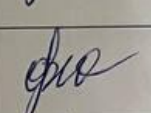
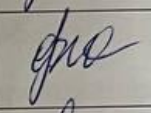
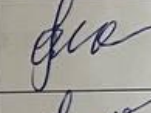
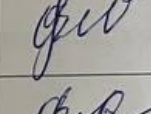
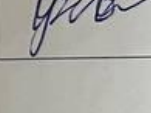
Кафедра
«Вычислительная техника»

«Утверждаю»
Заведующий кафедрой
М.А. Митрохин
« 19 » 04 2024 г.

**ЗАДАНИЕ
НА БАКАЛАВРСКУЮ РАБОТУ**

1. Студент Литвинов Андрей Юрьевич гр. 20ВВП1 факультета вычислительная техника
2. Тема работы «Приложение в жанре изометрическая головоломка. Графическая составляющая»
Тема утверждена приказом ПГУ №405/о от "19" апреля 2024г.
3. Руководитель бакалаврской работы к.т.н. доцент кафедры ВТ ПГУ Юрова О.В.
4. Задание на работу (назначение разработки, исходные данные и т.п.)
Назначение разработки – Приложение в жанре изометрическая головоломка.
Разработать сюжетное игровое приложение с головоломками разного типа. Подготовить 3D модели (моделирование, текстурирование, оптимизация) для создания уровней. Язык программирования C#, среда разработки Visual Studio Code, игровой движок Unity, программное обеспечение для создания 3D моделей Blender. Программа должна обладать интуитивно понятным интерфейсом и эстетичной графикой в определенной стилистике.
5. Перечень подлежащих разработке вопросов
Требуется произвести анализ предметной области, определить архитектуру приложения, выбрать инструменты для разработки, реализовать приложение.

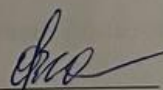
6. Календарный график выполнения работы

№ п/п	Наименование этапов работы	Объем работы	Срок выполнения	Подпись руководителя
1.	Анализ предметной области	10 %	19.04.24 - 25.04.24	
2.	Выбор архитектуры системы и технологий, применяемых при проектировании и разработке	15 %	26.04.24 - 02.05.24	
3.	Разработка сценария, логики головоломов	20 %	03.05.24 - 07.05.24	
4.	Разработка уровней, моделей, текстур и интерфейса	25 %	08.05.24 - 19.05.24	
5.	Тестирование и отладка	15 %	20.05.24 - 22.05.24	
6.	Выполнение задания по БЖД	5 %	23.05.24 - 25.05.24	
7.	Оформление выпускной работы	10 %	25.05.23 - 31.05.23	

Дата выдачи задания "19" апреля 2024г.

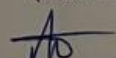
Руководитель бакалаврской работы _____

(подпись, дата)

 Юрова О.В.

Задание к исполнению принял студент _____

(подпись, дата)

 Литвинов А.Ю.

Работу к защите допустить
Декан факультета _____

(подпись, дата)

21.06.2024

 Филова Л.Р.

Реферат

Бакалаврская работа содержит 75 лист, 32 рисунка, 6 источников, 3 приложения

ГОЛОВОЛОМКА, МОДЕЛИРОВАНИЕ В BLENDER 3D, СОЗДАНИЕ ГРАФИКИ, ГЕЙМДИЗАЙН, ИЗОМЕТРИЧЕСКАЯ КАМЕРА, UNITY, ТЕКСТУРИРОВАНИЕ

Цель работы – разработать приложение в жанре изометрическая головоломка. Основным объектом разработки является графическая составляющая:

- эстетически приятный и понятный для пользователя интерфейс;
- следование одному визуальному стилю во всех аспектах приложения;
- создание 3D моделей и построение уровней для разрабатываемого приложения.

Результатом работы является разработанное игровое приложение, которое полностью удовлетворяет условиям поставленных задач, то есть:

- имеет индивидуальный визуальный стиль;
- проработанные уровни в трехмерном пространстве;
- минималистичный, но понятный и удобный пользовательский интерфейс.

					ПГУ1.090301.14.001 ПЗ				
Изм.	Лист	№ докум.	Подпись	Дата	Приложение в жанре изометрическая головоломка. Графическая составляющая Пояснительная записка	Лит.	Лист	Листов	
Разраб.		Литвинов А.Ю.		20.06.24					
Провер.		Юрова О.В.		20.06.24			2	75	
Н. Контр		Бычков А.С.		20.06.24		ФВТ, каф. ВТ гр. 20ВВП1			
Утверд.		Митрохин М.А.		20.06.24					

Оглавление

Введение.....	5
1 Приложение в жанре изометрическая головоломка.....	7
1.1 Анализ предметной области	7
1.2 Анализ аналогов	11
1.3 Постановка задачи	16
1.4 Сценарий использования.....	17
1.4 Выбор инструментов	26
1.4.1 Выбор программного обеспечения для моделирования.....	26
1.4.2 Выбор программного обеспечения для редактирования изображений	29
1.4.3 Выбор среды разработки игрового приложения	30
1.4.4 Выбор языка программирования и среды редактирования кода	31
1.4.5 Выбор программы для создания музыкального сопровождения игры	33
1.5 Архитектура графической составляющей приложения.....	34
1.5.1 Сцены уровней	34
1.5.2 Объекты	35
1.5.3 Текстуры	35
1.5.4 Графический интерфейс (UI)	35
1.6 Реализация	36
1.6.1 Исследование и сбор референсов.....	36
1.6.2 Разработка основных механик.....	37
1.6.3 Проработка сюжетной линии	37
1.6.4 Стилистика и визуальное оформление	37
1.6.5 Планирование уровней	38
1.6.6 Создание музыкального сопровождения.....	45

1.6.7 Реализация меню настроек приложения	47
1.6.8 Тестирование приложения	48
2 Охрана труда.....	50
2.1 Организация рабочего места при работе на персональном компьютере	50
2.1.1 Освещение	50
2.1.2 Общие эргономические требования к рабочему месту оператора ...	51
2.1.3 Климатические условия	53
2.2 Психофизиологические аспекты труда программиста	53
2.3 Электробезопасность	53
2.3.1 Использование исправного оборудования	54
2.3.2 Заземление и защитные устройства	54
2.4 Пожарная безопасность	54
Заключение.....	55

Введение

В последние несколько десятилетий игровая индустрия пережила бурное развитие, предлагая игрокам все более разнообразные и инновационные формы развлечений. Игры становятся не только способом досуга, но и важным элементом культуры, средством обучения и инструментом для развития когнитивных способностей. В этом контексте особое место занимают игры жанра головоломка, которые благодаря своей структуре и механикам стимулируют логическое мышление, развивают поиск решения проблем и творческий подход.

Головоломки как жанр имеют долгую историю, уходящую корнями в настольные игры и задачи, требующие интеллектуальных усилий. С переходом в цифровую форму головоломки приобрели новую глубину и сложность, становясь доступными широкому кругу пользователей благодаря развитию технологий. Видеоигры данного жанра отличаются разнообразием форм и стилей: от классических пазлов и кроссвордов, до сложных стратегических задач и физических симуляций.

Создание игры в жанре головоломка требует тщательной проработки всех аспектов: от концепции и геймдизайна, до реализации и тестирования. Важно учитывать интересы и ожидания целевой аудитории, предлагая не только увлекательные и сложные задачи, но удобный интерфейс, а также визуально привлекательное оформление. Особое внимание следует уделить балансу между сложностью и доступностью игры, чтобы она была интересна как новичкам, так и опытным игрокам.

Настоящий дипломный проект посвящен разработке игры в жанре головоломка. В ходе работы будут рассмотрены такие ключевые этапы создания игры, как разработка графики и геймдизайн уровней, выбор инструментов и технологий для реализации, а также тестирование и оптимизация продукта. Целью проекта является создание красивой,

увлекательной и качественной игры, способной привлечь внимание игроков и занять достойное место на рынке видеоигр.

1 Приложение в жанре изометрическая головоломка

1.1 Анализ предметной области

Головоломки – это увлекательные интеллектуальные игры, требующие решения сложных задач или нахождения скрытых закономерностей. Их разнообразие поражает воображение: от классических кроссвордов и sudoku до механических пазлов и логических загадок. Головоломки привлекают внимание людей различных возрастов и интересов, ведь они предлагают уникальное сочетание развлечения и умственного вызова.

Интерес к головоломкам объясняется их способностью стимулировать когнитивные процессы. Решение головоломок требует концентрации, логического мышления, внимания к деталям и креативности. Эти игры часто предполагают нестандартный подход и изобретательность, что делает процесс решения захватывающим и удовлетворяющим.

Кроме развлечения, головоломки приносят значительную пользу для развития различных навыков и умений. Регулярное занятие головоломками улучшает память, повышает способность к анализу и синтезу информации, а также способствует развитию пространственного мышления. В образовательной среде головоломки используются для развития критического мышления и способностей к решению проблем у студентов.

Таким образом, головоломки представляют собой не только увлекательное времяпрепровождение, но и мощный инструмент для интеллектуального развития.

Изометрические головоломки, особенно в жанре escape game, занимают особое место в мире интеллектуальных игр. Эти игры предлагают уникальный опыт, сочетающий разнообразие головоломок и увлекательное повествование. Изометрическая перспектива, при которой игрок наблюдает за игровым

пространством под углом, придает играм глубину и позволяет детально исследовать окружающий мир.

Одним из ключевых преимуществ изометрических головоломок является их способность развивать широкий спектр навыков. Игрокам предстоит решать головоломки различных типов: логические задачи, головоломки на пространственное мышление, задачи на внимательность и креативное мышление. Например, для открытия запертой двери может потребоваться расшифровать код, собрать сложный механизм или найти спрятанные предметы. Такой подход способствует всестороннему развитию когнитивных способностей.

Еще одним значительным плюсом является наличие увлекательного сюжета. В играх жанра escape game игроки часто оказываются в таинственных и необычных ситуациях, из которых нужно найти выход. История, разворачивающаяся по мере прохождения игры, удерживает внимание и побуждает к дальнейшему исследованию. Интересный сюжет не только повышает мотивацию к решению головоломок, но и делает игровой процесс более осмысленным и запоминающимся.

Сочетание разнообразных головоломок и захватывающей истории делает изометрические escape games популярными среди широкого круга игроков. Они не только развлекают, но и оказывают позитивное влияние на умственные способности, предлагая уникальный и увлекательный способ провести время с пользой.

На сегодняшний день графическая составляющая головоломок является неотъемлемой частью общей разработки. Это важно, как с эстетической стороны, чтобы конечному пользователю нравилось взаимодействовать с интерфейсом, так и со стороны функциональной. Графическая составляющая помогает раскрыть каждую функцию приложения и использовать на полную его потенциал.

Изометрическая головоломка может быть представлена в нескольких визуальных форматах:

простая двумерная графика с мнимым эффектом глубины: достигается с помощью правильного изображения перспективы каждого объекта;

3D визуализация сцен и предметов: в данном случае объекты будут иметь более реалистичный вид и добавят большего погружения в использовании.

Оба этих направления имеют свои недостатки и достоинства.

К существенным плюсам двумерного представления относятся:

- более простое и доступное создание графики: нет особой необходимости в специализированных программах;

- меньше затрат ресурсов, что является большим плюсом в разработке приложений для мобильных платформ;

- особенность создания такой графики сокращает время на представление общей сцены, так как художнику не обязательно делать дополнительные манипуляции с перемещением камеры и расстановкой объектов на сцене.

К минусам же можно отнести:

обязательное наличие базовых знаний в рисунке или образование художника;

ограниченность инструментов создания такой графики;

меньшая интерактивность, которая может не заинтересовать конечного пользователя.

Теперь рассмотрим достоинства трехмерного представления:

вариативность представления сцен и объектов при построении – можно создавать различные формы, использовать разные виды перспективы, что добавляет глубины и интереса к головоломке;

с точки зрения дизайна такой вид графики дает большой простор для фантазии, так как возможно реализовать более сложные и вариативные уровни;

работа с 3d графикой не требует навыков художника и знаний в построении линий перспективы, потому что она реализуется на программном уровне;

создание объектов имеет больше уровней разработки, что дает четкий контроль над каждым мелочами этого процесса.

Однако, данный вид представления также имеет ряд недостатков:

- больше затраченных ресурсов, из этого следует большая нагруженность, особенно на мобильные устройства;
- необходимость знаний в специализированных программах разработки трехмерной графики;
- увеличенные затраты памяти устройства из-за большого объема данных, выраженных в виде моделей и текстур.

В современных реалиях многие недостатки трехмерной графики нивелируются быстрым развитием технологий. Многие стационарные компьютеры и ноутбуки сейчас имеют большие мощности, а мобильные устройства часто снабжаются объемом данных с запасом, не говоря уже о способах расширения с добавлением физических и облачных хранилищ данных.

Также современные среды разработки и специализированное программное обеспечение постоянно развиваются. Это дает простор для создания больших и проработанных миров и историй и, конечно, для интересных и затягивающих головоломок. Возможен вариант представления не только типичной головоломки, которая включает в себя определенную задачу для достижения цели, но и передачу истории мира, созданного для данного приложения.

Трехмерная графика на данный момент используется во многих сферах деятельности. Это и развлечения: игры, мультфильмы, кинематограф; и сфера дизайна: мебель, готовые интерьеры, дизайн планы помещений; а также производство: визуализация компонентов технических устройств, моделирование деталей для 3D печати.

В реализации приложения в жанре изометрическая головоломка можно использовать один из нескольких визуальных стилей этого направления.

Стиль реализм – включает в себя отражение в графике объектов, приближенных визуально к вещам в реальном мире. Это достигается за счет точных пропорций и объемов, а также использованием текстур, созданных из

фотографий реальных материалов. При создании объектов в таком стиле можно прибегнуть к методу фотограмметрии. В таком случае с помощью специального программного обеспечения, реальный объект фотографируется с множества сторон, и на основе этих фотографий программа создает модель с уже готовыми сгенерированными текстурами цветов.

Стиль low poly представляет собой визуально упрощенный вариант имеющегося объекта, являющимся ориентиром. Ему присущи простые палитры цветов, ровные линии и невысокая детализация модели. Данный стиль отлично подходит для визуализации головоломки, так как не будет отвлекать пользователя от решения поставленной задачи своей излишней перегруженностью деталей, как цветовой, так и объектной. Данный стиль был выбран исходя из жанра нашего приложения, а также по причине лучшей оптимизации на любых устройствах, потому как текстуры и объекты легче отрисовывать, они занимают меньше места на устройствах хранения данных.

Еще одним не менее важным моментом является звуковое сопровождение внутри приложения. Оно дает более глубокое погружение в процесс. Это передается не только через ненавязчивое музыкальное сопровождение, но и с помощью использования звуков, которые должны издавать объекты.

Данная область будет рассматриваться в этой работе только на базовом уровне, потому что основой является именно графическая часть.

1.2 Анализ аналогов

Головоломки – это одни из старейших развлечений в истории человечества. Они стали неотъемлемой частью культур многих стран и народов. И появление все новых и новых видов и воплощений данного занимательного и развивающего вида досуга не является чем-то необычным.

Для более полной картины исследования предметной области введем несколько аналогичных приложений этого жанра:

– tiny room stories: town mystery;

- the almost gone;
- nox escape game.

Первое представлено на рисунке 1.1.

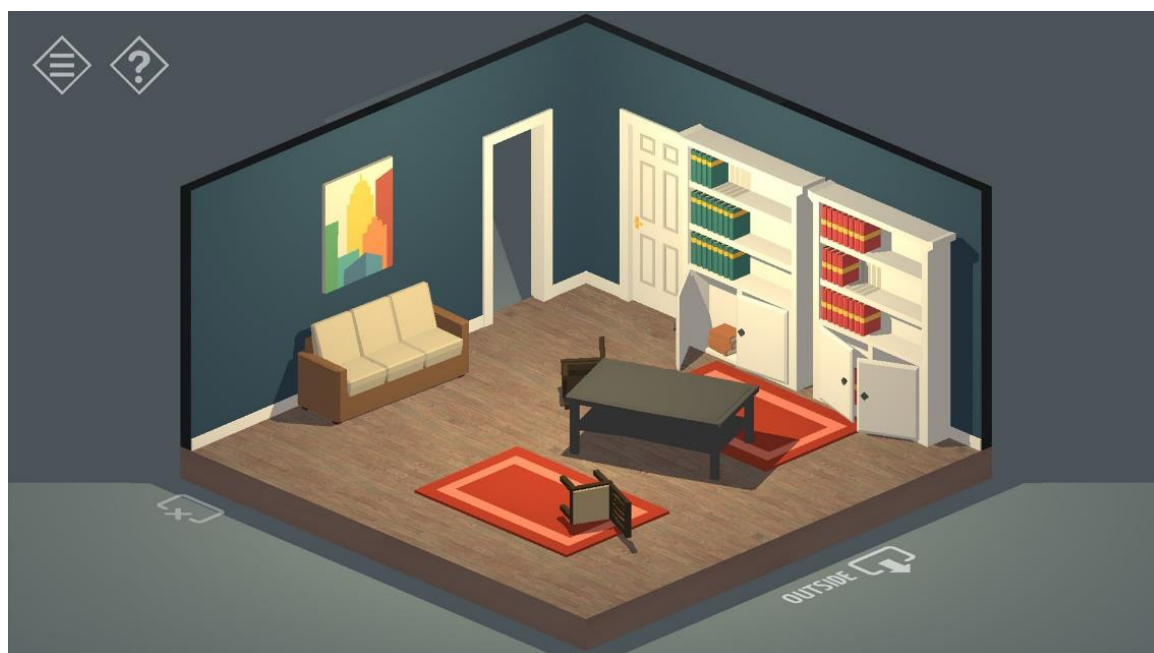


Рисунок 1.1 – Визуальная часть «Tiny Room Stories: Town Mystery»

Изометрическая головоломка с сюжетом в жанре детектив, где игрок исследует различные маленькие комнаты и пространства, решая головоломки и раскрывая загадки.

Ее достоинствами является:

- следование одному визуальному стилю на протяжении всей игры;
- правильная передача атмосферы через визуализацию объектов и локаций, присущая истории в жанре детектив;
- многие визуальные решения работают на улучшение нарратива, и как следствие большого погружения в процесс продвижения по сюжету.

Головоломка имеет следующие недостатки:

- визуальное решение разворота локации с постепенным исчезновением объектов стен перегружает общую картину происходящего;

– слишком быстрое движение объектов в кадре в скором времени влияет негативно на восприятие пользователя – а именно – возникновение головокружения. Данная проблема актуальна для людей со слабым вестибулярным аппаратом;

– постоянное появление на экране главного героя, который комментирует каждое действие пользователя, когда тот взаимодействует с не имеющими функциональной значимости объектами;

– размещение большого количества объектов, с которыми можно взаимодействовать, при этом никакого значения для решения задач они не имеют: излишняя интерактивность создает ощущение перегруженности.

Еще одна изометрическая головоломка с драматическим сюжетом и необычным визуальным стилем показана на рисунке 1.2.

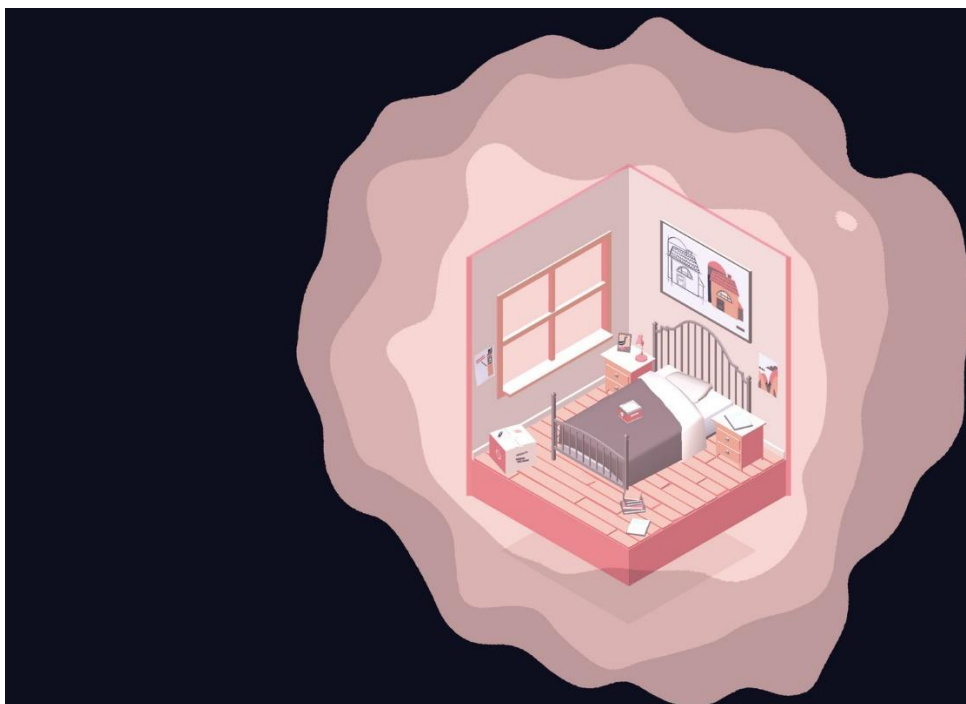


Рисунок 1.2 - Визуальная часть «The Almost Gone»

Достоинствами данного приложения можно назвать:

– графическая часть выполнена в сказочном и стиле мультфильмов, что вкупе с драматической историей дает необыкновенный опыт;

- вариативность локаций по ходу прохождения;
- отличная работа с освещением и эффектами, которые добавляют атмосферы общей картине происходящего.

К недостаткам, однако, можно отнести:

- визуальная скованность локации за счет их небольших размеров, создается впечатление ограниченности и появляется необходимость непрерывно вращать поле взаимодействия;
- малое количество взаимодействия с камерой, так как приближение объектов реализовано с помощью появления дополнительных окон вида, из-за чего теряется погружение;
- легкость решения головоломок.

Последним примером приложения в жанре изометрическая головоломка стала Nox escape game, которая представлена на рисунке 1.3.



Рисунок 1.3 - Визуальная часть «NOX escape game»

В ней предлагается исследовать различные сцены и решать головоломки, чтобы найти выход. Стандартная задача для данного жанра.

Из плюсов организации графической части можно назвать следующее:

- приятное и понятное перемещение камеры с вращением сцены на экране;
- освещение на уровнях работает на улучшение атмосферы и глубины 3D окружения и объектов;

Существенным недостатком является смешение визуальных стилей, а именно:

- при общем представлении объектов в стиле low poly редко появляются модели в воксельном стиле. Это мешает погружению и добавляет ощущение неуместности их наличия на сцене. Пользователь видит несоответствие графического представления общей картины с отдельными элементами;

- аналогичная ошибка была допущена на стадии текстурирования моделей. При общей мягкой палитре цветов и плавности и ровности линий на некоторых объектах нанесены декали в пиксельном стиле, что не объяснено логически на уровне нарратива.

Анализ существующих аналогов приложения показывает, что в зависимости от сюжетной составляющей и направления, важно подобрать соответствующий визуальный стиль, который не будет конфликтовать и, как следствие, портить атмосферу и погружение в происходящее. На стадии создания 3D моделей нужно иметь концепт конечного варианта. Соблюдение всех правил создания и текстурирования объектов, а также правильное расположение их на сцене поможет вовлечению пользователя и будет сохранять его интерес по ходу прохождения всех загадок и головоломок.

Конечно, еще одним важным моментом при создании является правильное расположение камеры, ее приближение к объектам взаимодействия, плавность и перемещение при смене сцены.

1.3 Постановка задачи

При разработке приложения головоломки графическая часть имеет важную роль в процессе его использования, поэтому необходимо обеспечить приятный пользовательский опыт.

При разработке приложения необходимо сделать:

– создание окружения:

- 1) создание трехмерной изометрической среды, включая фоны и объекты;
- 2) разработка текстур для окружения;

– дизайн объектов:

- 1) разработка трехмерных моделей предметов и других объектов;
- 2) создание текстур и анимаций для объектов.

– разработка пользовательского интерфейса, включая меню, кнопки, иконки и т. д.

– оптимизация и тестирование:

- 1) оптимизация графики для повышения производительности;
- 2) корректировка и исправление ошибок.

– дополнительной задачей будет создание звукового сопровождения и звуковых эффектов.

В игре будут размещены несколько типов моделей:

– статичные объекты окружения:

- 1) объекты, которые будут располагаться на сцене независимо от поворота камеры (пол, элементы мебели, предметы, не закрывающие обзор);
- 2) объекты, которые под определенным углом будут скрываться для предоставления видимости необходимых объектов (стены, большие элементы мебели, препятствующие видимости).

– объекты, с которыми можно взаимодействовать:

- 1) объекты, которые можно передвигать по заданной траектории;

1.4 Сценарий использования

Стартовая сцена игры будет представлять общую картину первого уровня с надписью предложения нажать на любую часть экрана. После этого действия камера переместится на первую открывающую головоломку в виде 3D паззла. Данный паззл познакомит игрока с одной из базовых механик игры, а именно сбор предметов в инвентарь и дальнейшее их использование для решения какой-либо задачи.

Если конкретизировать первую задачу, нужно упомянуть о сюжете игры. Мы реализуем историю о сыне и отце, где конечный пользователь будет отыгрывать роль ребенка, пытающегося разобраться в происшествии, которое случается после решения первой головоломки, и ищущего своего отца. Первый паззл является подарком от отца и выполнен он в виде своеобразного конструктора автомобиля. Он показан на рисунке 1.4.

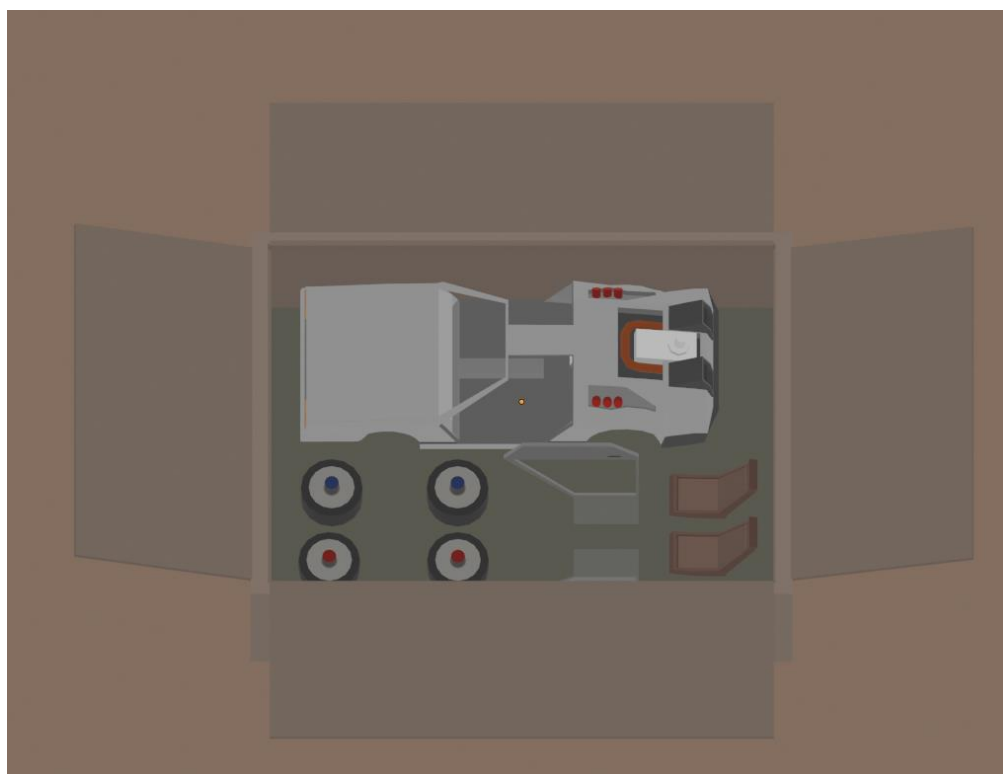


Рисунок 1.4 – Головоломка – конструктор автомобиля

Игроку необходимо взять все предметы из коробки они добавятся в инвентарь, и после этого собрать всю конструкцию воедино. Когда эти действия будут совершены, запустится скрипт происшествия, о котором было написано ранее. На экране появится текст, а весь уровень сильно потемнеет, что укажет на отключение света в доме – это видно на рисунке 1.5.



Рисунок 1.5 – Комната без света

Игроку с помощью подсказки будет предложено повернуть всю комнату, что познакомит его с еще одной основной механикой игры. Поворот комнаты в дальнейшем будет служить для поиска предметов и решения головоломок. На данном этапе прохождение является линейным, все головоломки заблокированы, кроме одной. Для продвижения к ней необходимо использовать механику перемещения между уровнями, которая будет реализована через нажатия на двери, находящиеся на одной из сторон комнаты, это видно на рисунке 1.6.

Изм.	Лист	№ докум.	Подпись	Дата



Рисунок 1.6 – Вторая сторона комнаты с дверью

Игрок будет перемещаться по комнатам без ограничений, но, чтобы открыть доступ ко всем головоломкам, необходимо включить свет в гараже. Реализовано это через переключение рубильников, что является еще одной разновидностью взаимодействия с объектами на сцене. На этом этапе уже есть небольшая нелинейность, так как на уровнях в доме есть несколько объектов и головоломок, с которыми можно взаимодействовать. Для примера: пользователь может исследовать уровень «Гараж», где был включен свет.

Во-первых, на щитке с рубильниками, который показан на рисунке 1.7, видно отсутствие одного из них, во-вторых, при правильном ракурсе камеры на уровень будет замечен объект «лопата», который можно поднять и поместить в инвентарь. Оба этих случая ведут к решениям разных головоломок.

Первый служит причиной для поиска недостающей детали и приводит к еще одной головоломке.

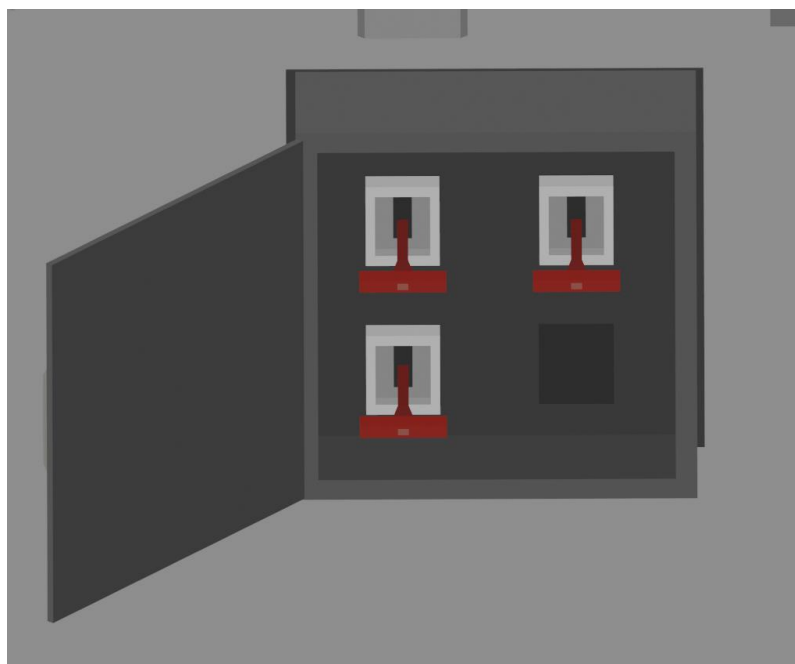


Рисунок 1.7 – Считок для включения электричества

Второй же служит для отвлечения фокуса внимания игрока, так как реализовать предмет «лопата» из инвентаря можно только после включения света на уровне «Задний двор». На рисунке 1.8 видно изменение освещения.



Рисунок 1.8 – Изменения освещения

Нелинейная часть следует и далее. Игрок может начать решение головоломки с недостающим выключателем света, для доступа к головоломке на уровне «Улица». Для этого необходимо вернуться на уровень «Детская» и достать большую коробку, которая стоит на нижней полке шкафа. Но для того, чтобы ее открыть также нужно найти ножницы. Для их нахождения игрок должен будет повернуть комнату под нужным углом и обратить внимание под кровать. Ножницы представлены на рисунке 1.9.



Рисунок 1.9 – Предмет «ножницы»

Далее взять их и применить к коробке. После этого действия откроется доступ к следующей головоломке, где будет необходимо по очереди в порядке возрастания вытащить все детали, ориентируясь на количество мест скрепления.

Она показана на рисунке 1.10.



Рисунок 1.10 – Головоломка конструктор

Из всех этих деталей будет собран недостающий рубильник, который можно будет применить на свободное место в щитке на уровне «Гараж». Это действие позволит разблокировать головоломку на уровне «Задний двор».

Область для взаимодействия с предметом показана на рисунке 1.11.

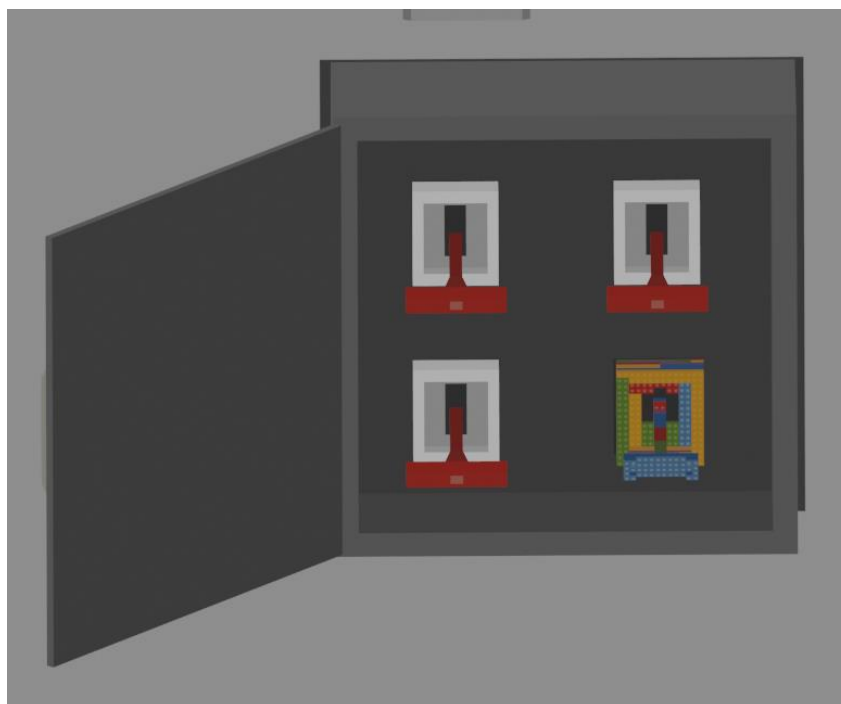


Рисунок 1.11 – Добавление недостающего элемента

Для доступа к «Амбару» пользователь должен решить головоломку на уровне «Задний двор», который показан на рисунке 1.13.



Рисунок 1.13 – Уровень «задний двор»

Так после решения задачи со светом, вращая уличный уровень, игрок должен найти камень рядом с дверью в «Амбар», показанный на рисунке 1.14.



Рисунок 1.14 – Приближенный вариант уровня

Изм.	Лист	№ докум.	Подпись	Дата

На нем имеется оранжевый крест для обозначения загадки. Цвет был выбран не случайно, он точно совпадает с цветом рукоятки «лопаты», которая представлена на рисунке 1.15, и к этому моменту уже должна храниться в инвентаре игрока.



Рисунок 1.15 – Предмет «лопата» на уровне «гараж»

Применив этот предмет к «камню», пользователь откроет доступ к «ключу» от «амбара». Это видно на рисунке 1.16.



Рисунок 1.16 – Решение головоломки с камнем

Последним доступным уровнем является «Амбар», который показан на рисунке 1.17. Единственный объект, с которым можно взаимодействовать, это верстак. Он сдвигается в сторону для того, чтобы открыть доступ к двери на уровень «Лаборатория».

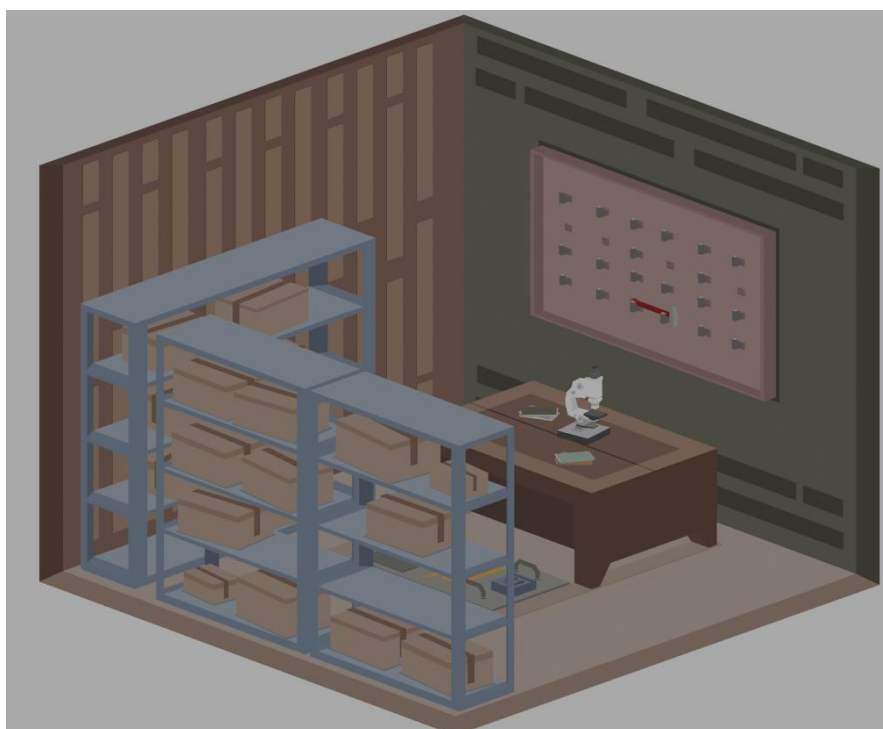


Рисунок 1.17 – Последний уровень

Изм.	Лист	№ докум.	Подпись	Дата

ПГУ1.090301.14.001 ПЗ

Лист

25

На этом моменте демоверсия игры заканчивается.

1.4 Выбор инструментов

1.4.1 Выбор программного обеспечения для моделирования

Выбор подходящего программного обеспечения для создания 3D моделей имеет огромное значение для художников и разработчиков. Для тех, кто сосредоточен на создании low poly моделей, важно учитывать функциональные возможности, стоимость, кривую обучения и интеграцию с другими инструментами. В этом обзоре будут рассмотрены четыре популярных программы: Blender 3D, 3ds Max, Maya и ZBrush. Особое внимание уделено тому, что Blender 3D выбран как предпочтительное решение для создания low poly моделей. Основной интерфейс данного ПО показан на рисунке 1.18.

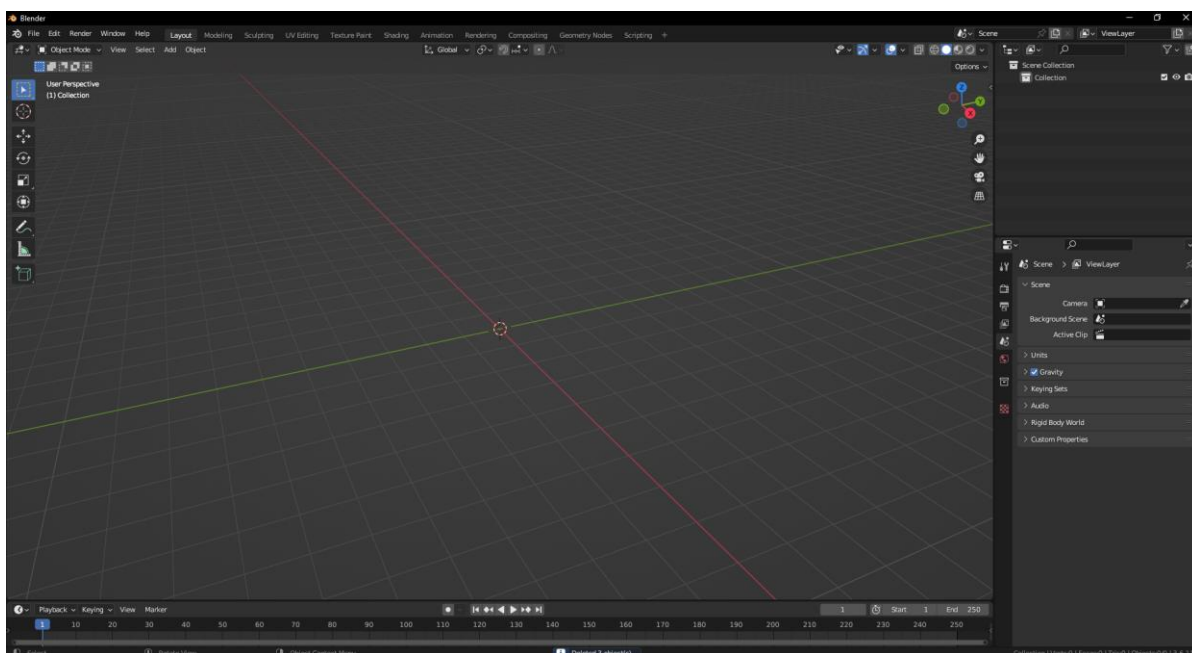


Рисунок 1.18 – Основной интерфейс Blender 3D

Blender 3D - бесплатное и открытое программное обеспечение, которое идеально подходит для создания low poly моделей благодаря своей гибкости и широкому набору инструментов.

Преимущества Blender 3D:

– бесплатность и открытость. Полностью бесплатное ПО с открытым исходным кодом, что делает его доступным для всех категорий пользователей;

– мощный набор инструментов для моделирования. Инструменты для полигонального моделирования и скульптинга позволяют эффективно создавать low poly модели.

– активное сообщество. Большое количество обучающих материалов, форумов и плагинов, поддерживающих развитие навыков;

– частые обновления. Регулярные обновления обеспечивают постоянное улучшение функциональности.

Недостатки Blender 3D:

– сложный интерфейс. Многим новичкам может быть сложно освоить интерфейс, однако многочисленные обучающие ресурсы помогают преодолеть этот барьер;

– переизбыток функций. Для начинающих пользователей может быть сложно разобраться в обилии доступных инструментов и настроек.

3ds Max — это профессиональное программное обеспечение от Autodesk, широко используемое в индустрии архитектурной визуализации и геймдизайне.

Преимущества 3ds Max:

– профессиональные инструменты для моделирования. Обширный набор инструментов для полигонального моделирования, идеально подходящих для создания low poly моделей;

– широкая индустриальная поддержка. Программа является стандартом в архитектурной визуализации и геймдизайне;

– плагины и расширения. Широкий выбор плагинов для улучшения функциональности.

Недостатки 3ds Max:

– высокая стоимость. Программа требует значительных затрат на покупку лицензии, что может быть ограничением для индивидуальных пользователей и небольших студий;

– ограниченная поддержка Mac OS. 3ds Max официально не поддерживается на Mac OS, что ограничивает его использование на компьютерах Apple;

– кривая обучения. сложность интерфейса и некоторых функций может вызвать трудности у начинающих пользователей.

Maya — еще одно мощное программное обеспечение от Autodesk, часто используемое в киноиндустрии и анимации.

Преимущества Maya:

– высокая гибкость и мощность. Программа предоставляет продвинутые инструменты для моделирования, анимации и риггинга;

– индустриальный стандарт. Широко используется в киноиндустрии и анимации, что делает ее важным инструментом для профессионалов;

– поддержка скриптинга. Встроенный скриптинг на Python и MEL позволяет автоматизировать задачи и создавать кастомные инструменты.

Недостатки Maya:

– высокая стоимость. как и 3ds Max, Maya требует значительных затрат на покупку лицензии;

– сложный интерфейс. Крутая кривая обучения, особенно для тех, кто не знаком с другими продуктами Autodesk;

– переизбыток функций для low poly. Программа может быть излишне сложной для простых задач low poly моделирования.

ZBrush — это специализированное программное обеспечение для скульптинга и детализированного моделирования, разработанное компанией Pixologic.

Преимущества ZBrush:

– мощные инструменты для скульптинга. Уникальные инструменты для создания высокодетализированных моделей;

– инновационные функции. Такие функции, как Dynamesh и ZRemesher, позволяют легко управлять топологией модели;

– отлично подходит для создания высоко детализированных моделей. возможности ретопологии позволяют преобразовать высокополигональные модели, как следствие, такие мощности и возможности попросту не нужны для создания моделей в, выбранном для данного проекта, стиле.

Недостатки ZBrush:

– сложность использования для новичков. Интерфейс и рабочий процесс могут быть непривычными и сложными для начинающих пользователей.

– стоимость. Программа требует значительных затрат на покупку лицензии.

– ограниченные возможности для традиционного полигонального моделирования. Не так эффективен для задач, требующих стандартного полигонального моделирования.

Blender 3D является отличным выбором для создания low poly моделей благодаря своей бесплатности, широкому набору инструментов и активному сообществу. Хотя 3ds Max, Maya и ZBrush предлагают мощные возможности и профессиональные инструменты, их высокая стоимость и сложность освоения могут быть барьерами для начинающих пользователей и небольших студий. Blender 3D предоставляет сбалансированное решение, которое позволяет эффективно создавать качественные low poly модели без значительных финансовых вложений.

1.4.2 Выбор программного обеспечения для редактирования изображений

Для редактирования текстур и графического интерфейса игрового приложения был выбран Adobe Photoshop. Этот профессиональный редактор изображений, разработанный компанией Adobe Inc., зарекомендовал себя как

лидер в индустрии благодаря многолетнему присутствию на рынке и постоянным обновлениям. Эти обновления не только поддерживают программу на современном уровне, но и регулярно добавляют новые функции, расширяющие возможности пользователей.

Adobe Photoshop предоставляет мощный и разнообразный функционал, включающий инструменты для редактирования и создания графики, работы со слоями и масками, применения различных фильтров и эффектов. Все это делает его идеальным выбором для создания высококачественных текстур и графических элементов игрового интерфейса.

Кроме того, Adobe Photoshop поддерживает обширную библиотеку обучающих материалов. Пользователи имеют доступ к многочисленным видео-урокам, статьям и форумам, что значительно упрощает процесс освоения программы и позволяет быстро найти решения для возникающих задач. Такая поддержка со стороны сообщества и компании-разработчика делает Photoshop незаменимым инструментом для профессионалов и начинающих художников, работающих над проектами любой сложности.

1.4.3 Выбор среды разработки игрового приложения

Для разработки игрового приложения был выбран Unity engine. Это кроссплатформенный игровой движок, разработанный Unity Technologies. Он широко используется в индустрии для создания 2D и 3D игр, а также интерактивных симуляций и других визуализаций.

Для написания скриптов движок использует язык программирования C#, считающийся несложным в изучении и работе. В редакторе Unity простой интерфейс, что позволяет легко производить отладку игры. Главными преимуществами Unity считают кросс-платформенность и наличие визуальной среды разработки [1]. Движок поддерживающая более 25 платформ, включая Windows, macOS, iOS, Android, PlayStation, Xbox и другие. Это позволяет разрабатывать игры, которые могут быть легко портированы на различные устройства и операционные системы.

Интерфейс данной среды разработки представлен на рисунке 1.19.

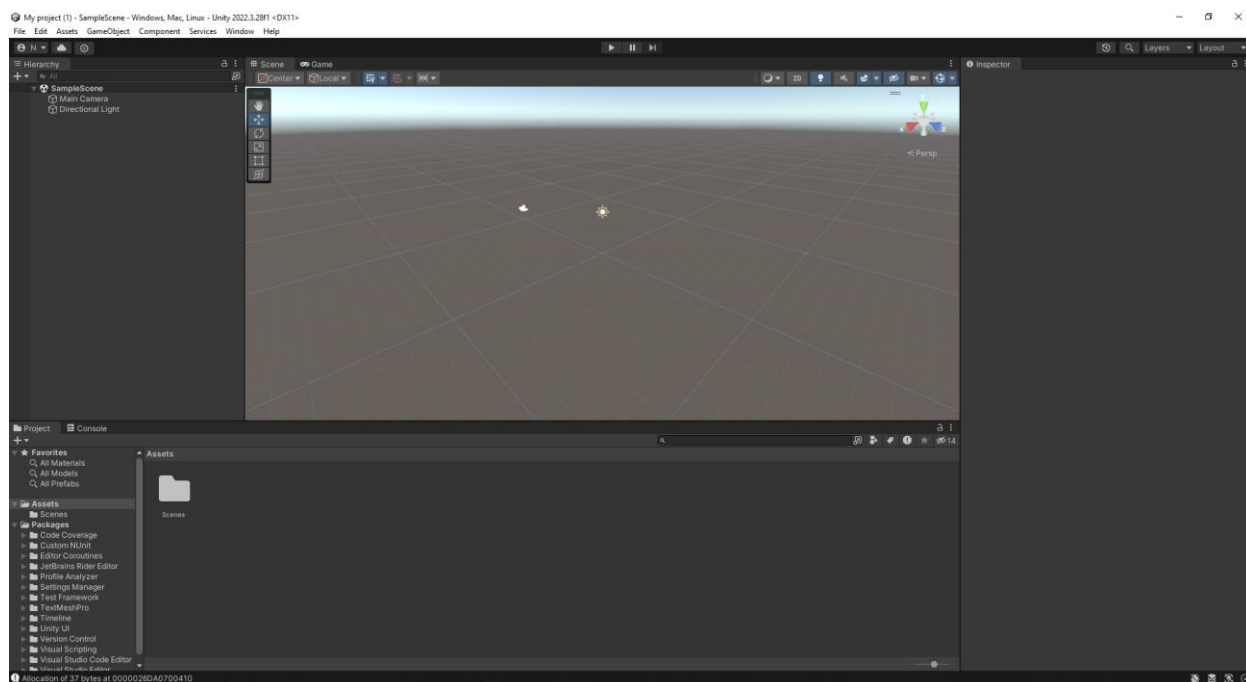


Рисунок 1.19 – Интерфейс Unity

1.4.4 Выбор языка программирования и среды редактирования кода

Как уже было сказано движок, который мы используем в разработке поддерживает C# - это объектно-ориентированный язык программирования, разработанный компанией Microsoft. Он является частью платформы .NET и широко используется для разработки различных приложений, включая веб-приложения, настольные приложения и игры. В контексте Unity, C# является основным языком программирования, используемым для написания скриптов и управления игровыми объектами.

C# обладает четким и понятным синтаксисом, что облегчает его изучение и использование. Он поддерживает объектно-ориентированные принципы программирования, такие как наследование, полиморфизм и инкапсуляция. В Unity C# позволяет легко взаимодействовать с игровыми объектами, использовать встроенные функции и API Unity, а также создавать сложные игровые механики. Строгая типизация в C# помогает избежать многих ошибок

на этапе компиляции и обеспечивает высокую надежность и стабильность кода [2].

Для редактирования скриптов используется среда Visual Studio (VS) code – это бесплатный, кроссплатформенный редактор исходного кода, разработанный компанией Microsoft. С момента своего релиза в 2015 году он быстро стал одним из самых популярных инструментов среди разработчиков благодаря своей гибкости, производительности и богатому набору функций. VS Code является мощным и гибким инструментом, идеально подходящим для разработки игр с использованием Unity и C#. Его кроссплатформенность, производительность, расширяемость и поддержка множества языков делают его одним из лучших редакторов кода на сегодняшний день. Использование VS Code в сочетании с Unity и C# предоставляет разработчикам все необходимые инструменты для эффективного и продуктивного рабочего процесса. Визуально интерфейс показан на рисунке 1.20.

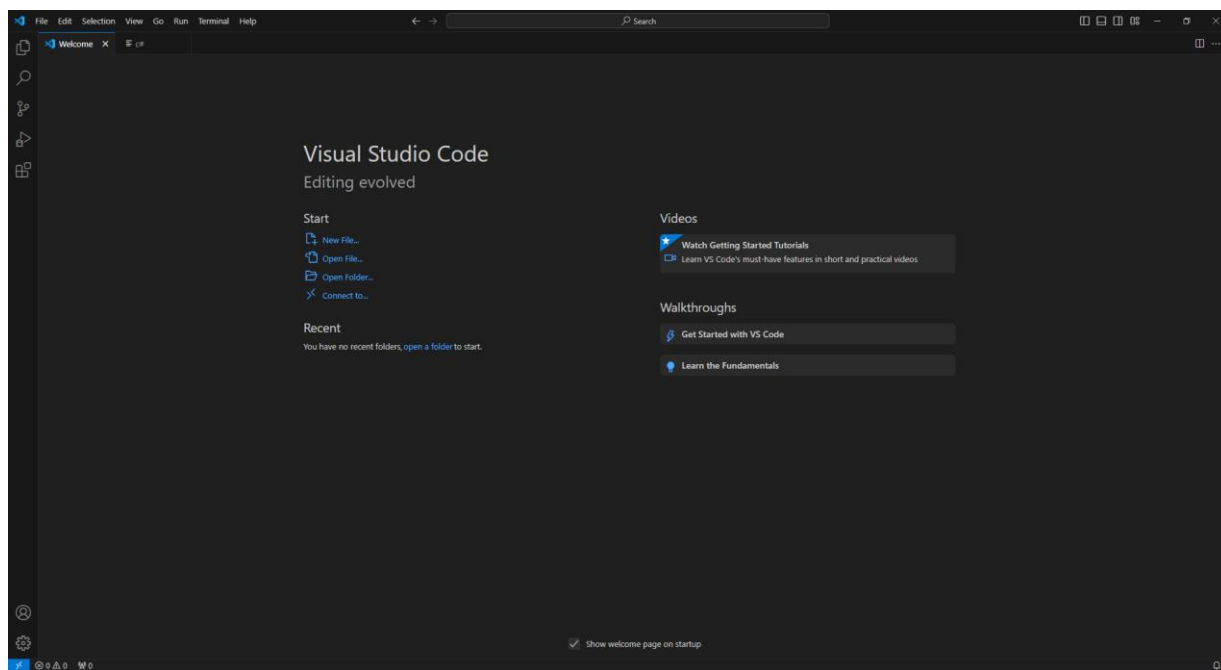


Рисунок 1.20 – Редактор кода Visual Studio Code

1.4.5 Выбор программы для создания музыкального сопровождения игры

Для создания музыкального сопровождения игры был выбран FL Studio, профессиональный цифровой аудиоредактор, разработанный компанией Image-Line. Этот выбор обоснован рядом преимуществ, которые делают FL Studio идеальным инструментом для создания высококачественного эмбиентного фона и другой музыкальной составляющей игрового приложения. На рисунке 1.21 показано рабочее пространство программы.

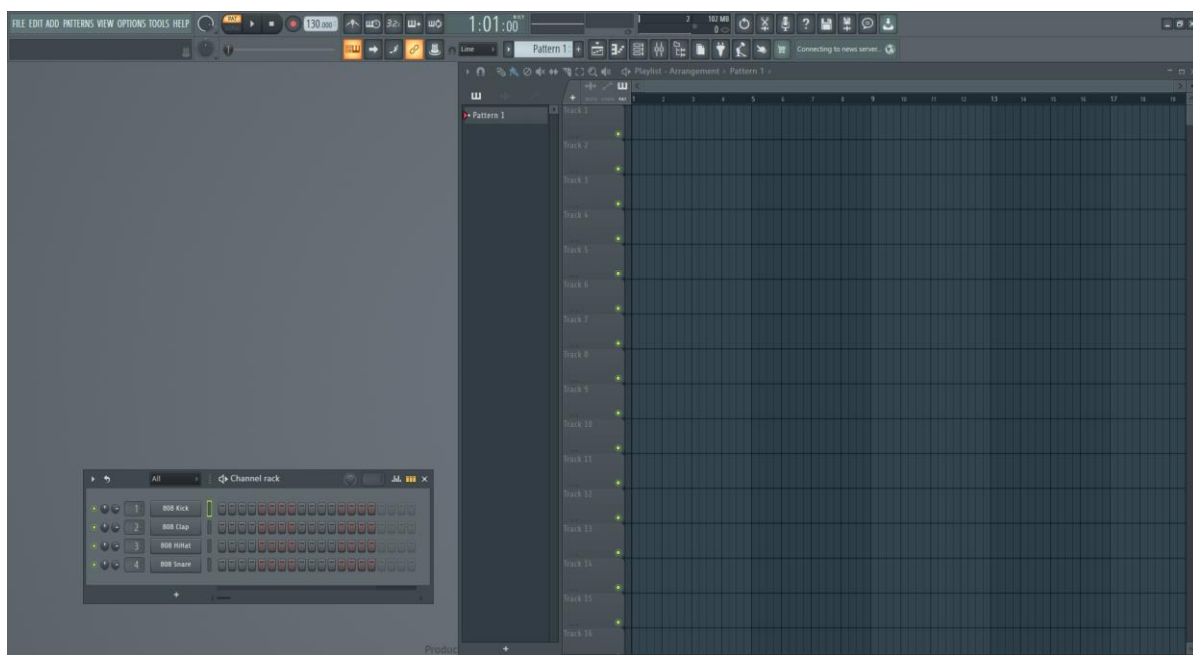


Рисунок 1.21 – Интерфейс Fl Studio

Преимущества FL Studio:

- интуитивный интерфейс. FL Studio славится своим интуитивным и удобным интерфейсом, который делает процесс создания музыки доступным даже для новичков. Программа предлагает множество окон и панелей, которые можно настраивать под свои нужды, что облегчает работу с проектами различной сложности;

- широкий набор инструментов и эффектов. FL Studio предоставляет богатый набор встроенных инструментов, синтезаторов и эффектов, которые позволяют создавать любые виды музыкальных композиций, включая эмбиент;

– мощный секвенсор. FL Studio обладает мощным секвенсором, который позволяет легко создавать и редактировать музыкальные треки. Вы можете быстро создавать ритмы, мелодии и аранжировки с помощью пошагового секвенсора и Piano Roll;

– поддержка VST и AU плагинов. программа поддерживает VST и AU плагины, что позволяет интегрировать дополнительные инструменты и эффекты, расширяя функциональные возможности FL Studio. Это особенно полезно для создания уникального звукового сопровождения.

Эмбиентная музыка требует особого подхода, поскольку она должна создавать атмосферу и подчеркивать игровые события, не отвлекая игрока.

FL Studio является мощным и гибким инструментом для создания музыкального сопровождения игр. Его богатый набор функций, удобный интерфейс и поддержка множества плагинов делают его отличным выбором для создания эмбиентной музыки. Благодаря FL Studio можно создать атмосферное и захватывающее звуковое сопровождение, которое будет эффективно поддерживать и обогащать игровой процесс.

1.5 Архитектура графической составляющей приложения

1.5.1 Сцены уровней

В игровом приложении представлено шесть сцен, каждая из которых является отдельным игровым уровнем. Эти сцены включают:

- детская;
- прихожая;
- кухня-гостиная;
- гараж;
- задний двор;
- амбар.

Каждая из этих сцен разработана с учетом уникальной атмосферы и сюжетной линии, которые помогают игрокам погружаться в игровой мир. Сцены создаются таким образом, чтобы передавать ощущение реальности и визуально обогащать игровой процесс.

1.5.2 Объекты

Объекты, присутствующие в каждой сцене, делятся на две основные категории:

– интерактивные объекты. Эти объекты позволяют игрокам взаимодействовать с ними, выполняя различные действия. Интерактивные объекты могут включать двери, которые можно открыть, предметы, которые можно поднять или использовать, и прочие элементы, с которыми игроки могут взаимодействовать для решения головоломок;

– элементы декора. Хотя с этими объектами нельзя взаимодействовать, они играют важную роль в создании атмосферы и рассказе истории игрового мира. Декоративные элементы помогают визуально обогатить сцены, делая их более глубокими и интересными для изучения.

1.5.3 Текстуры

Для всех объектов в приложении используется единая текстура. Эта текстура выполняет функцию палитры цветов, используемых для всех объектов. Единая текстура помогает создавать гармоничный визуальный стиль и упрощает процесс разработки, так как позволяет использовать одни и те же цветовые решения для различных элементов. Текстура содержит различные цвета и оттенки, которые применяются к объектам для создания целостного и эстетически приятного игрового мира.

1.5.4 Графический интерфейс (UI)

Графический интерфейс (UI) позволяет пользователю взаимодействовать с игровым приложением. Основные функции интерфейса включают:

– вращение сцены. Пользователь может изменять угол обзора сцены, что позволяет рассмотреть объекты под разными углами, что дает возможность лучше реализовать некоторые головоломки;

– меню. Доступ к различным настройкам и опциям через меню;

– регулировка громкости. В интерфейсе предусмотрены настройки для регулировки громкости звуковых эффектов и музыки, а также возможность полного отключения звука.

Графический интерфейс играет важную роль в обеспечении удобства взаимодействия с приложением, позволяя пользователю легко управлять игровыми процессами и настраивать параметры в соответствии с личными предпочтениями.

Таким образом, основными компонентами игрового приложения являются сцены уровней, объекты, текстуры и графический интерфейс. Каждая сцена содержит интерактивные объекты и элементы декора, создавая уникальную атмосферу и помогая рассказывать историю игрового мира. Единая текстура обеспечивает гармоничный визуальный стиль, а графический интерфейс позволяет пользователю удобно управлять игровым процессом и настройками. Эти компоненты вместе создают целостное и увлекательное игровое приложение, предлагающее пользователям насыщенное и интерактивное игровое переживание.

1.6 Реализация

1.6.1 Исследование и сбор референсов

Первым шагом стало исследование и поиск примеров, которые могли бы послужить вдохновением для будущих уровней и объектов. Были собраны разнообразные референсы из различных источников, таких как:

– существующие игры. Изучение успешных игр, схожих по жанру, помогло понять, какие элементы и механики наиболее эффективны;

– фильмы и литература. Кино и книги предоставили визуальные и сюжетные идеи, которые можно было адаптировать в игровой мир.

1.6.2 Разработка основных механик

После сбора референсов началась разработка основных механик игры. Это включало определение того, как игрок будет взаимодействовать с игровым миром, какие действия он сможет выполнять и какие задачи перед ним будут поставлены. Основные механики включали:

- интерактивные объекты. Определение типов объектов, с которыми игроки могут взаимодействовать, и разработка способов этого взаимодействия;
- паззлы и головоломки. Создание логических задач, которые игрокам предстоит решать для продвижения по сюжету;
- навигация и управление. Определение способов перемещения по игровому миру и взаимодействия с интерфейсом.

1.6.3 Проработка сюжетной линии

Следующим важным этапом стала разработка сюжетной линии. Этот процесс включал:

- определение основного сюжета. Разработка главной сюжетной линии, которая будет проходить через все уровни и связывать их в единое целое;
- планирование событий. Определение ключевых событий, которые будут происходить в каждой сцене, и их влияния на развитие сюжета.

1.6.4 Стилистика и визуальное оформление

Для создания уникальной атмосферы игры была разработана стилистика, включающая визуальное оформление и цветовую палитру. Это включало:

- выбор художественного стиля. Определение общей визуальной эстетики, которая будет использоваться на всех уровнях;
- создание текстур и моделей. Разработка текстур, которые будут применяться ко всем объектам, и создание 3D моделей, соответствующих выбранному стилю.

1.6.5 Планирование уровней

На основе собранных референсов, разработанных механик и сюжетной линии было проведено планирование уровней. Это включало:

- составление макетов уровней: создание схематичных макетов каждого уровня с размещением ключевых объектов и интерактивных элементов.
- определение задач и головоломок: распределение задач и головоломок по уровням, чтобы обеспечить интересный и сбалансированный игровой процесс.
- проработка деталей: определение всех деталей, которые будут присутствовать на уровнях, включая декоративные элементы и мелкие объекты, создающие атмосферу.

Эти подготовительные этапы позволили создать прочную основу для дальнейшей разработки уровней и объектов, обеспечив целостность и увлекательность игрового опыта.

Теперь подробнее взглянем на этапы создания графики. Был создан проект в Blender 3D для прототипирования сцен, используемых в дальнейшем в игре. Он представляет собой сборку всех уровней с в виде кубических комнат, расположенных в том порядке, который будет использован в игре. Это показано на рисунке 1.22. Основная часть работы будет выполнена во вкладке layout. По умолчанию здесь открыто окно 3D Viewport – это основное окно, где визуально представлены 3D модели. Справа открыты: сверху outliner – группированные списки моделей, и снизу properties – настройки объектов, рендер, модификаторы и так далее.

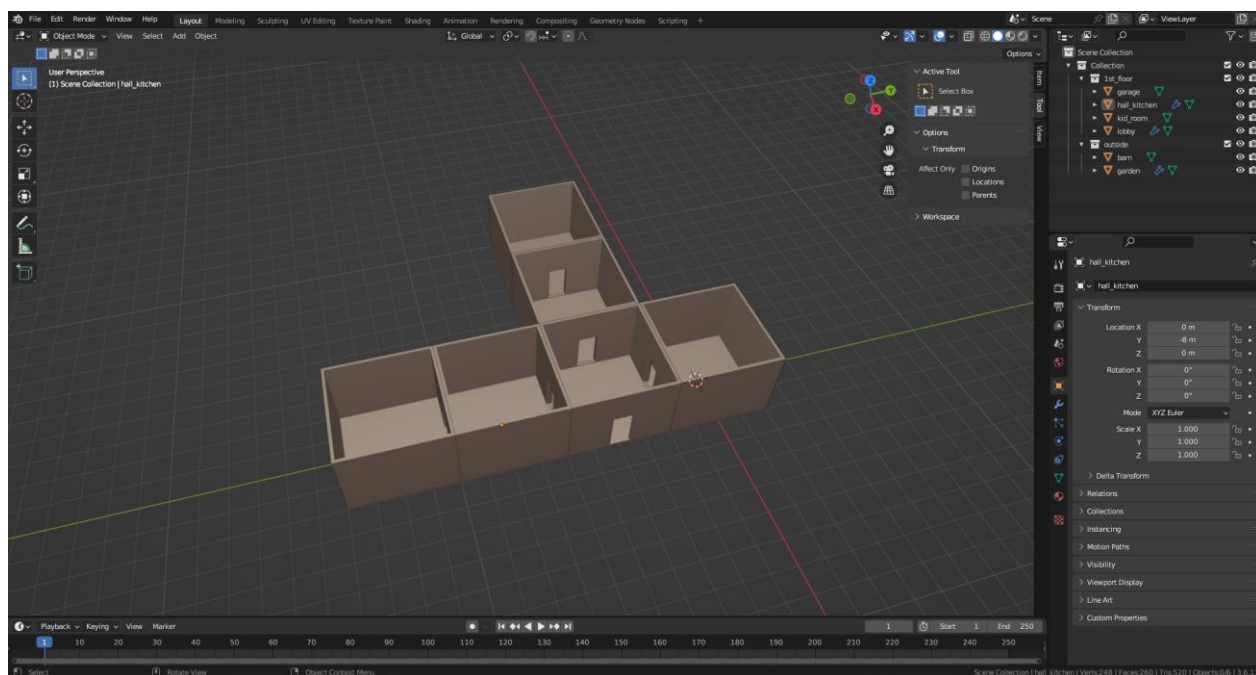


Рисунок 1.22 – Визуальное представление последовательности уровней

Для того чтобы все комнаты были идентичны по размеру был использован инструмент дублирования Duplicate Objects. Для того, чтобы им воспользоваться, нужно либо вызвать контекстное меню инструментов, как на рисунке 1.23, либо воспользоваться сочетанием клавиш. Вообще, большинство функций в этой программе можно попользоваться как раз через сочетания двух и более клавиш. Это очень ускоряет процесс разработки, потому что не нужно постоянно смещать фокус внимания на поиск и выбор функций. Полный список всех сочетаний можно найти в официальной документации к программе Blender 3D. Но для нашей работы не будут использоваться некоторые из них по причине отсутствия необходимости в полном наборе функций.

Изм.	Лист	№ докум.	Подпись	Дата

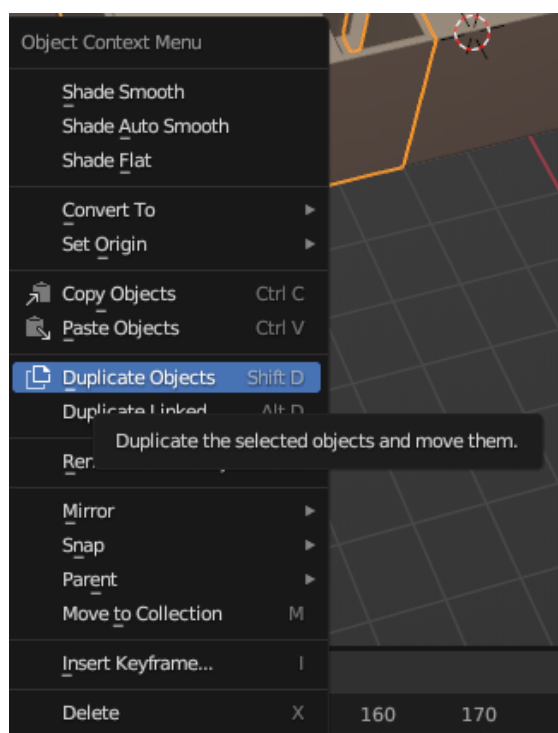


Рисунок 1.23 – Функция Duplicate Objects

Далее с помощью инструмента Move все объекты были расставлены в нужном порядке. Данный инструмент будет очень часто использоваться в дальнейшей работе на пару с Rotate. У каждого из них есть привязки к осям координат, что помогает при построении сцены. Поворот и перемещение по нужной оси в сочетании с набором на клавиатуре нужного градуса и расстояния с помощью цифр помогает добиться симметрии и реализации задуманного расположения объектов на сцене.

Еще одним немаловажным инструментом для реализации объектов является переход из режима объектов (Objective Mode), нужного для их расположения, в режим их редактирования (Edit Mode). С помощью него были созданы отверстия для дверей [3]. В дальнейшей работе этот режим будет использоваться постоянно для создания сложных форм объектов.

После наброска основного расположения сцен началась работа над их наполнением. Каждый объект создавался по референсам с соблюдением общих размеров и стилистики уровней.

Основной процесс создания моделей выглядел так:

- добавление на сцену меша основной формы объекта, чаще использовались square и circle, у последнего при добавлении задавалось необходимое количество вершин;

- создание основной формы объекта с помощью инструмента Extrude.

Данный инструмент может быть применен как к отдельным вершинам меша, так и к граням, и плоскостям объекта. Основа его назначения – это добавление дополнительных вершин, граней и плоскостей, которые автоматически связываются с уже имеющимися и повторяют их форму. Это позволяет буквально наращивать на объект дополнительную геометрию.

- далее идет необязательная для всех объектов часть – использование модификаторов. В данном проекте понадобились два из них:

- первый модификатор Mirror – нужен для зеркального дублирования геометрии объекта, что сокращает время создания модели и дает четкую симметрию [4]. Самым лучшим примером будет служить модель на рисунке 1.24. Кузовная часть отзеркалена по оси Y.

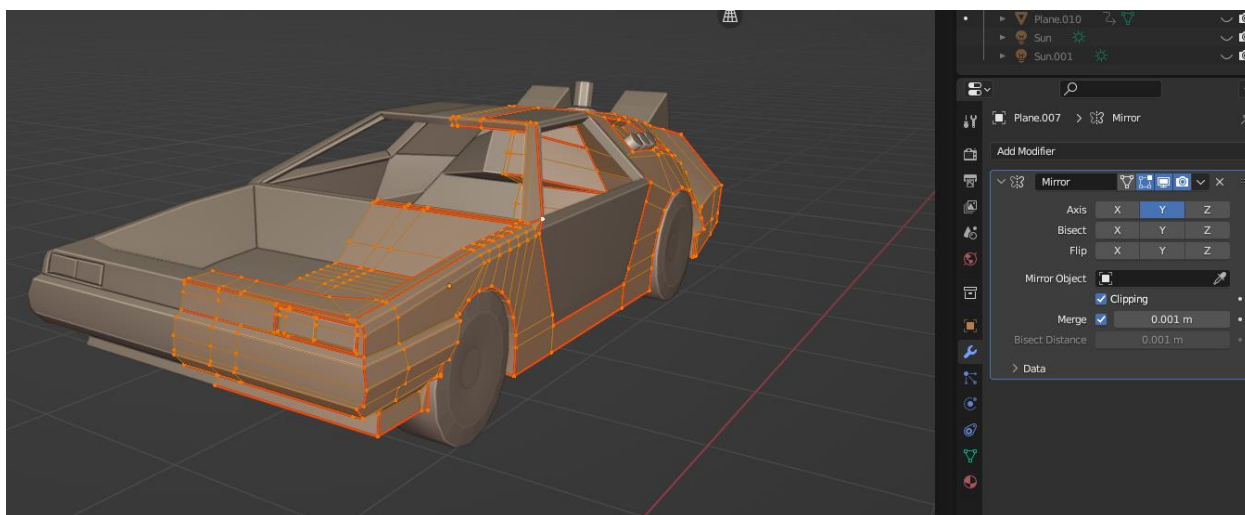


Рисунок 1.24 – Примененный модификатор Mirror для создания симметричной геометрии

– второй модификатор – Array – использовался для неоднократного дублирования одного и того же объекта. Нужно это, например, при моделировании лестниц или в нашем случае жалюзи на двери в гараж. На рисунке 1.25 показано, как один объект продублирован 29 раз по локальной для него оси X.



Рисунок 1.25 – Примененный модификатор Array для создания идентичных мешей

– после того как объект применял необходимые формы, необходимо подготовить UV-развертку. UV-развертка это двумерное представление 3D объекта, которое образуется после определения швов на модели и применения специального инструмента UV unwrap. Этот процесс используется в 3D-графике для того, чтобы координаты текстуры (UV-координаты) соответствовали вершинам, рёбрам и граням 3D-модели.

Полученная UV-карта часто требует ручной корректировки для минимизации искажений и оптимального размещения текстуры. Это делается с помощью инструментов перемещения, масштабирования и вращения в редакторе UV. В данной работе для оптимизации ресурсов, как на разработку, так и

вычислительных в конечном виде, был применен альтернативный подход к созданию UV-развертки и текстур.

Каждый объект на сценах не имеет швов для разреза, а развертка делалась через автоматический сброс развертки. В этом случае все грани объектов будут принимать единый вид квадрата, что видно на рисунке 1.26.



Рисунок 1.26 – вид полигонов на UV-разветке

Сделано это для того, чтобы сократить время на создание, а также дает возможность использовать технологию overlapping. Дословно это означает перекрытие, то есть мы можем накладывать грани объекта друг на друга без потери нужных цветов текстуры при создании развертки. Данное решение позволяет распределить грани на развертке так, что даже для разных объектов они могут находиться на одном и том же месте. Это показано на рисунке 1.27.

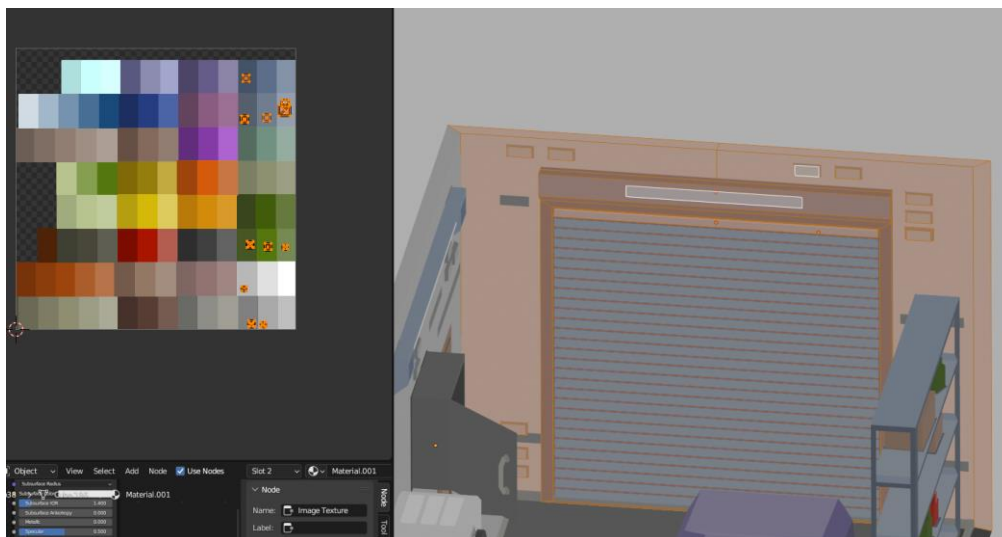


Рисунок 1.27 – Overlapping на UV-развертке

Важно отметить, что для всех объектов в проекте используется всего одна PNG-текстура. Этот подход приносит несколько значимых преимуществ:

Во-первых, в итоговом приложении экономятся ресурсы памяти устройства. Используется одно изображение с разрешением 1024x1024 пикселя, размер которого составляет всего 14 КБ.

Во-вторых, это ускоряет запуск приложения, так как загружается лишь одно текстурное изображение.

В-третьих, данный подход позволяет увеличить количество полигонов в финальной сцене без потери производительности.

В-четвёртых, это решение идеально подходит для реализации графики в стиле минимализм. Low poly объекты в сочетании с простыми мягкими цветами без узоров, градиентов и дополнительных рисунков выглядят органично в общей картине.

При создании текстуры использовалось программное обеспечение Adobe Photoshop. Изначально был указан нужный размер полотна, который описан выше. После отдельными слоями были добавлены элементы палитры. Важно было подобрать цвета, которые подходили бы друг другу по стилю, а именно легкие, немного приглушенные. Также с помощью инструмента Палитра цветов, были выбраны схожие оттенки для каждого. Такое решение позволяет при процессе текстурирования делать акцент на гранях, которые скрыты от источника света или наоборот к нему обращены, так как физического источника света на сценах нет. Пример подбора цвета для палитры в программе редактирования изображений показан на рисунке 1.28.

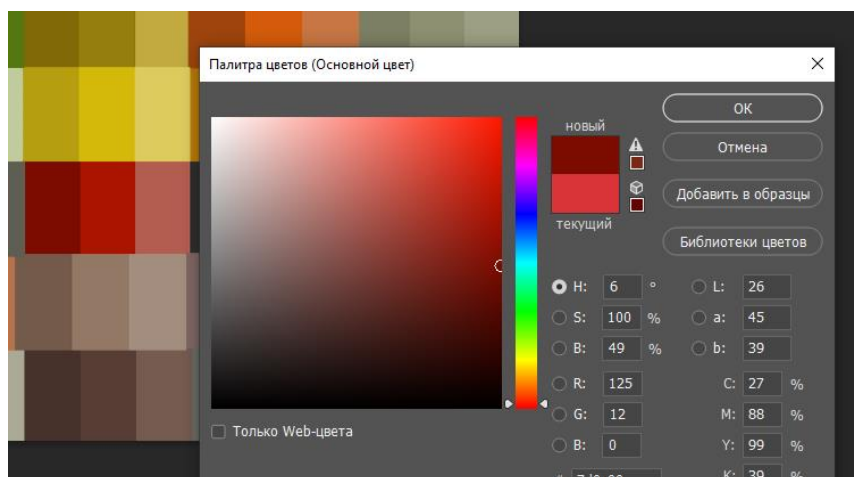


Рисунок 1.28 – Выбор цветов для текстурной палитры

После завершения работы моделирования объектов на сценах, их необходимо было экспортировать в формате .fbx. А уже после импортировать в проект на Unity.

1.6.6 Создание музыкального сопровождения

Следующим этапом является создание музыкального сопровождения игры. Было принято решение создать одну мелодию эмбиент.

Главные требования звучания:

- ненавязчивое и не отвлекающее игрока от процесса;
- подходящее под настроение загадочности и неизвестности происходящего;
- плавный переход между началом и концом мелодии для удобства заикливания мелодии.

Мелодия создана в программе FL studio. Для нее были выбраны 3 вида синтезаторов и один ударный инструмент, что видно на рисунке 1.29.



Рисунок 1.29 – Основные инструменты для создания мелодии

Далее написана основная мелодия с использованием первого синтезатора 3x Osc и на него наложены плагины для того что бы придать индивидуальность звучания. Добавлены остальные два синтезатора с повторяющимися нотами на заднем плане. К ним так же применены плагины. И завершающим этапом было добавление ударных, что добавило ритмичности мелодии, и небольшого эффекта постороннего присутствия. В конечном итоге проект показан на рисунке 1.30.

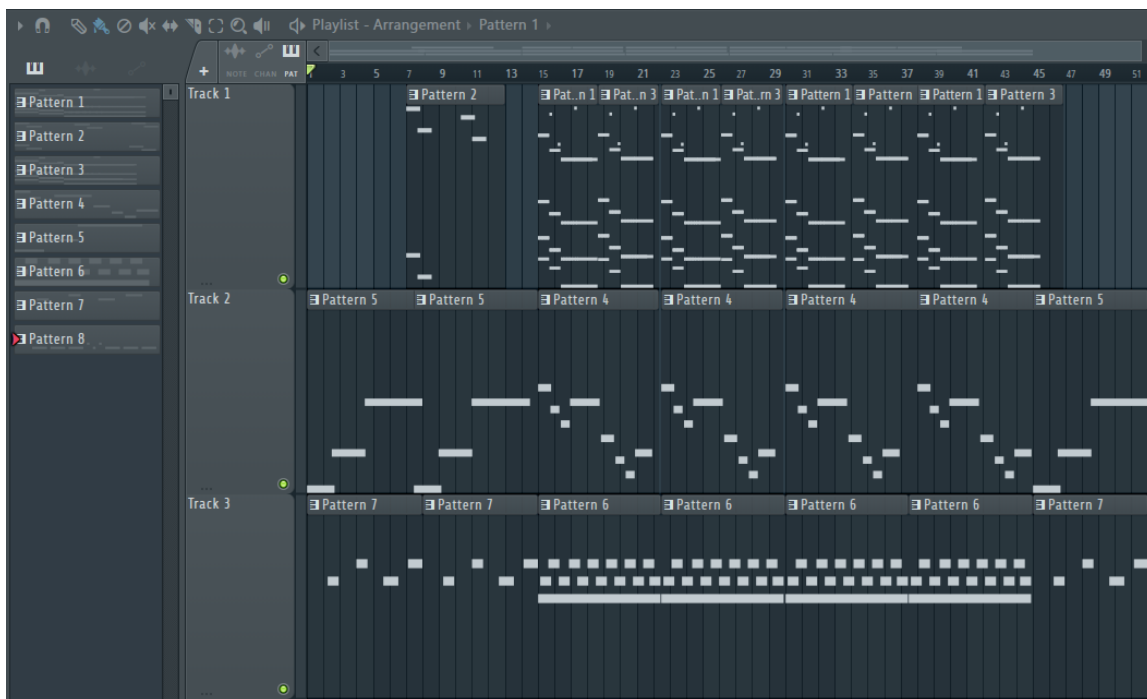


Рисунок 1.30 – Общее представление всего законченного проекта

1.6.7 Реализация меню настроек приложения

Создание меню в Unity включает несколько шагов, от проектирования пользовательского интерфейса до написания кода для управления настройками. Стоит задача создать меню с кнопкой открытия, панель настроек с двумя переключателями (для включения/выключения звуков и музыки), двумя слайдерами для регулирования громкости и кнопкой выхода из игры.

Для начала надо будет добавить элементы пользовательского интерфейса (UI). В окне Hierarchy выбирал Canvas. Это создаст новое полотно, которое будет содержать все UI элементы. Canvas необходимо установить в режиме Screen Space - Overlay для стандартного меню [5].

Далее создал кнопку для вызова меню. Внутри Canvas нужно нажать правой кнопкой мыши и выбирать Button. Это создаст кнопку, которая в дальнейшем переименовывается в "MenuButton". Далее настраивается её внешний вид и позиция в инспекторе, чтобы она была доступна в верхнем правом углу экрана.

Теперь необходимо создать панель настроек. Настроил её размеры и позицию так, чтобы она занимала значительную часть экрана. По умолчанию панель должна быть скрыта, поэтому установил её компонент CanvasGroup альфа-канал в 0.

После этого добавил элементы на панель. Внутри панели "SettingsPanel" добавил текстовые метки для описания настроек. Расположил переключатели для включения/выключения звуков и музыки, слайдеры для регулирования громкости звуков и музыки, и кнопку выхода, которая имеет название - "ExitButton". Далее необходимо написать скрипты.

SoundManager реализует менеджер звуков, который отвечает за воспроизведение звуковых эффектов и музыки в игре. Он использует паттерн Singleton для обеспечения того, что в игре будет только один экземпляр этого класса. С его помощью релизованы настройки звуковых эффектов и музыки, а именно: включение/выключение звука и музыки, а также настройка их

громкости. В проект загружен полный набор звуков, которые были взяты из открытых источников, а также импортирован эмбиент, речь о котором шла выше. Данный скрипт находит звук по его названию, это удобно для добавления определенных звуков для конкретных объектов. Все аудиоклипы загружаются из папки Resources/Sounds и добавляются в словарь audioClips [6].

MenuManager управляет интерфейсом настроек игры, обеспечивая взаимодействие пользователя с параметрами звука и музыки. Он тесно связан с первым скриптом SoundManager, который отвечает за управление звуковыми эффектами и музыкой. Набор методов в данном скрипте выполняет следующие задачи:

- переключение видимости панели настроек, то есть если панель была скрыта, она станет видимой, и наоборот;
- вызов соответствующих методов из SoundManager для изменения настроек звука и музыки (включение/выключение для звуков и музыки и установка уровня громкости);
- реализация кнопки закрытия игрового приложения.

Таким образом, MenuManager обеспечивает пользовательский интерфейс для управления звуковыми настройками игры. Все изменения в настройках передаются в SoundManager, который отвечает за непосредственное управление воспроизведением звуков и музыки.

1.6.8 Тестирование приложения

В ходе проверки и тестирования было найдено множество багов, связанных с 3D объектами и сценами. Было исправлено положение линий нормали для корректного отображения мешей на сцене, также изменены pivot point (точки основания объектов) для более удобного размещения объектов на сцене, трансформации, размещения коллайдеров и анимирования. Это поспособствовало правильному отображению всех объектов, а также привести к эстетически приятному виду графическую составляющую.

Благодаря оптимизации графики на стадии моделирования и отрисовывания проблем с производительностью на персональных компьютерах и ноутбуках выявлено не было.

					<i>ПГУ1.090301.14.001 ПЗ</i>	Лист
						49
Изм.	Лист	№ докум.	Подпись	Дата		

2 Охрана труда

2.1 Организация рабочего места при работе на персональном компьютере

В современном мире информационных технологий вопросы безопасности жизнедеятельности становятся все более актуальными. При разработке программного обеспечения, как и в любой другой области деятельности, необходимо учитывать требования охраны труда и обеспечения безопасности. В данном разделе рассматриваются аспекты безопасности жизнедеятельности, связанные с профессиональной деятельностью программиста.

Согласно ГОСТ 12.2.032-78 "ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования", рабочее место программиста должно быть организовано таким образом, чтобы минимизировать вредное воздействие на здоровье.

2.1.1 Освещение

Рабочее место должно быть оборудовано достаточным освещением. Согласно СНиП 23-05-95 "Естественное и искусственное освещение", уровень освещенности на рабочем месте программиста должен составлять не менее 300-500 люкс. Рекомендуется использовать комбинацию естественного и искусственного освещения, чтобы избежать чрезмерной нагрузки на зрение. Необходимо учитывать:

- расположение источников света, чтобы избежать бликов на экране монитора;
- использование регулируемых ламп для индивидуальной настройки освещения.

2.1.2 Общие эргономические требования к рабочему месту оператора

Рабочее место с экраном должно обеспечивать комфортные условия для работы в положении сидя, не вызывая перегрузок костно-мышечной системы оператора.

Основные элементы рабочего места оператора включают рабочий стол, рабочий стул (кресло), дисплей и клавиатуру; дополнительно могут использоваться пюпитр и подставка для ног. Общая организация рабочего места показана на рисунке 2.1.

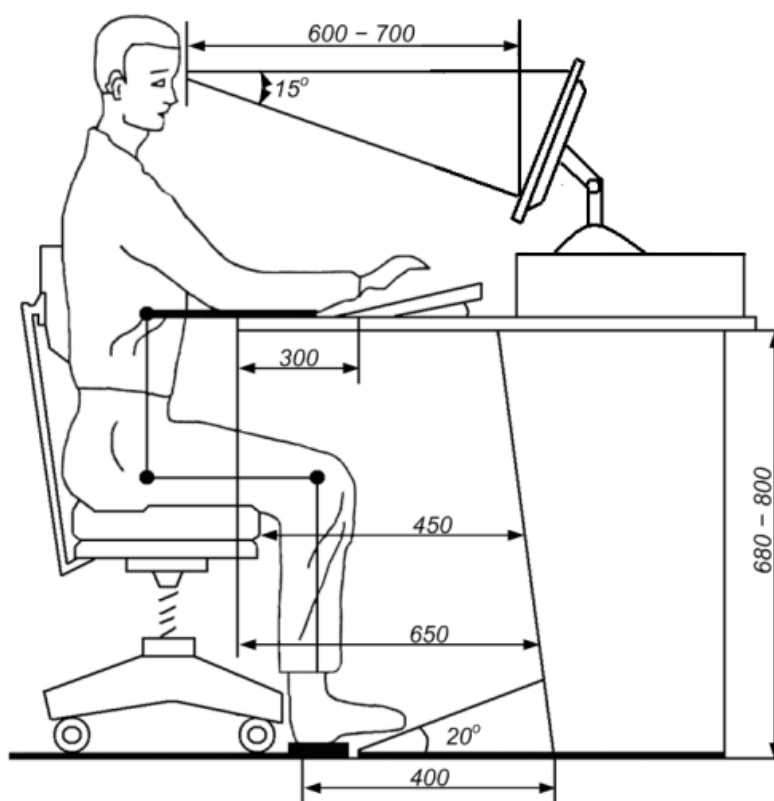


Рисунок 2.1 – Общая организация рабочего места

Рабочий стол должен быть спроектирован таким образом, чтобы на его поверхности можно было разместить необходимое оборудование и документы, учитывая характер работы оператора.

Высота рабочей поверхности стола может быть регулируемой (680-800 мм) или фиксированной (725 мм), в зависимости от конструкции. Размеры стола должны обеспечивать глубину не менее 600 мм и ширину не менее 1200 мм.

Рабочая поверхность стола должна быть без острых углов, покрыта материалом с коэффициентом отражения 0,45-0,50.

Рабочее кресло должно поддерживать физиологически правильное положение тела оператора, позволяя менять позу для снижения напряжения мышц спины и шеи.

Кресло должно быть регулируемым по высоте и углам наклона сиденья и спинки, а также по расстоянию спинки от переднего края сиденья. Подлокотники должны быть регулируемыми по высоте и ширине.

Подставка для ног должна регулироваться по высоте и углу наклона, обеспечивая комфортную поддержку ног оператора.

Поверхность подставки должна быть рифленой с бортиком для предотвращения скольжения.

Дисплей должен быть установлен на рабочем месте таким образом, чтобы оператор мог четко видеть изображение без лишнего напряжения глаз.

Угол наклона экрана относительно горизонтальной линии взгляда оператора не должен превышать 60° , показано на рисунке 2.1.

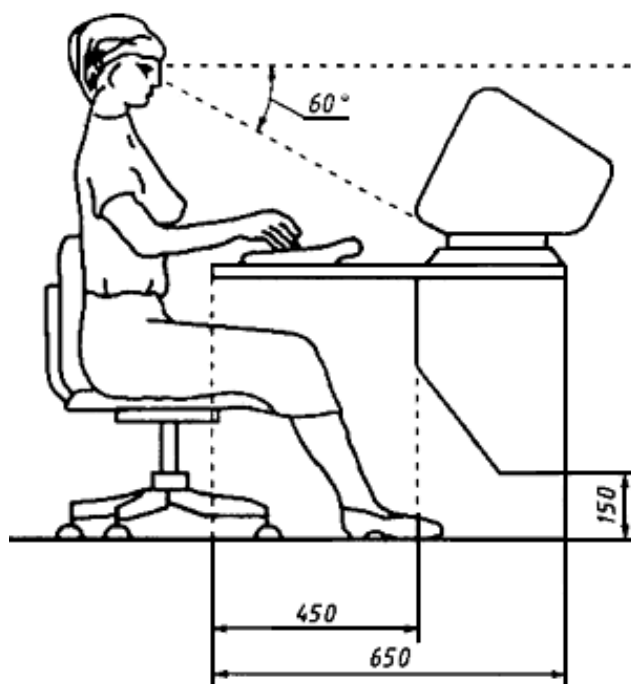


Рисунок 2.2 – Расположение дисплея на столе

Клавиатура должна быть расположена на достаточном расстоянии от оператора, чтобы обеспечивать оптимальную видимость экрана.

Клавиатура должна иметь возможность свободного перемещения или быть установлена на регулируемой по высоте поверхности.

2.1.3 Климатические условия

В помещении, где работает программист, должны поддерживаться оптимальные климатические условия. Согласно ГОСТ 30494-2011 "Здания жилые и общественные. Параметры микроклимата в помещениях", температура воздуха должна составлять 18–22°C, относительная влажность – 40–60%. Также необходимо обеспечить:

- регулярное проветривание помещения;
- использование систем кондиционирования и увлажнения воздуха.

2.2 Психофизиологические аспекты труда программиста

Работа программиста связана с высокой умственной нагрузкой и длительным пребыванием в одной позе, что может приводить к утомлению и снижению работоспособности.

В соответствии с СанПиН 2.2.2/2.4.1340-03 "Гигиенические требования к персональным электронно-вычислительным машинам и организации работы", необходимо соблюдать режим труда и отдыха. Рекомендуется:

- делать перерывы через каждые 1,5-2 часа работы по 10-15 минут;
- выполнять физические упражнения для снятия напряжения с мышц и улучшения кровообращения.

2.3 Электробезопасность

При работе с компьютером программист подвергается риску поражения электрическим током. Для обеспечения электробезопасности необходимо

соблюдать требования ГОСТ 12.1.019-79 "Электробезопасность. Общие требования и номенклатура видов защиты".

2.3.1 Использование исправного оборудования

Все электронные устройства должны быть исправны и иметь сертификацию соответствия стандартам безопасности. Важно регулярно проверять оборудование на предмет повреждений и своевременно устранять выявленные дефекты.

2.3.2 Заземление и защитные устройства

Рабочее место должно быть оснащено розетками с заземлением и устройствами защитного отключения (УЗО), которые предотвращают поражение электрическим током в случае короткого замыкания или утечки тока. Необходимо также соблюдать правила использования удлинителей и сетевых фильтров.

2.4 Пожарная безопасность

Для обеспечения пожарной безопасности в помещении, где работают программисты, должны быть соблюдены требования ГОСТ 12.1.004-91 "Пожарная безопасность. Общие требования".

В помещении должны быть установлены средства пожаротушения (огнетушители), система оповещения о пожаре и пути эвакуации, обозначенные согласно ГОСТ Р 12.4.026-2015 "ССБТ. Цвета сигнальные и знаки безопасности". Важно регулярно проверять исправность средств пожаротушения и наличие свободных путей эвакуации.

Все сотрудники должны проходить регулярные инструктажи по пожарной безопасности, включая действия в случае возникновения пожара и использование средств пожаротушения. Важно также проводить тренировки по эвакуации для отработки действий в чрезвычайных ситуациях.

Заключение

В результате выполнения выпускной квалификационной работы была разработана графическая часть для игрового приложения в жанре изометрическая головоломка. В качестве движка был выбран Unity, графический редактор Adobe Photoshop, а также программное обеспечение для 3D моделирования Blender 3D, язык программирования C#, а для редактирования кода Visual Studio Code.

При создании графической составляющей приложения были учтены и исправлены недостатки, обнаруженные в аналогичных приложениях данного жанра. Разработанное приложение обладает значительным потенциалом для дальнейшего развития, включая расширение игровых механик, добавление новых уровней и сюжетных актов. Представленный в данной работе сюжет может быть завершен, что открывает возможности для привлечения большего количества новых пользователей. Это позволяет не только поддерживать интерес текущих игроков, но и постоянно расширять аудиторию, предлагая им новые и увлекательные элементы геймплея.

Благодаря высокому качеству и перспективам развития, приложение можно будет выложить на онлайн-сервисы цифрового распространения компьютерных игр и программ. Это обеспечит его доступность для широкой аудитории и поможет привлечь новых пользователей, увеличивая популярность и доходность проекта.

Список использованных источников

1. Уточкин, В. Н. Хочу в геймдев! Основы игровой разработки для начинающих / В. Н. Уточкин, К. С. Сахнов. – Москва : Бомбра, 2022. – 224 с.
2. docs.unity.com: [Электронный ресурс]. URL: <https://docs.unity3d.com/2023.2/Documentation/ScriptReference/MonoBehaviour.html> (дата обращения: 9.05.2023).
3. Baechler, O. Blender 3D By Example. Blender 3D на примерах: на англ. яз. / О. Baechler. – Москва : Packt Publishing, 2020. – 658 с. – ISBN 9781789612561.
4. docs.blender.org: [Электронный ресурс]. URL: <https://docs.blender.org/manual/ru/3.6/modeling/modifiers/index.html> (дата обращения: 4.05.2023).
5. Хокинг Дж. Unity в действии. Мультиплатформенная разработка на C# / Дж. Хокинг. – Питер : ООО Издательство «Питер», 2016. – 331 с.
6. docs.unity.com: [Электронный ресурс]. URL: <https://docs.unity3d.com/2023.2/Documentation/ScriptReference/AudioClip.html> (дата обращения: 16.05.2023).

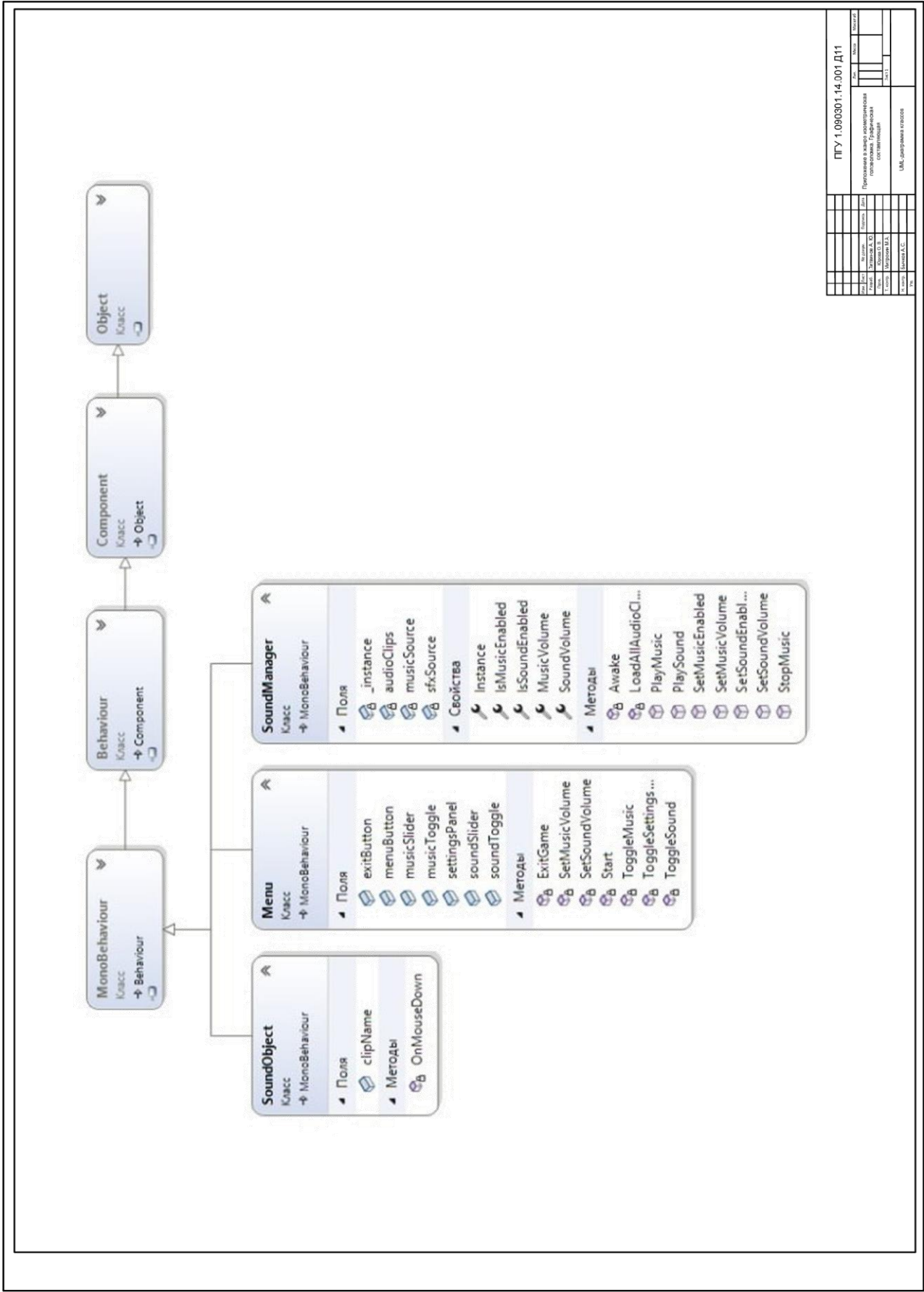
Приложение А

(обязательное)

Приложение в жанре изометрическая головоломка. Графическая
составляющая
UML-диаграммы

					<i>ПГУ1.090301.14.001 ПЗ</i>	Лист
						57
Изм.	Лист	№ докум.	Подпись	Дата		

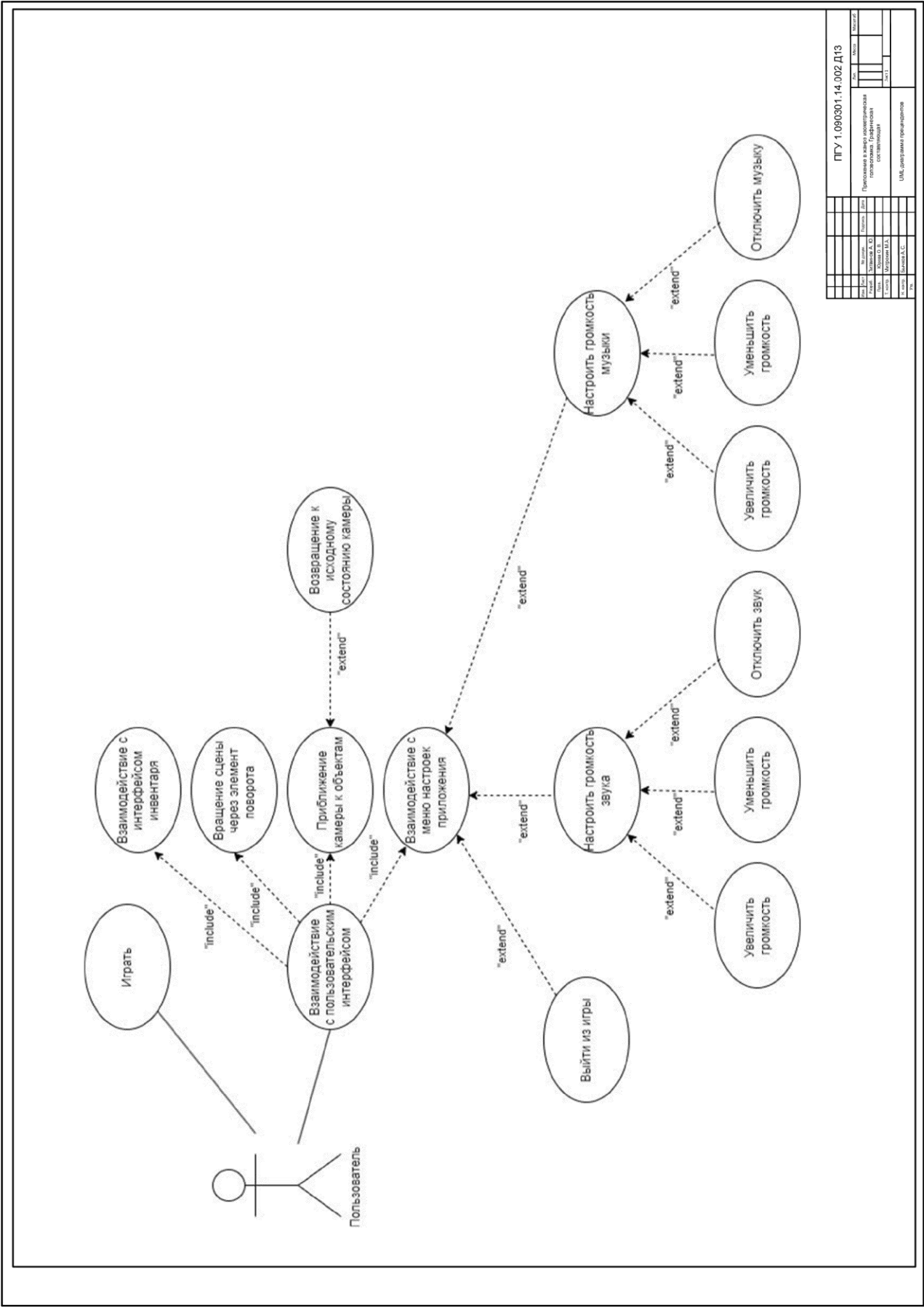
Приложение А.1 – UML-диаграмма классов



ПГУ 1.090301.14.001 Д11									
Изм.	Лист	№ докум.	Подпись	Дата	Изм.	Лист	№ докум.	Подпись	Дата
Приложение к курсовому проекту по дисциплине «УML-моделирование»									
УМЛ-диаграмма классов									

Изм.	Лист	№ докум.	Подпись	Дата

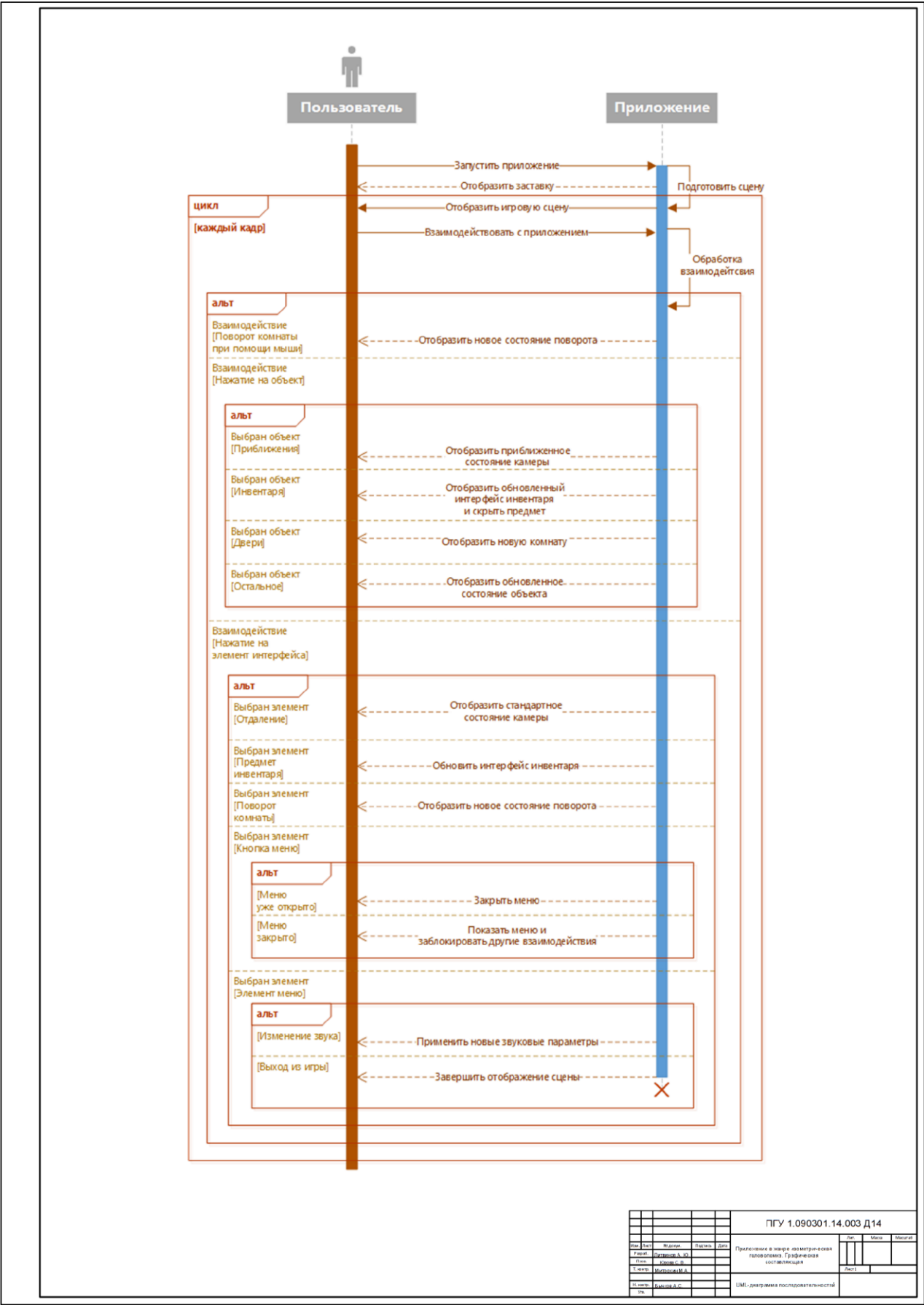
Приложение А.2 – UML-диаграмма прецедентов



ПГУ 1.090301.14.002 Д13			
№ п/п	Ф.И.О.	Подпись	Дата
1	Сидорова А.А.		2023
2	Сидорова А.А.		2023
3	Сидорова А.А.		2023
4	Сидорова А.А.		2023
5	Сидорова А.А.		2023
6	Сидорова А.А.		2023
7	Сидорова А.А.		2023
8	Сидорова А.А.		2023
9	Сидорова А.А.		2023
10	Сидорова А.А.		2023
11	Сидорова А.А.		2023
12	Сидорова А.А.		2023
13	Сидорова А.А.		2023
14	Сидорова А.А.		2023
15	Сидорова А.А.		2023
16	Сидорова А.А.		2023
17	Сидорова А.А.		2023
18	Сидорова А.А.		2023
19	Сидорова А.А.		2023
20	Сидорова А.А.		2023
21	Сидорова А.А.		2023
22	Сидорова А.А.		2023
23	Сидорова А.А.		2023
24	Сидорова А.А.		2023
25	Сидорова А.А.		2023
26	Сидорова А.А.		2023
27	Сидорова А.А.		2023
28	Сидорова А.А.		2023
29	Сидорова А.А.		2023
30	Сидорова А.А.		2023
31	Сидорова А.А.		2023
32	Сидорова А.А.		2023
33	Сидорова А.А.		2023
34	Сидорова А.А.		2023
35	Сидорова А.А.		2023
36	Сидорова А.А.		2023
37	Сидорова А.А.		2023
38	Сидорова А.А.		2023
39	Сидорова А.А.		2023
40	Сидорова А.А.		2023
41	Сидорова А.А.		2023
42	Сидорова А.А.		2023
43	Сидорова А.А.		2023
44	Сидорова А.А.		2023
45	Сидорова А.А.		2023
46	Сидорова А.А.		2023
47	Сидорова А.А.		2023
48	Сидорова А.А.		2023
49	Сидорова А.А.		2023
50	Сидорова А.А.		2023
51	Сидорова А.А.		2023
52	Сидорова А.А.		2023
53	Сидорова А.А.		2023
54	Сидорова А.А.		2023
55	Сидорова А.А.		2023
56	Сидорова А.А.		2023
57	Сидорова А.А.		2023
58	Сидорова А.А.		2023
59	Сидорова А.А.		2023
60	Сидорова А.А.		2023
61	Сидорова А.А.		2023
62	Сидорова А.А.		2023
63	Сидорова А.А.		2023
64	Сидорова А.А.		2023
65	Сидорова А.А.		2023
66	Сидорова А.А.		2023
67	Сидорова А.А.		2023
68	Сидорова А.А.		2023
69	Сидорова А.А.		2023
70	Сидорова А.А.		2023
71	Сидорова А.А.		2023
72	Сидорова А.А.		2023
73	Сидорова А.А.		2023
74	Сидорова А.А.		2023
75	Сидорова А.А.		2023
76	Сидорова А.А.		2023
77	Сидорова А.А.		2023
78	Сидорова А.А.		2023
79	Сидорова А.А.		2023
80	Сидорова А.А.		2023
81	Сидорова А.А.		2023
82	Сидорова А.А.		2023
83	Сидорова А.А.		2023
84	Сидорова А.А.		2023
85	Сидорова А.А.		2023
86	Сидорова А.А.		2023
87	Сидорова А.А.		2023
88	Сидорова А.А.		2023
89	Сидорова А.А.		2023
90	Сидорова А.А.		2023
91	Сидорова А.А.		2023
92	Сидорова А.А.		2023
93	Сидорова А.А.		2023
94	Сидорова А.А.		2023
95	Сидорова А.А.		2023
96	Сидорова А.А.		2023
97	Сидорова А.А.		2023
98	Сидорова А.А.		2023
99	Сидорова А.А.		2023
100	Сидорова А.А.		2023

Изм.	Лист	№ докум.	Подпись	Дата

Приложение А.3 – UML-диаграмма последовательности



ПГУ 1.090301.14.003 Д14					
Имя	Фамилия	Имя	Фамилия	Имя	Фамилия
Имя	Фамилия	Имя	Фамилия	Имя	Фамилия
Имя	Фамилия	Имя	Фамилия	Имя	Фамилия
Имя	Фамилия	Имя	Фамилия	Имя	Фамилия
Имя	Фамилия	Имя	Фамилия	Имя	Фамилия
Имя	Фамилия	Имя	Фамилия	Имя	Фамилия
Имя	Фамилия	Имя	Фамилия	Имя	Фамилия
Имя	Фамилия	Имя	Фамилия	Имя	Фамилия
Имя	Фамилия	Имя	Фамилия	Имя	Фамилия

Изм.	Лист	№ докум.	Подпись	Дата

Приложение Б

(обязательное)

Приложение в жанре изометрическая головоломка. Графическая

составляющая

Листинг программы

					<i>ПГУ1.090301.14.001 ПЗ</i>	Лист
						61
Изм.	Лист	№ докум.	Подпись	Дата		

Файл MenuManager.cs

```
using UnityEngine;
using UnityEngine.UI;

public class MenuManager : MonoBehaviour
{
    public GameObject settingsPanel;
    public GameObject blockPanel;
    public Button menuButton;
    public Button soundButton;
    public Button musicButton;
    public Slider soundSlider;
    public Slider musicSlider;
    public Button exitButton;
    public Sprite soundOnIcon;
    public Sprite soundOffIcon;
    public Sprite musicOnIcon;
    public Sprite musicOffIcon;
    private bool isRoomRotatable;
    private bool isBoxRotatable;

    private void Start()
    {
        settingsPanel.SetActive(false);
        blockPanel.SetActive(false);

        menuButton.onClick.AddListener(ToggleSettingsPanel);
        soundButton.onClick.AddListener(ToggleSound);
        musicButton.onClick.AddListener(ToggleMusic);
        musicSlider.onValueChanged.AddListener(SetMusicVolume_);
        soundSlider.onValueChanged.AddListener(SetSoundVolume_);
        exitButton.onClick.AddListener(Application.Quit);

        soundSlider.value =
SoundManager.instance.startSoundVolume;
        musicSlider.value =
SoundManager.instance.startMusicVolume;
    }

    private void ToggleSettingsPanel()
    {
        settingsPanel.SetActive(!settingsPanel.activeSelf);
        blockPanel.SetActive(settingsPanel.activeSelf);
        if(settingsPanel.activeSelf){
            isRoomRotatable = RoomManager.instance.Rotatable;
            isBoxRotatable = BoxController.instance.Rotatable;
            RoomManager.instance.Rotatable = false;
            BoxController.instance.Rotatable = false;
        } else{
            RoomManager.instance.Rotatable = isRoomRotatable;
        }
    }
}
```

```

        BoxController.instance.Rotatable = isBoxRotatable;
    }
}
private void SetSoundVolume_(float value){
    SoundManager.instance.SetSoundVolume(value);
}
private void SetMusicVolume_(float value){
    SoundManager.instance.SetMusicVolume(value);
}

private void ToggleSound()
{
    SoundManager.instance.SetSoundEnabled(!SoundManager.instance.isSoundEnabled);

    if (SoundManager.instance.isSoundEnabled){
        soundButton.image.sprite = soundOnIcon;
        return;
    }
    soundButton.image.sprite = soundOffIcon;
}

private void ToggleMusic()
{
    SoundManager.instance.SetMusicEnabled(!SoundManager.instance.isMusicEnabled);
    if (SoundManager.instance.isMusicEnabled){
        musicButton.image.sprite = musicOnIcon;
        return;
    }
    musicButton.image.sprite = musicOffIcon;
}
}

```

Файл SoundManager.cs

```

using UnityEngine;
using System.Collections.Generic;

public class SoundManager : MonoBehaviour
{
    private static SoundManager instance_;
    public static SoundManager instance
    {
        get
        {
            instance_ = FindObjectOfType<SoundManager>();
            return instance_;
        }
    }
}

```

```

    }

    public string mainSoundName = "main";
    public AudioSource soundSource;
    public AudioSource musicSource;
    private Dictionary<string, AudioClip> audioClips;
    public bool isSoundEnabled { get; private set; } = true;
    public bool isMusicEnabled { get; private set; } = true;

    [Range(0.0f, 1.0f)]
    public float startSoundVolume = 1.0f;
    [Range(0.0f, 1.0f)]
    public float startMusicVolume = 1.0f;

    private void Start()
    {
        if(soundSource == null || musicSource == null){
            Destroy(this);
            return;
        }

        musicSource.loop = true;
        audioClips = new Dictionary<string, AudioClip>();
        LoadAudioClips();

        musicSource.volume = startMusicVolume;
        soundSource.volume = startSoundVolume;
        PlayMusic(mainSoundName);
    }

    private void LoadAudioClips()
    {
        AudioClip[] clips =
Resources.LoadAll<AudioClip>("Audio");
        foreach (var clip in clips)
        {
            if (audioClips.ContainsKey(clip.name))
                continue;
            audioClips.Add(clip.name, clip);
        }
    }

    public void PlaySound(string key)
    {
        if(!isSoundEnabled)
            return;

        if (!audioClips.ContainsKey(key))
            return;

        soundSource.PlayOneShot(audioClips[key]);
    }

```

```

    }

    public void PlayMusic(string key)
    {
        if (!isMusicEnabled)
            return;

        if (!audioClips.ContainsKey(key))
            return;

        musicSource.clip = audioClips[key];
        musicSource.Play();
    }

    public void StopMusic()
    {
        if (musicSource.isPlaying)
            musicSource.Stop();
    }

    public void SetSoundEnabled(bool flag)
    {
        isSoundEnabled = flag;
    }

    public void SetMusicEnabled(bool flag)
    {
        isMusicEnabled = flag;
        if (!isMusicEnabled && musicSource.isPlaying)
        {
            musicSource.Stop();
        }
        else if (isMusicEnabled && !musicSource.isPlaying)
        {
            musicSource.Play();
        }
    }

    public void SetSoundVolume(float volume)
    {
        soundSource.volume = volume;
    }

    public void SetMusicVolume(float volume)
    {
        musicSource.volume = volume;
    }
}

```

Файл Door.cs

using UnityEngine;

```

public class Door : MonoBehaviour
{
    public RoomController targetRoom;
    public AudioClip clip;

    private void OnMouseDown()
    {
        if (ViewManager.instance.isZoomed)
            return;

        RoomManager.instance.SwitchRoom(targetRoom);
        SoundManager.instance.PlaySound("door");
    }
}

```

					<i>ПГУ1.090301.14.001 ПЗ</i>	Лист
						66
Изм.	Лист	№ докум.	Подпись	Дата		

Приложение В

(обязательное)

Приложение в жанре изометрическая головоломка. Графическая составляющая
Презентация

					<i>ПГУ1.090301.14.001 ПЗ</i>	Лист
						67
Изм.	Лист	№ докум.	Подпись	Дата		

ПРИЛОЖЕНИЕ В ЖАНРЕ ИЗОМЕТРИЧЕСКАЯ ГОЛОВОЛОМКА. ГРАФИЧЕСКАЯ СОСТАВЛЯЮЩАЯ

Автор работы:

Студент 20ВВП1 Литвинов А.Ю.

Научный руководитель:

к.т.н. доцент кафедры ВТ Юрова О.В

Цель работы:

- Разработать графическую часть игрового приложения в жанре «изометрическая головоломка» - приложение должно обладать простым и понятным интерфейсом; эстетически приятной и логично составленной графикой; игровые уровни должны содержать в себе интерактивные и статичные объекты.
- Оснастить игровое приложение звуковыми эффектами при взаимодействиями с объектами, создать музыкальное сопровождение, отвечающее общей атмосфере «головоломки»

Изм.	Лист	№ докум.	Подпись	Дата

ПГУ1.090301.14.001 ПЗ

Лист

68

Популярность и польза ГОЛОВОЛОМОК



Аналоги



Изм.	Лист	№ докум.	Подпись	Дата

ПГУ1.090301.14.001 ПЗ

Лист

69

Выбор инструментов для разработки

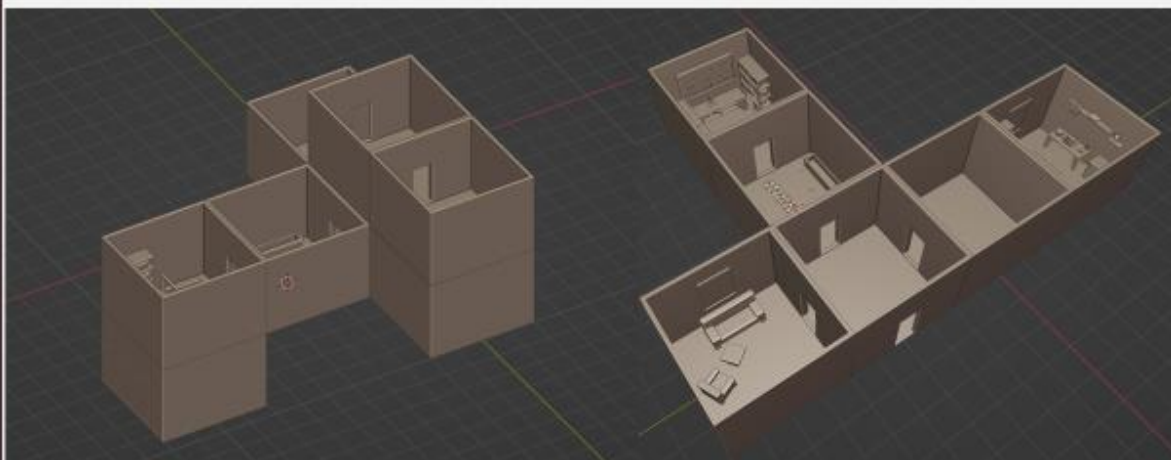


Сюжет игры

- По сюжету главный герой мальчик неожиданно попадает в необычную ситуацию. Его отец малоизвестный, но гениальный ученый работал в своей лаборатории над очередным изобретением, которое имело огромный потенциал. В результате экспериментов что-то пошло не так. Сын услышал странные звуки из своего окна, а следом во всем доме отключился свет. Ему предстоит разобраться в том, что произошло на самом деле, по пути решая головоломки и загадки, которые не только возникли из-за этого происшествия, но которые оставил отец на случай его непредвиденной пропажи.

Изм.	Лист	№ докум.	Подпись	Дата

Планирование уровней



Визуальный стиль



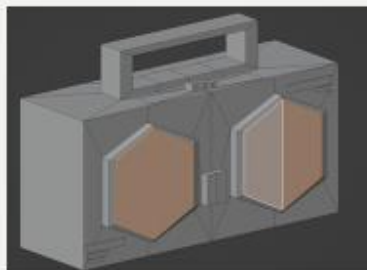
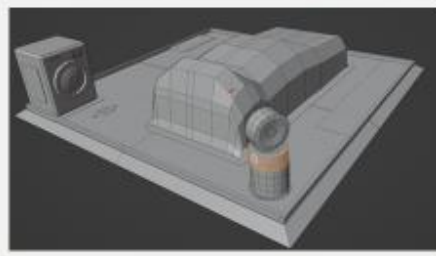
Изм.	Лист	№ докум.	Подпись	Дата

ПГУ1.090301.14.001 ПЗ

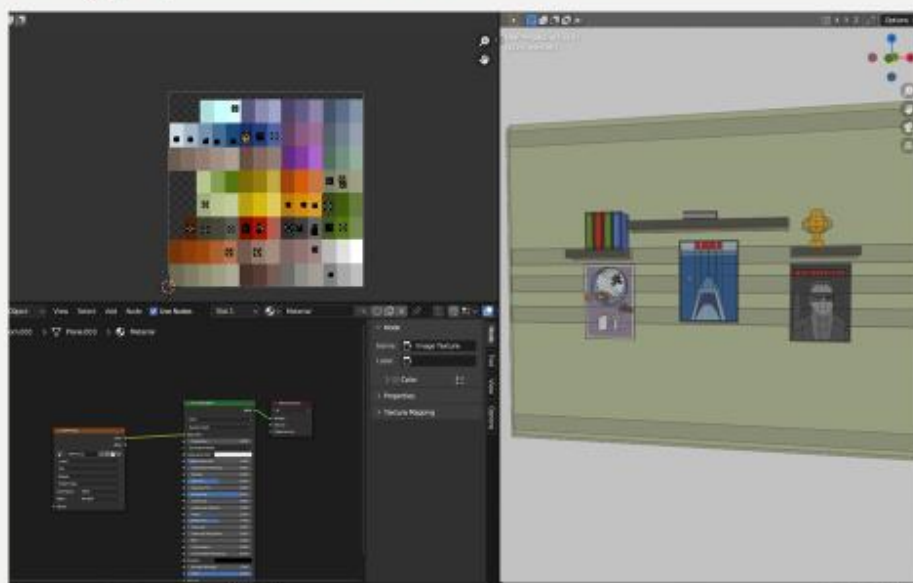
Лист

71

Моделирование объектов



Текстурирование объектов



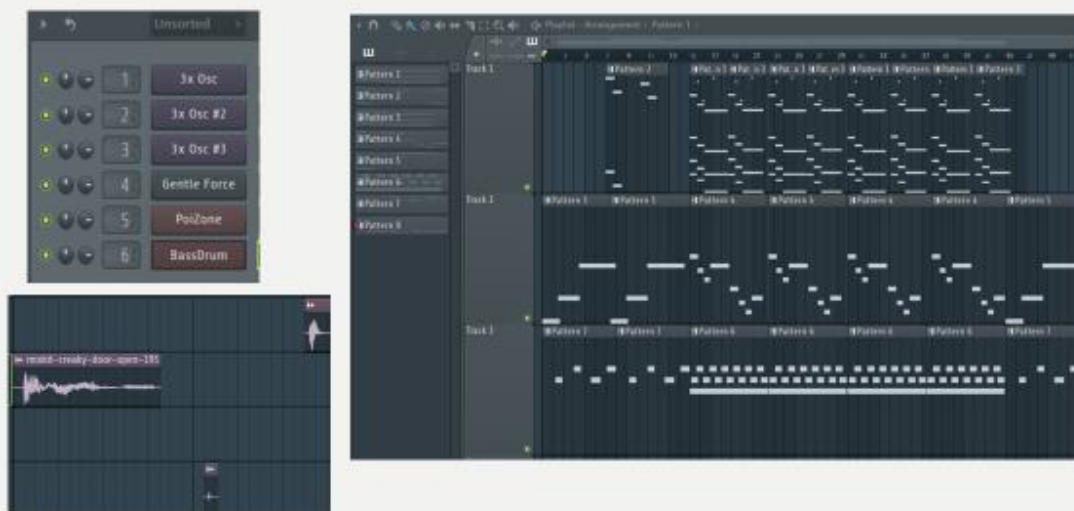
Изм.	Лист	№ докум.	Подпись	Дата

ПГУ1.090301.14.001 ПЗ

Лист

72

Музыкальное сопровождение игры



Геймдизайн



Изм.	Лист	№ докум.	Подпись	Дата

ПГУ1.090301.14.001 ПЗ

Лист

73

Пользовательский интерфейс



Демонстрация геймплея



Изм.	Лист	№ докум.	Подпись	Дата

ПГУ1.090301.14.001 ПЗ

Лист

74

Спасибо за внимание!

					<i>ПГУ1.090301.14.001 ПЗ</i>	Лист
						75
Изм.	Лист	№ докум.	Подпись	Дата		