

Пермский филиал федерального государственного автономного
образовательного учреждения высшего образования
«Национальный исследовательский университет
«Высшая школа экономики»

Факультет экономики, менеджмента и бизнес-информатики

Чепокhov Елизар Сергеевич

**РЕАЛИЗАЦИЯ МНОГОПОЛЬЗОВАТЕЛЬСКОЙ КОМПЬЮТЕРНОЙ ИГРЫ
В ЖАНРЕ РПГ**

Курсовая работа

студента образовательной программы бакалавриата «Программная инженерия»
по направлению подготовки 09.03.04 Программная инженерия

Руководитель:
преподаватель кафедры
информационных
технологий

В. В. Лебедев

Пермь, 2021 год

Аннотация

Название: Реализация многопользовательской компьютерной игры в жанре РПГ

Автор: Чепокhov Елизар Сергеевич, студент третьего курса образовательной программы «Программная инженерия».

Руководитель: Лебедев Виктор Валерьевич, преподаватель кафедры информационных технологий.

Данная курсовая работа посвящена разработке многопользовательской компьютерной игры.

Работа включает 78 страниц формата А4, из них в основной части 37 страниц.

Основная часть работы включает в себя 21 иллюстрацию и 13 таблиц.

Библиографический список состоит из 14 публикаций.

Работа включает в себя 6 приложений.

Оглавление

Оглавление	3
Введение	4
Глава 1 Анализ.....	6
1.1 Изучение особенностей игр в жанре «RPG»	6
1.2 Изучение особенностей строения игровых уровней	8
1.3 Обзор и сравнение программного обеспечения для создания игр	9
1.4 Обзор игр-аналогов.....	12
1.5 Требования к разрабатываемой игре	15
1.6 Выводы по главе	17
Глава 2 Проектирование	18
2.1 Проектирование алгоритмов.....	18
2.1.1 Модель вариантов использования	18
2.1.2 Диаграммы классов.....	25
2.2 Проектирование графического интерфейса	26
2.3 Выводы по главе	29
Глава 3 Разработка и тестирование.	30
3.1 Обоснование выбора средств разработки	30
3.2 Описание используемых функций.	31
3.3 Тестирование программного продукта.....	32
3.4 Выводы по главе	33
Заключение.....	34
Список сокращений и условных обозначений	35
Библиографический список.....	36
ПРИЛОЖЕНИЕ А	38
ПРИЛОЖЕНИЕ Б	40
ПРИЛОЖЕНИЕ В.....	41
ПРИЛОЖЕНИЕ Г	59
ПРИЛОЖЕНИЕ Д.....	73
ПРИЛОЖЕНИЕ Е	77

Введение

В настоящее время игры пользуются мировой популярностью и дают человеку расслабиться и получить новые эмоции, так же во время пандемии игры стали одним из основных времяпрепровождений с друзьями. За 2020 год количество людей, которые начали играть в игры увеличилось в 1,5 раза и преодолело отметку в 3 миллиарда человек, а рыночная стоимость игровой индустрии увеличилась на 20% (до 174,9 миллиарда долларов) по сравнению с 2019 годом. Во время пандемии игры стали неотъемлемой частью досуга для 40% населения земли и количество новых пользователей будет расти с каждым годом. Особой популярностью пользуются простые многопользовательские игры, так как они не сильно нагружают мозг человека, а также позволяют проводить совместное время с друзьями, когда нет возможности встретиться в живую.

RPG (Role play game) – игры в этом жанре делают акцент на обыгрывание каждым игроком своей роли, что позволяет сильнее погрузиться в игру и представить себя в качестве персонажа игры. Неотъемлемой частью игрового процесса в данном жанре является персонализация персонажа и выбор одной или нескольких ролей в соответствии с предпочтениями игрока. Из-за чего спрос на игры в подобном жанре с каждым годом растет, ведь каждый игрок стремится получить опыт, который не может получить в реальной жизни.[13]

Актуальность данной темы состоит в том, что игры подобного жанра пользуются огромным спросом, особенно во время и после пандемии, а также в том, что игровой рынок не может предоставить достаточного количество простых, по стилистике игр, которые бы были посты в освоении и визуальной стилистики игры. Именно поэтому разработка игры в жанре «RPG» актуальная тема.

Объектом данного исследования являются компьютерные игры в жанре «RPG». **Предметом** исследования является разработка алгоритма многопользовательской игры.

Целью данной работы является разработка компьютерной игры в жанре «RPG» с наличием многопользовательской игры.

Задачи, для достижения поставленной цели:

1. Провести анализ действующих алгоритмов.
2. Проанализировать, сравнить и выбрать визуализатор для игры.

3. Изучить способы разработки многопользовательских игр.
4. Спроектировать алгоритм многопользовательской игры.
5. Спроектировать интерфейс для игры.
6. Провести опрос и проанализировать ответы по поводу интерфейса, внести изменения.
7. Реализовать код программы и предварительный интерфейс.
8. Реализовать проект с использованием алгоритма многопользовательской игры.
9. Написать пользовательскую документацию для программы.
10. Провести тестирование, для выявления недочетов и сбора критики тестировщиков.
11. Внести изменения на основании выявленных недочетов на этапе тестирования.

Работа состоит из трех глав:

1. В первой главе будет проведен анализ предметной области, исследование особенностей строения игры и действующих алгоритмов, а также обзор программного обеспечения для написания игр и обзор игр-аналогов. Результатом первой главы являются поставленные требования к написанию работы.
2. Во второй главе будет представлено проектирование будущей игры, включающее в себя проектирование интерфейса и алгоритмов. Результатом главы станут спроектированные алгоритмы и интерфейс игры.
3. В третьей главе будет показана разработка игры и ее алгоритмов и интерфейса, а также представлены результаты тестирования. Результатом главы станет готовый продукт.

Практической значимостью данной работы является расширение знаний в области разработки компьютерных игр и разработки алгоритмов для многопользовательских игр, а также создание многопользовательской игры с возможной последующей публикацией на сервисах дистрибуции игр.

По итогу после завершения данной работы будет разработана многопользовательская игра в жанре «RPG».

Глава 1 Анализ

В данной главе будут рассмотрены особенности игр в жанре «RPG». Будет выбран визуализатор для игры на основании анализа используемых визуализаторов для игр, выявления их преимуществ, недостатков и особенностей. Так же будут рассмотрены и проанализированы алгоритмы многопользовательских игр, выявлены их недостатки и преимущества. Результатом первой главы являются поставленные требования к разрабатываемой системе.

1.1 Изучение особенностей игр в жанре «RPG»

Компьютерная ролевая игра, обозначаемая аббревиатурой «RPG» - жанр компьютерных игр, базирующийся на элементах игрового процесса традиционных настольных ролевых игр. В игре предоставленного жанра игрок управляет одним или несколькими персонажами, любой из которых описан набором численных характеристик, списком способностей и умений таких как очки здоровья, показатели силы, ловкости и т. п.[9]

Из признаков данного жанра выделяют следующие:

1. Ролевая система
2. Исследование
3. Сюжет
4. Боевая система

Ролевая система – описывает способы создания и развития персонажей[9]. Для внедрения данной системы требуется реализовать необходимые элементы, из которых:

1. Управление одним или несколькими уникальными персонажами;
2. Улучшение внутриигровых значений персонажа пользователя;
3. Пользователь может создавать и улучшать своего персонажа;
4. Пользователь должен планировать развитие персонажа;
5. Основным способом взаимодействия с игровым миром является применение способностей персонажа.

Исследование – описывает способы перемещения персонажа по игровому миру и взаимодействия с ним через нахождение игровых областей, предметы или объекты игрового мира. Для данной системы выделяют следующие элементы:

1. Персонаж пользователя может взаимодействовать с игровым миром, находить новые игровые области, находить предметы и источники информации;
2. Пользователь может выбрать вариант прохождения из нескольких предложенных;
3. Персонаж может воздействовать на игровой мир;
4. Игровой мир может воздействовать на персонажа;
5. Игровые области должны открываться в зависимости от развития персонажа;

Сюжет – описывает элементы повествования, игровой мир, персонажей и способы взаимодействия этих компонентов[9]. Для данной системы стоит выделить следующие элементы:

1. Персонаж пользователя может получать информацию из информационных источников игры с помощью исследования игрового мира;
2. Персонаж пользователя может выполнять задания и продвигаться по цепочке связанных событий;
3. Продвижение по сюжету требует от игрока обдумывания;
4. Сюжет зависит от решений пользователя;
5. В процессе взаимодействия пользователь может сделать выбор;

Боевая система – объясняет влияние ролевой системы, исследований и сюжета на результаты сражения. В сражениях должен присутствовать элемент случайности, а эффективность в сражении зависит от навыков персонажа.

Таким образом данный жанр компьютерных игр представляет огромную вариативность в создании проектов и является важным как для пользователя, так и для разработчика[13].

Из достоинств жанра, главным достоинством является возможность создавать персонализированного персонажа пользователем, что позволяет глубже погрузиться в процесс игры и проводить в ней больше времени. Так же стоит выделить вариативность игрового пространства, которая предоставляет возможность создавать игровые уровни с разным визуальным стилем и разным построением.

Недостатком жанра «RPG» является создание игрового пространства, для размещения в нем сюжета игры, так как на разработку таких пространств уходит большое количество времени[13].

1.2 Изучение особенностей строения игровых уровней

Построение карт в играх не является строгим. В некоторых играх количество уровней и обширность игрового мира может составлять тысячи или десятки тысяч локаций. Игровое пространство может организовываться по-разному, из основных видов построения выделяют:

1. Текстовая графика (см. Рисунок 1.1).

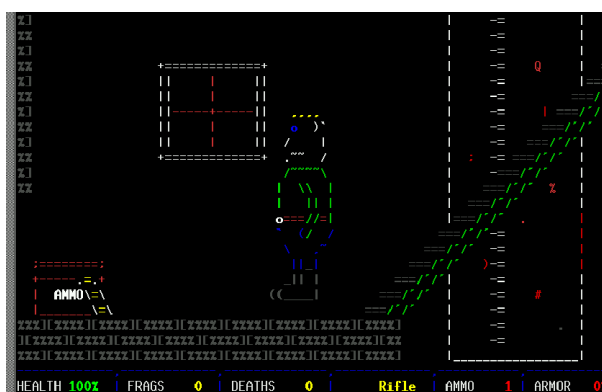


Рисунок 1.1 – Пример текстовой графики

2. Двумерная графика (см. Рисунок 1.2).



Рисунок 1.2. Пример двухмерной графики

3. Изометрическая графика (см. Рисунок 1.3).

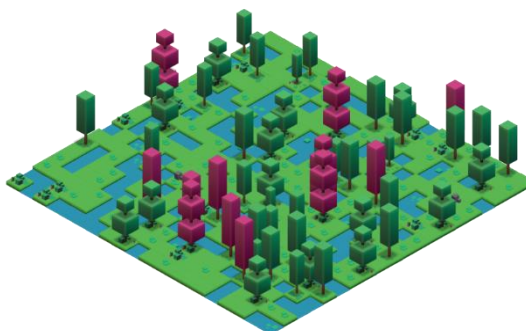


Рисунок 1.3. Пример изометрической графики

4. Трехмерная графика (см. Рисунок 1.4).

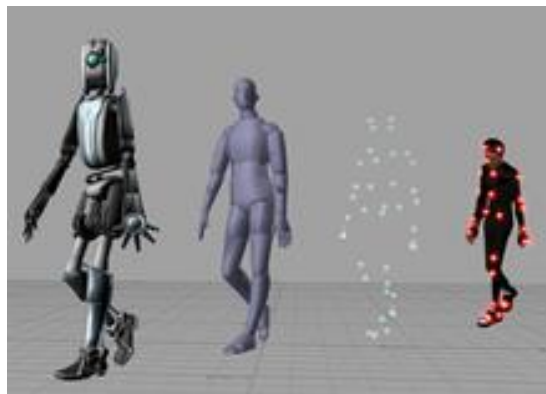


Рисунок 1.4. Пример трехмерной графики

Для создания игрового поля различают процедурную и фиксированную генерацию. Для процедурной генерации свойственно автоматическое создание и возведение игрового пространства с использованием алгоритмов, этот способ увеличивает реиграбельность каждой новой игры, но приводит к усложнению игрового процесса, шанс выигрыша при подобной генерации значительно уменьшается, а сложность игры увеличивается. Фиксированная генерация используется чаще, так как при таком варианте игру возможно спроектировать так, чтобы гарантировать её прохождение.[10]

При наличии многопользовательской игры фиксированная генерация преобладает, так как процедурная генерация может облегчить игровой процесс для некоторых пользователей, что будет являться нечестным по отношению к остальным пользователям. Так же процедурная генерация может привести к появлению ошибок в генерации игровых пространств, что может повлечь за собой застревание или выход за область игрового пространства или недоступность прохождения игрового пространства.[7]

1.3 Обзор и сравнение программного обеспечения для создания игр

Для создания и отображения игр требуется визуализатор, чтобы отобразить код программы в изображение. В качестве визуализатора для несложных и не затратных игр можно применить и обычные визуализаторы языка, к примеру, Visual Studio. Но для реализации более комплексных алгоритмов приходится сильнее углубляться в изучение языка и основы работы различных классов. Таким образом даже имея высокую квалификацию разработчик программного обеспечения имеет возможность легко допустить трудноуловимые ошибки, некоторые из которых имеют все шансы оказаться фатальными.[3]

Ввиду этого были разработаны программы для создания игр, которые не только упрощают разработку игры и избавляют от большого количества ошибок, но и привносят что-то новое, чего невозможно добиться обычными визуализаторами. Это программное обеспечение, или же иначе «game engine», в данный момент применяется практически во всех играх. Из ведущих применяемых программных обеспечений стоит отметить следующие: Unity и Unreal Engine.

Unity - это мультиплатформенная среда для разработки 2D и 3D игр и интерактивного контента. Первый выпуск состоялся в 2005 году и с того времени постоянно обновлялся, вводя что-то новое и исправляя старое. Основные преимущества Unity это наличие визуальной среды разработки, обширная база данных, межплатформенная поддержка и абсолютная бесплатность для начинающих разработчиков[4]. Данная технология наиболее подходит для создания игрового пространства и интерактивного интерфейса компьютерной игры. Основным языком программирования – C#, так же поддерживается и JavaScript. На рисунке 1.5 показан интерфейс программы.

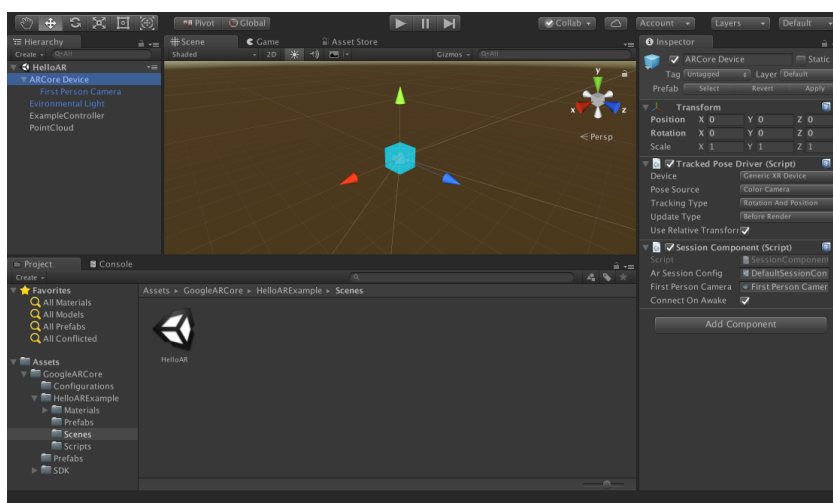


Рисунок 1.5 – Интерфейс Unity

Unreal Engine – среда для разработки компьютерных игр, разработанная в 1998 году и являющаяся собственностью компании «Epic Games». Последняя версия – Unreal Engine 4, вышедшая в 2014 году, позволяет создавать приложения на десяти операционных системах. Из преимуществ выделяют инструмент для создания уровней, интегрированный в программу и не имеющий аналогов и поддержка различных систем рендеринга. С 2015 года стала бесплатной, для всех разработчиков. Основным языком программирования – C++. В основном направлена на создание 3D игр и фильмов. Интерфейс программы представлен на рисунке 1.6.

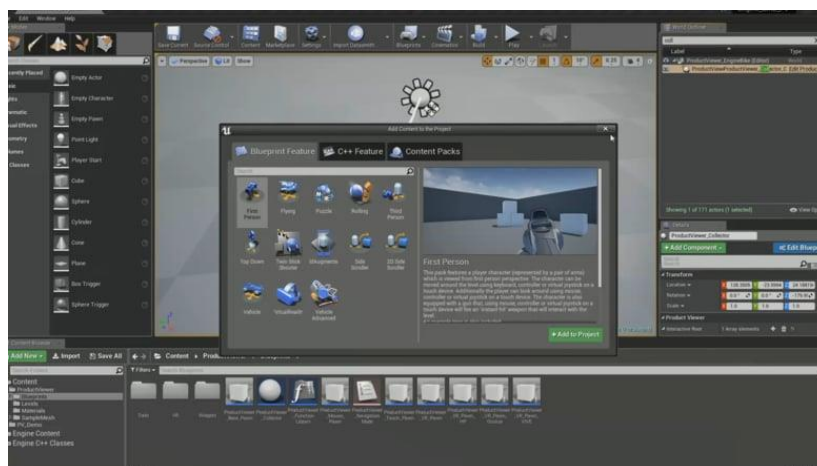


Рисунок 1.6 – Интерфейс Unreal Engine

Для того чтобы выбрать программное обеспечение для написания системы игры, нужно сравнить представленные выше программы по критериям. В таблице 1.1 выявляем основные требования к среде в которой будем создавать игру. На основе имеющихся данных можно определить какая среда для написания игр подходит лучше всего для создаваемого программного продукта.

Таблица 1.1 - Критерии сравнения среды для написания игры

Среда	Поддерживаемые форматы	Порог вхождения	Основной язык программирования	Наличие документации	Стоимость лицензии
1. Unity	2D, 3D	Низкий	C#	да	Бесплатно
2. Unreal Engine	2D, 3D	Высокий	C++	да	Бесплатно

На основании сравнения, удобными в использовании является Unreal Engine. Главными же факторами, для начинающего разработчика являются наличие документации и бесплатная лицензия. Таким образом лучшим вариантом для создания тестового проекта служит программное обеспечение Unity из-за большого количества документации, бесплатной лицензии и низкого порога вхождения. При модернизации проекта для публикации стоит перенести игру на язык C++, так как он является более энергоемким и ресурсоемким, из-за чего показатели быстродействия увеличатся.

1.4 Обзор игр-аналогов

Для выявления преимуществ разрабатываемой игры необходимо провести анализ игр-аналогов и выявить их преимущества и недостатки.

Так как жанр «RPG» игр является одним из самых популярных, то в рамках данной курсовой работы будут рассмотрены более приближенные игры-аналоги. Из аналогов стоит выделить следующие:

1. Enter the Gungeon;
2. Darkest Dungeon;
3. World of Warcraft;
4. Genshin Impact;

Enter the Gungeon – компьютерная игра в жанрах «RPG» и «Roguelike» с видом сверху, выпущенная 5 апреля 2016. Игра написана на движке Unity. Из преимуществ стоит выделить возможность запуска игры на слабых компьютерах, а так же захватывающий игровой процесс и интерфейс игры. К минусам относятся отсутствие полноценной многопользовательской игры, отсутствие сюжета и отсутствие полноценной персонализации персонажа, так же игра является платной. Интерфейс игры представлен на рисунке 1.7.



Рисунок 1.7 – Интерфейс Enter the Gungeon

Darkest Dungeon – компьютерная ролевая игра с roguelike-элементами, разработанная и выпущенная независимой канадской студией Red Hook Studios 19 января 2016 года. Игра написана на движке Unity. К плюсам данной игры стоит отнести полное соответствие жанру «RPG» и возможность запускать игру в оффлайн режиме. Из минусов стоит выделить отсутствие многопользовательской игры, малая

вариативность игрового процесса и отсутствие персонализации, так же игра является платной. Интерфейс игры представлен на рисунке 1.8.



Рисунок 1.8 – Интерфейс Darkest Dungeon

World of Warcraft – глобальная многопользовательская ролевая онлайн-игра, созданная и издаваемая компанией Blizzard Entertainment с 23 ноября 2004 года. К плюсам данной игры стоит отнести возможность многопользовательской игры и обширную персонализацию персонажа. Из минусов преобладают отсутствие возможности запуска на слабых компьютерах, а также условная бесплатность игры. Интерфейс игры представлен на рисунке 1.9.



Рисунок 1.9 – Интерфейс World of Warcraft

Genshin Impact – компьютерная игра в жанре Action-adventure с открытым миром и элементами RPG, разработанная китайской компанией miHoYo Limited в 2020. Игра была создана на движке Unity. Из плюсов стоит выделить возможность бесплатной игры, возможность запуска на слабых компьютерах и мобильных

устройствах, а также проработанный сюжет. К минусам относятся урезанный вариант многопользовательской игры и отсутствие полноценной персонализации персонажа. Интерфейс игры представлен на рисунке 1.10.



Рисунок 1.10 – Интерфейс Genshin Impact

Выделим основные критерии для сравнения аналогов и разрабатываемой игры (см. Таблица 1.2):

- Присутствует возможность играть онлайн через интернет?
- Присутствует возможность запускать игру в оффлайн режиме?
- Присутствует возможность многопользовательской игры?
- Присутствует возможность персонализации персонажа?
- Присутствует сюжет в игре?
- Присутствует возможность запуска на мобильных устройствах?
- Присутствует возможность запуска на слабых компьютерах?
- Присутствуют варианты игры без продвижения по сюжету (миниигры)?
- Есть возможность бесплатной игры?

Таблица 1.2 - Критерии сравнения среды для написания игры

Среда	Enter the Gungeon	Darkest Dungeon	World of Warcraft	Genshin Impact	Разрабатываемая игра
Возможность играть онлайн через интернет	-	-	+	+	+
Возможность запускать игру в оффлайн режиме	+	+	-	-	+
Возможность многопользовательской игры	+	-	+	+	+

Возможность персонализации персонажа	+	-	+	+	+
Наличие сюжета в игре	-	+	+	+	+
Возможность запуска на мобильных устройствах	-	-	-	+	+
Возможность запуска на слабых компьютерах	+	+	-	-	+
Игры без продвижения по сюжету	-	-	+	+	+
Возможность бесплатной игры	-	-	+	+	+

По полученным результатам можно сделать вывод, что разрабатываемая игра будет лучше аналогов из-за возможности запуска на мобильных устройствах, возможности бесплатной игры, возможности запуска как в онлайн, так и в оффлайн режиме, а также из-за возможности многопользовательской игры по сети.

Таким образом стоит выделить перечень требований, заимствованных у аналогов:

1. Возможность запускать игру в оффлайн и онлайн режиме;
2. Возможность многопользовательской игры;
3. Возможность персонализации персонажа;
4. Возможность запуска на мобильных устройствах и слабых компьютерах;
5. Наличие сюжета в игре;
6. Возможность бесплатной игры;
7. Наличие игр без продвижения по сюжету;

1.5 Требования к разрабатываемой игре

Для разрабатываемой системы были определены функциональные и нефункциональные требования:

Функциональные требования:

1. Система должна загружать выбранные пользователем игровые пространства;
2. Система должна предоставлять возможность передвижения пользователя по игровому пространству;

3. Система должна взаимодействовать с пользователем и откликаться на его действия;
4. Система должна предоставлять пользователю возможность подключения к сети интернет;
5. Система должна оповещать пользователя при возникновении ошибок;
6. Система должна давать пользователю возможность персонализации персонажа;
7. Система должна предоставлять возможность выхода из системы;
8. Уровни игры должны быть спроектированы с помощью фиксированной генерации;

Нефункциональные требования:

1. Интерфейс игры должен быть простым для восприятия пользователя;
2. Архитектура системы должна позволять добавлять к нему дополнительные модули не нарушая работоспособность других модулей;
3. Система должна отвечать на запрос пользователя не более 5 секунд;
4. Функции ввода и вывода данных должны быть спроектированы таким образом, чтобы их удавалось повторно использовать на уровне объектного кода в других приложениях;
5. Архитектура игровых пространств должна позволять добавлять к нему новые игровые пространства не нарушая целостность других игровых пространств;
6. Система должна предоставлять интуитивность в использовании пользователем;
7. Система должна предоставлять безопасность данных пользователей;
8. Система должна своевременно обновляться;
9. Система должна давать пользователю возможность изменять настройки системы;
10. Система должна предоставлять возможность авторизации пользователя;
11. Интерфейс системы должен быть читабельным и понятным для восприятия;
12. Система должна давать пользователю возможность принимать решения в игровом пространстве;

13. Система должна предоставлять возможность запуска на слабых компьютерах;

1.6 Выводы по главе

В результате анализа игрового жанра и особенностей строения игровых уровней был выбран вариант визуализации игрового процесса на основании изучения особенности строения игровых уровней и приняты решения по внедрению механик игры подходящих жанру «RPG». Были выдвинуты функциональные требования, предъявляемые к системе. Проведен краткий обзор и сравнение основных движков, а также выбран движок для реализации игры. Рассмотрены плюсы и минусы аналогов и определены преимущества разрабатываемой игры.

Глава 2 Проектирование

В представленной главе спроектированы и рассмотрены методы и алгоритмы, которые потребуются для реализации игры и для удовлетворения требований, выявленных в ходе анализа. Результатом данной главы является спроектированная игра и её интерфейс.

2.1 Проектирование алгоритмов

2.1.1 Модель вариантов использования

Визуализация требований реализована с помощью use-case диаграммы (см. рисунок 2.1), созданная при помощи средств моделирования UML.[5]

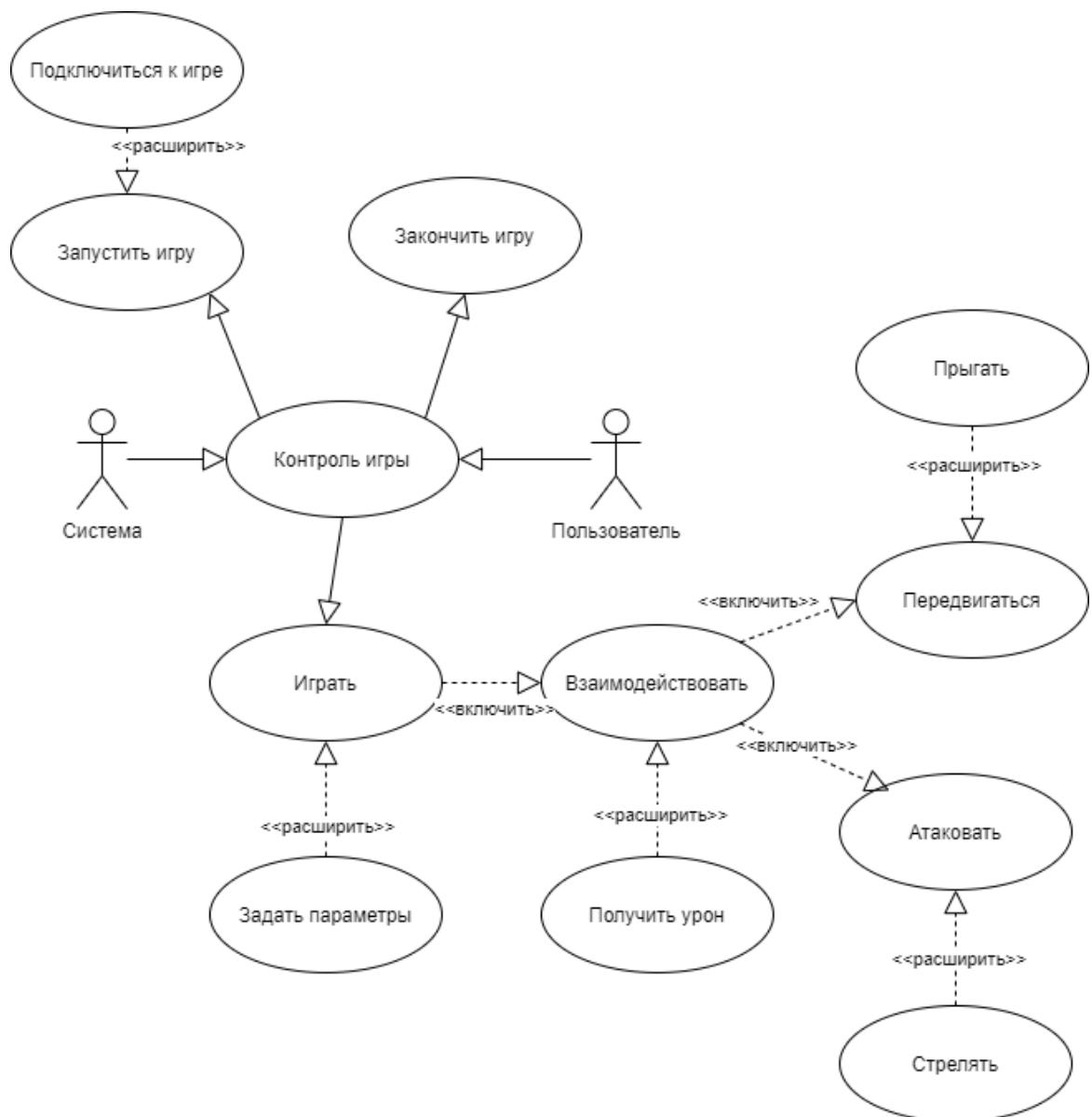


Рисунок 2.1 – Диаграмма прецедентов

Документирование прецедентов представлено ниже в таблицах (см. таблицы 2.1. – 2.11).

Таблица 2.1 – Прецедент «Контроль игры»

Идентификатор и название	УС-1. Контроль игры.		
Автор	Чепокоев Елизар	Дата	02.02.2021
Акторы	Пользователь. Система.		
Описание	Прецедент позволяет взаимодействовать пользователю и системе в игровом процессе.		
Триггер	Пользователь запускает игру.		
Предварительные условия	PRE-1. Наличие подключения к сети интернет. PRE-2. Наличие на диске файла с сохраненной конфигурацией игры.		
Выходные условия	POST-1. Игра запущена. POST-2. Пользователь отправляет команды системе.		
Основной поток	1.0 Запуск игры: 1. Выполняется прецедент «Запустить игру» (см 1.0.E1 и 1.0.E2)		
Альтернативные потоки	-		
Исключения	1.0.E1. Отсутствует подключение к сети интернет. 1. Система выводит сообщение о том, что отсутствует подключение к сети интернет. 2. Система возвращается к главному экрану. 1.0.E2. Файлы игры повреждены. 1. Система выводит сообщение о том, что файлы игры повреждены. 2. Система возвращается к главному экрану.		
Приоритет	Высокий		
Точка расширения	УС-2. Запустить игру. УС-3. Закончить игру. УС-4. Играть.		

Таблица 2.2 – Прецедент «Запустить игру»

Идентификатор и название	УС-2. Запустить игру.		
Автор	Чепокоев Елизар	Дата	02.02.2021
Акторы	Пользователь. Система.		
Описание	Прецедент позволяет пользователю запустить игру в онлайн или оффлайн режиме.		
Триггер	Пользователь запускает игру.		
Предварительные условия	PRE-1. Наличие подключения к сети интернет. PRE-2. Наличие на диске файла с сохраненной конфигурацией игры.		
Выходные условия	POST-1. Игра запущена.		
Основной поток	1.0 Запуск игры: 2. Система показывает главный экран. 3. Пользователь нажимает на кнопку «Играть онлайн». (см. 1.1) 4. Система выводит окно загрузки. 5. Система проверяет подключение к сети интернет. (см. 1.0.E1) 6. Выполняется прецедент «Подключиться к игре».		
Альтернативные потоки	1.1 Пользователь выбрал оффлайн игру: 1. Пользователь нажимает на кнопку «Играть оффлайн». 2. Система выводит окно загрузки. 3. Система проверяет наличие файла сохранения. (см. 1.1.E1) 4. Система восстанавливает конфигурацию и состояние из файла и загружает игровое поле.		

Исключения	1.0.E1. Отсутствует подключение к сети интернет. 3. Система выводит сообщение о том, что отсутствует подключение к сети интернет. 4. Система возвращается к главному экрану. 1.1.E1. Файл сохранения поврежден. 1. Система выводит сообщение о том, что файл поврежден. 2. Система возвращается к главному экрану.
Приоритет	Высокий
Точка расширения	УС-6. Подключиться к игре.

Таблица 2.3 – Прецедент «Закончить игру»

Идентификатор и название	УС-3. Закончить игру.		
Автор	Чепокоев Елизар	Дата	02.02.2021
Акторы	Пользователь. Система.		
Описание	Прецедент позволяет выйти из системы.		
Триггер	Пользователь нажимает на кнопку «Выйти».		
Предварительные условия	-		
Выходные условия	POST-1. Игра завершена.		
Основной поток	1.0 В игре: 1. Игрок нажимает кнопку «Выйти в меню». 2. Система запрашивает подтверждение выхода «Да\Нет». 3. Система выходит в главное меню. 2.0 В главном меню: 1. Игрок нажимает кнопку «Выйти». 2. Система запрашивает подтверждение выхода «Да\Нет». 3. Система завершает свое выполнение и выходит на рабочий стол.		
Альтернативные потоки	-		
Исключения	-		
Приоритет	Средний		
Точка расширения	-		

Таблица 2.4 – Прецедент «Играть»

Идентификатор и название	УС-4. Играть.		
Автор	Чепокоев Елизар	Дата	02.02.2021
Акторы	Пользователь. Система.		
Описание	Прецедент представляет игровой процесс.		
Триггер	Пользователь запускает игру.		
Предварительные условия	PRE-1. Выполнен прецедент «Запустить игру».		
Выходные условия	POST-1. Пользователь играет в игру.		
Основной поток	1.0 Начало игры: 1. Система запускает основное игровое пространство. 2. Система позволяет пользователю выбирать следующие действия. (см 1.1, 1.2) 3. Система позволяет пользователю передвигаться и взаимодействовать с игровым пространством. 4. Пользователь передвигается к месту сражения. 5. Пользователь выбирает оружие для сражения. 2.0 Фаза выбора оружия: 1. Пользователь выбирает вид оружия (ближнее или дальнее). 3.0 Фаза сражения: 1. Пользователь атакует противника в ближнем бою. (см 3.1)		

	2. Пользователь уклоняется от ударов противника. 3. Пользователь выигрывает в бою (см 3.2) 4.0 Завершение боя: 1. Пользователь выигрывает сражение. 2. Система вознаграждает пользователя очками опыта и игровой валютой. 3. Система отправляет пользователя в основное игровое пространство. 4. Система позволяет пользователю выбирать следующие действия. (см 1.1, 1.2)
Альтернативные потоки	1.1 Пользователь нажал кнопку «Выйти»: 1. Выполняется прецедент «Закончить игру» 1.2 Пользователь открыл настройки игры: 1. Выполняется прецедент «задать параметры» 3.1 Пользователь выбрал дальнейшее оружие: 1. Пользователь атакует противника в дальнем бою. 3.2 Пользователь проигрывает в бою: 1. Система отправляет пользователя в основное игровое пространство.
Исключения	-
Приоритет	Высокий
Точка расширения	УС-5. Задать параметры.

Таблица 2.5 – Прецедент «Задать параметры»

Идентификатор и название	УС-5. Задать параметры.		
Автор	Чепокоев Елизар	Дата	02.02.2021
Акторы	Пользователь.		
Описание	Прецедент позволяет пользователю задать параметры игры		
Триггер	Пользователь нажимает на кнопку «Параметры» или на клавишу «Esc».		
Предварительные условия	PRE-1. Нажата кнопка «Параметры» или клавиша «Esc».		
Выходные условия	POST-1. Параметры сохранены. POST-2. Игра настроена в соответствии с параметрами.		
Основной поток	1.0 Изменение параметров: 1. Система открывает окно настроек игры. 2. Пользователь вносит изменения в параметры игры.		
Альтернативные потоки	-		
Исключения	-		
Приоритет	Средний		
Точка расширения	-		

Таблица 2.6 – Прецедент «Подключиться к игре»

Идентификатор и название	УС-6. Подключиться к игре.		
Автор	Чепокоев Елизар	Дата	02.02.2021
Акторы	Пользователь. Система.		
Описание	Прецедент подключает пользователя к игре.		
Триггер	Пользователь нажимает на кнопку «Играть».		
Предварительные условия	PRE-1. Выполнен прецедент «Запустить игру».		
Выходные условия	POST-1. Пользователь подключился к игре.		
Основной поток	1.0 Выполняется подключение к игре: 1. Система считывает файл с конфигурацией игры. 2. Система запрашивает данные с сервера с данными о игровом пространстве.		

	3. Система получает данные с сервера. 4. Система загружает игровое пространство пользователя. (см. 1.1)
Альтернативные потоки	1.1 Подключение к чужой игре: 1. Система загружает игровое пространство другого пользователя основываясь на данных с сервера.
Исключения	-
Приоритет	Высокий
Точка расширения	-

Таблица 2.7 – Прецедент «Взаимодействовать»

Идентификатор и название	УС-7. Взаимодействовать.		
Автор	Чепокоев Елизар	Дата	02.02.2021
Акторы	Пользователь.		
Описание	Прецедент позволяет пользователю взаимодействовать с игровым миром.		
Триггер	Пользователь нажимает на кнопки «W», «A», «S», «D», «Пробел» или «ЛКМ» или игровой мир взаимодействует с пользователем.		
Предварительные условия	PRE-1. Запущен прецедент «Играть».		
Выходные условия	POST-1. Пользователь взаимодействует с игровым миром.		
Основной поток	1.0 Передвижение персонажа: 1. Пользователь нажал кнопку «W», «A», «S», «D». 2. Система производит анимацию передвижения. 3. Система выполняет прецедент «Передвигаться». 2.0 Прыжок персонажа: 1. Пользователь нажал кнопку «Пробел». 2. Система выполняет прецедент «Прыгать». 3.0 Атака персонажа: 1. Пользователь нажал кнопку «ЛКМ». 2. Система выполняет прецедент «Атаковать». 4.0 Пользователь получает урон: 1. Система выполняет прецедент «Получить урон». 2. Система выполняет прецедент «Атаковать» 3. Система производит анимацию получения урона пользователем. (см. 4.1)		
Альтернативные потоки	4.1 Пользователь выполнил прецедент «Передвигаться»: 1. Пользователь уклоняется от получения урона.		
Исключения	-		
Приоритет	Высокий		
Точка расширения	УС-8. Получить урон.		

Таблица 2.8 – Прецедент «Получить урон»

Идентификатор и название	УС-8. Получить урон.		
Автор	Чепокоев Елизар	Дата	02.02.2021
Акторы	Пользователь.		
Описание	Прецедент позволяет системе атаковать пользователя и получать урон пользователем.		
Триггер	Система атакует пользователя.		
Предварительные условия	PRE-1. Запущен прецедент «Играть». PRE-2. Запущена фаза сражения.		
Выходные условия	POST-1. Пользователь получает урон и теряет очки здоровья.		
Основной поток	1.0 Получение урона: 1. Система атакует пользователя. 2. Система проверяет уклонился ли пользователь. 3. Система производит анимацию попадания.		

	<ol style="list-style-type: none"> 4. Система генерирует число урона основываясь на уровне пользователя и очках защиты пользователя. 5. Система вычитает нанесенный урон из очков здоровья пользователя. 6. Система проверяет очки здоровья пользователя (если >0). 7. Система продолжает фазу сражения прецедента «Играть».
Альтернативные потоки	<ol style="list-style-type: none"> 1.1 Очки здоровья меньше 0: <ol style="list-style-type: none"> 1. Система производит анимацию смерти персонажа. 2. Система оповещает о смерти персонажа. 3. Система загружает основное игровое пространство.
Исключения	-
Приоритет	Средний
Точка расширения	-

Таблица 2.9 – Прецедент «Передвигаться»

Идентификатор и название	УС-9. Передвигаться.		
Автор	Чепиков Елизар	Дата	02.02.2021
Акторы	Пользователь.		
Описание	Прецедент позволяет пользователю передвигаться в игровом мире.		
Триггер	Пользователь нажимает на кнопки «W», «A», «S», «D».		
Предварительные условия	PRE-1. Выполнен прецедент «Запустить игру». PRE-2. Запущен прецедент «Играть».		
Выходные условия	POST-1. Параметры сохранены. POST-2. Игра настроена в соответствии с параметрами.		
Основной поток	1.0 Передвижение персонажа: <ol style="list-style-type: none"> 1. Пользователь нажал кнопку «W». (см. 1.1, 1.2, 1.3) 2. Система производит анимацию передвижения. 3. Система передвигает персонажа вглубь. 		
Альтернативные потоки	1.1 Пользователь нажал на кнопку «A»: <ol style="list-style-type: none"> 1. Система производит анимацию передвижения. 2. Система передвигает персонажа в левее. 1.2 Пользователь нажал на кнопку «S»: <ol style="list-style-type: none"> 1. Система производит анимацию передвижения. 2. Система передвигает персонажа вблизи. 1.3 Пользователь нажал на кнопку «D»: <ol style="list-style-type: none"> 1. Система производит анимацию передвижения. 2. Система передвигает персонажа в правее. 		
Исключения	-		
Приоритет	Средний		
Точка расширения	УС-10. Прыгать.		

Таблица 2.10 – Прецедент «Прыгать»

Идентификатор и название	УС-10. Прыгать.		
Автор	Чепиков Елизар	Дата	02.02.2021
Акторы	Пользователь.		
Описание	Прецедент позволяет пользователю прыгать в игровом пространстве.		
Триггер	Пользователь нажимает на кнопку «Пробел».		
Предварительные условия	PRE-1. Выполнен прецедент «Запустить игру». PRE-2. Запущен прецедент «Играть».		
Выходные условия	POST-1. Персонаж прыгнул.		
Основной поток	1.0 Прыжок персонажа: <ol style="list-style-type: none"> 1. Пользователь нажал кнопку «Пробел». 2. Система выполняет анимацию прыжка. 		

Альтернативные потоки	-
Исключения	-
Приоритет	Средний
Точка расширения	-

Таблица 1.11 – Прецедент «Атаковать»

Идентификатор и название	UC-11. Атаковать.		
Автор	Чепокоев Елизар	Дата	02.02.2021
Актеры	Пользователь.		
Описание	Прецедент позволяет пользователю атаковать противников в игре.		
Триггер	Пользователь нажимает на кнопку «ЛКМ».		
Предварительные условия	PRE-1. Выполнен прецедент «Запустить игру». PRE-2. Запущен прецедент «Играть». PRE-3. Запущена фаза сражения.		
Выходные условия	POST-1. Пользователь атакует.		
Основной поток	1.0 Атака персонажа: 1. Пользователь нажал кнопку «ЛКМ». 2. Система проверяет выбор оружия пользователя. 3. Система выполняет анимацию атаки ближнего боя. (см. 1.1)		
Альтернативные потоки	1.1 Атака из дальнего оружия: 1. Система выполняет прецедент «Стрелять».		
Исключения	-		
Приоритет	Средний		
Точка расширения	UC-12. Стрелять.		

Таблица 1.12 – Прецедент «Стрелять»

Идентификатор и название	UC-12. Стрелять.		
Автор	Чепокоев Елизар	Дата	02.02.2021
Актеры	Пользователь.		
Описание	Прецедент позволяет пользователю атаковать противников в игре.		
Триггер	Пользователь нажимает на кнопку «ЛКМ».		
Предварительные условия	PRE-1. Выполнен прецедент «Запустить игру». PRE-2. Запущен прецедент «Играть». PRE-3. Запущена фаза сражения.		
Выходные условия	POST-1. Пользователь атакует.		
Основной поток	1.0 Атака персонажа: 1. Пользователь нажал кнопку «ЛКМ». 2. Система выполняет анимацию атаки дальнего боя.		
Альтернативные потоки	-		
Исключения	-		
Приоритет	Средний		
Точка расширения	-		

2.1.2 Диаграммы классов

Для демонстрации игровой логики были созданы диаграммы классов. Диаграмма, изображенная на рисунке 2.2 описывает игровую логику при инициализации игры и отображение игровых объектов при загрузке игрового мира.

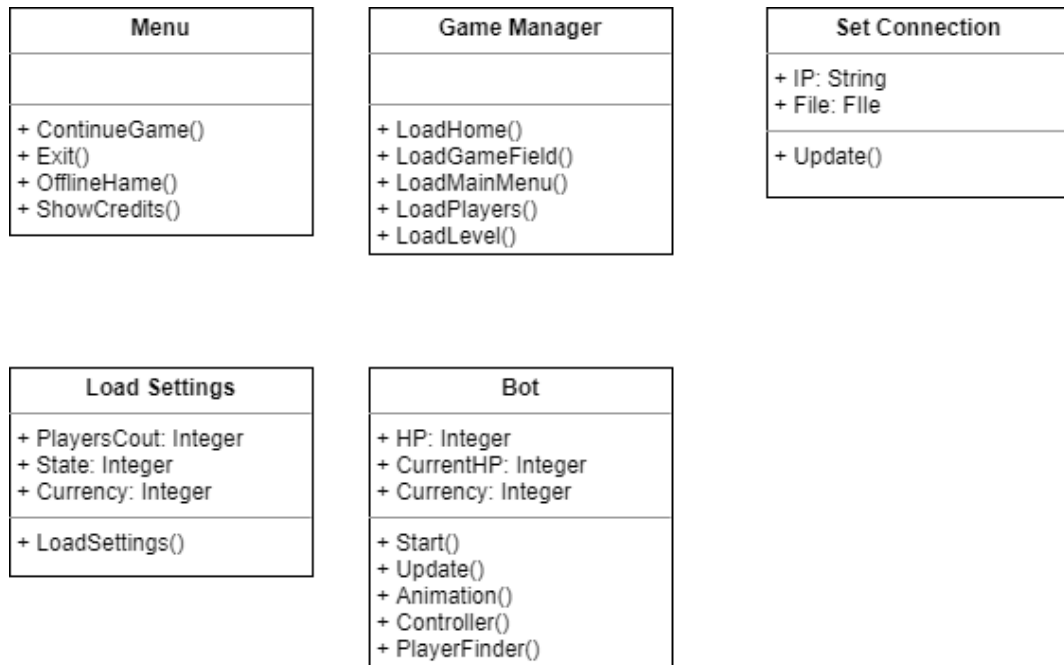


Рисунок 2.2 – Диаграмма классов при инициализации

На рисунке 2.3 представлена диаграмма классов, описывающая игровую логику.

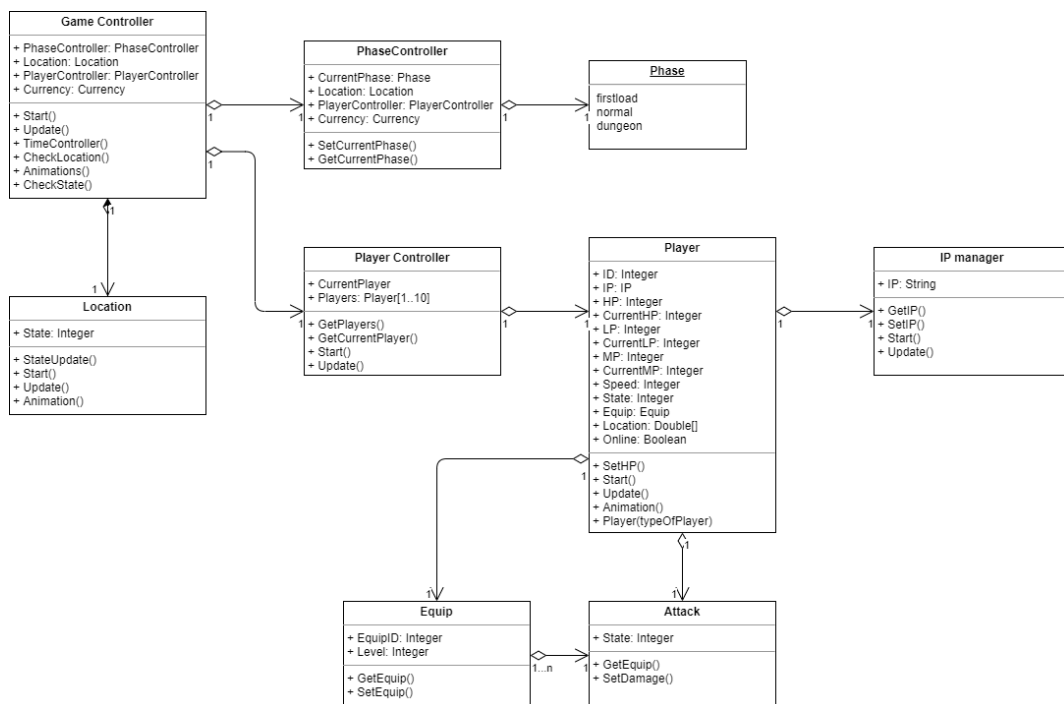


Рисунок 2.3 – Диаграмма классов

Данные классы связаны классом Game Controller отвечающим за игровой процесс и запускающим события остальных классов системы.

2.2 Проектирование графического интерфейса

Для проектирования графического интерфейса требуется визуализировать основные алгоритмы работы интерфейса для последующей адаптации данных алгоритмов с помощью инструментов разработки Unity.

На рисунке 2.4 изображен алгоритм главного меню, которое будет запущено при запуске игры. На данном алгоритме представлен процесс выбора следующих действий пользователем.

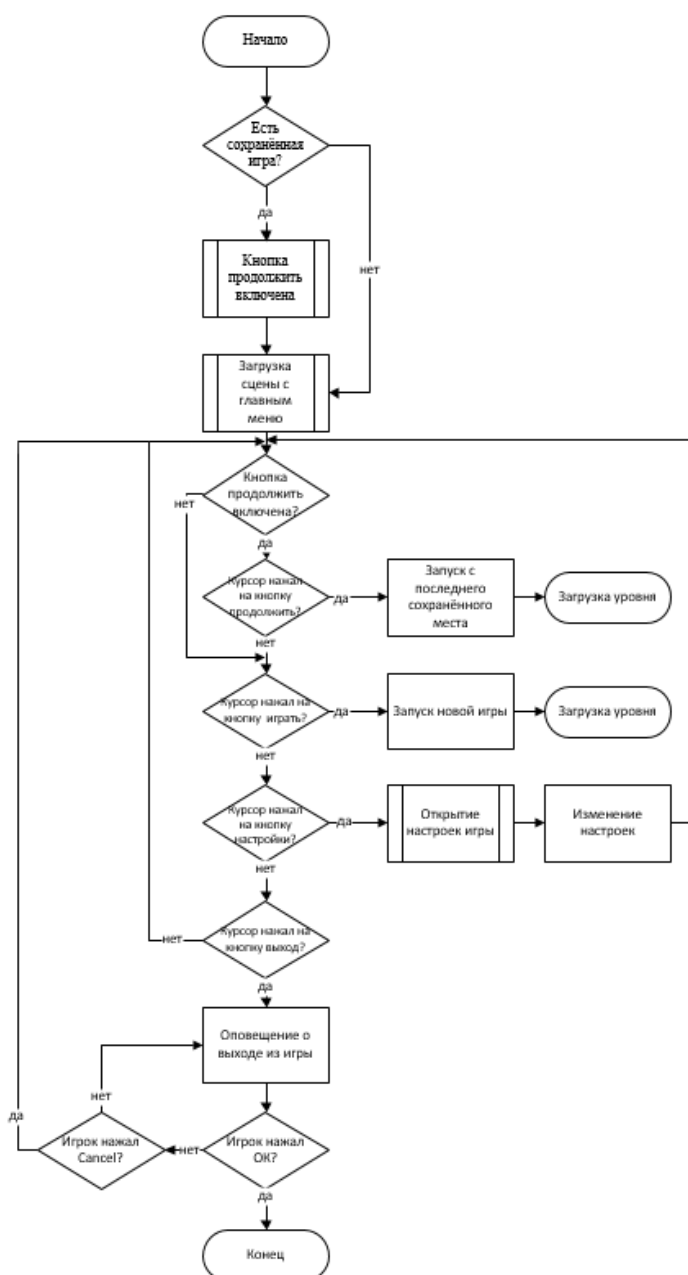


Рисунок 2.4 – Алгоритм работы главного меню

Из данного алгоритма следует что в первый запуск программы пользователю будет доступно всего три кнопки: «Играть», «Настройки» и «Выход», из-за того, что это первый запуск программы и сохранений еще нет, то кнопка «Продолжить» деактивирована и будет показана при следующих загрузках.

Кнопка «Играть» выполняет первую загрузку уровня и инициализацию всех игровых объектов. При подключении к своему миру так же генерирует ключ комнаты по которому другие игроки смогут подключиться к миру пользователя.

Кнопка «Настройки» выводит окно настроек для задания параметров конфигурации игры. В данном окне (см. Рисунок 2.6) будут располагаться варианты отображения графики, уровень прорисовки изображения, уровни громкости и изменяемые кнопки для взаимодействия пользователя с игровым пространством. На рисунке 2.5 представлен алгоритм работы окна настроек.

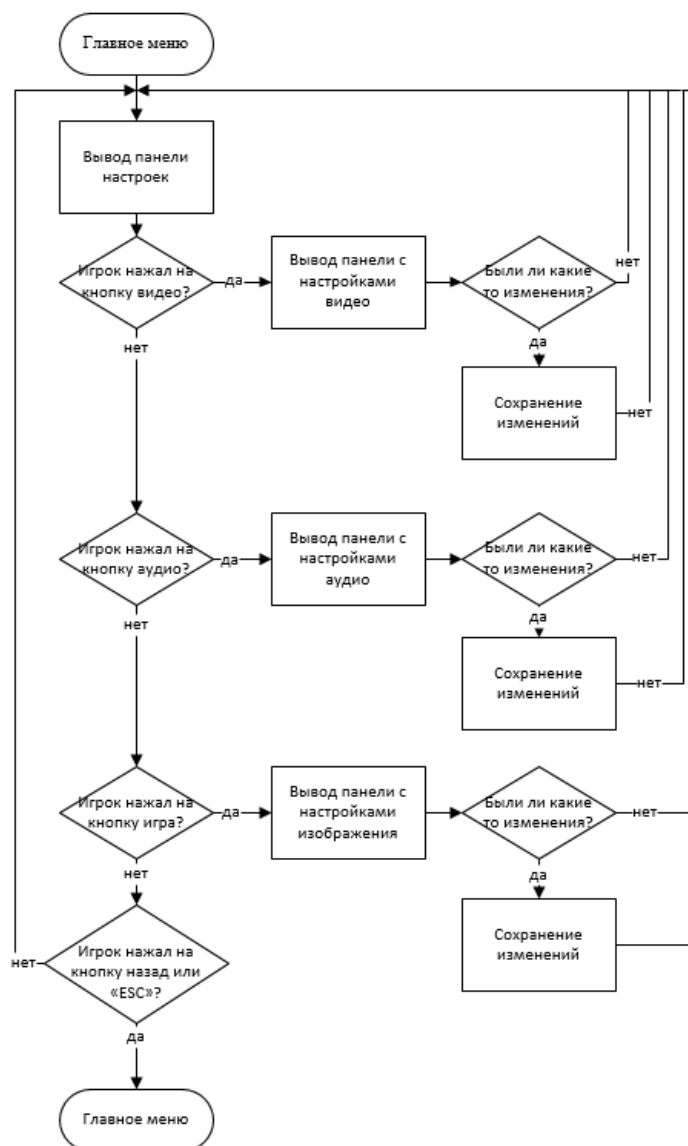


Рисунок 2.5 – Алгоритм работы окна настроек

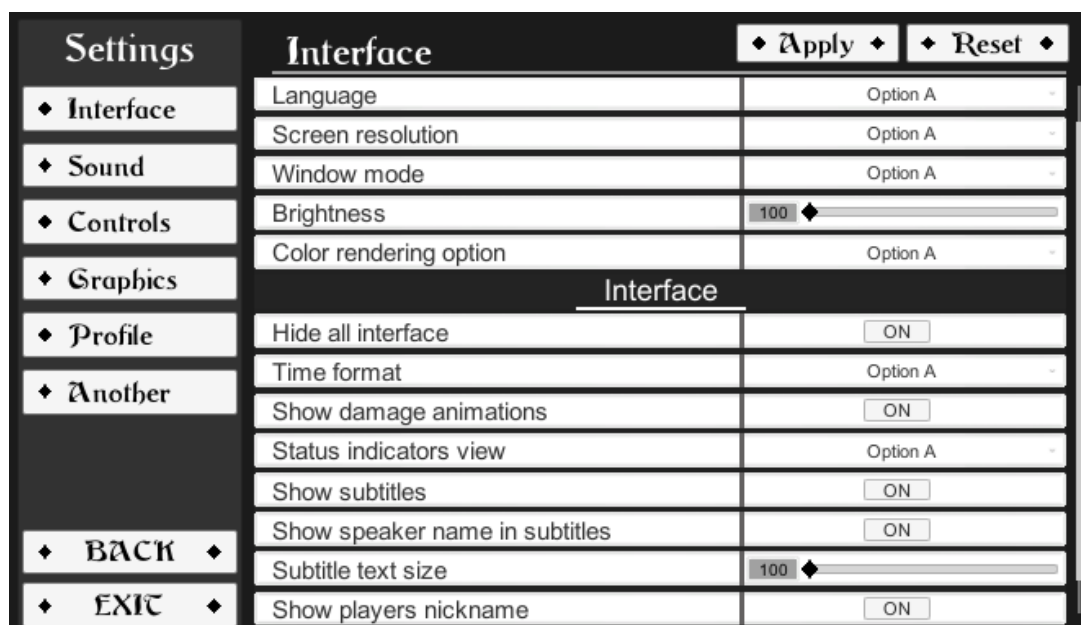


Рисунок 2.6 – Интерфейс настроек

Кнопка «Выход» позволяет пользователю выйти из приложения.

Кнопка «Продолжить» позволяет загрузить игру с заранее сохраненным прогрессом.

Для реализации интерфейса игры была использована система UI игрового движка Unity. Все элементы UI расположены на полотне «Canvas», которое определяет размещение 2D объектов в игровом пространстве. За это отвечает свойство Render Mode (см. Рисунок 2.7), которое содержит следующие значения:

Screen Space - Overlay. Полотно масштабируется по размерам экрана и рисуется поверх всех остальных графических элементов без связи со сценой или камерой. В случае изменения размера окна, полотно будет растянуто под размеры экрана.

Screen Space - Camera. Полотно рисуется на плоскости перпендикулярной взгляду камеры, на некотором расстоянии от точки взгляда. Размер масштабируется относительно заданных значений, чтобы заполнять разрез видимости камеры. Интерфейс будет заслоняться любыми 3D элементами, которые находятся перед плоскостью интерфейса.

World Space. Полотно располагается в мировых координатах и является плоским 3D объектом.

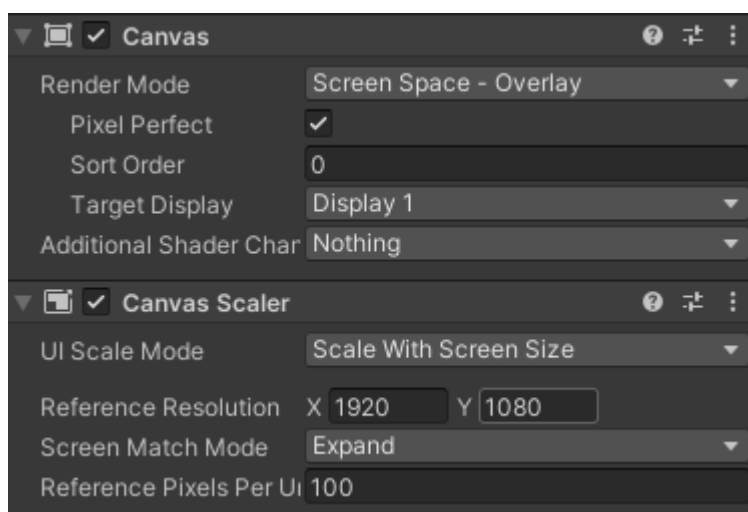


Рисунок 2.7 – Настройки полотна «Canvas»

На рисунке 2.8 показан построенный макет главного меню. Полотно, на котором располагаются UI элементы имеет свойство RenderMode с установленным значением Screen Space – Overlay.



Рисунок 2.8 – Главное меню

2.3 Выводы по главе

На основании результатов, полученных из первой главы, была спроектирована алгоритмическая часть архитектуры игры, представленная в виде диаграммы прецедентов и диаграммы классов. Так же был спроектирован графический интерфейс пользователя с помощью движка Unity. При помощи блок-схем описан алгоритм действий пользователя.

Глава 3 Разработка и тестирование.

Данная глава обозревает реализацию и тестирование конечного продукта. В данной главе рассматриваются все основы создания программы и тестирование. Итоговым результатом третьей главы является конечный продукт.

3.1 Обоснование выбора средств разработки

На основании сопоставления программного обеспечения во время анализа, для создания программного продукта, в качестве среды проектирования игры было выбрано программное обеспечение Unity с поддержкой Visual Studio, на основе языка программирования C#. [2]

Предоставленная система была выбрана по критериям, представленным ниже:

1. Большое количество документации и справочников, помогающие разобраться в написании игры.
2. Наличие большого набора структурных типов данных.
3. Лёгкость и удобность программного обеспечения.
4. Лицензия программного обеспечения является бесплатной.
5. Поддержка разработки трехмерных игр.

Одной из главных особенностей игрового движка Unity является удобный и полностью настраиваемый интерфейс, который сочетает в себе редактор сцен, игровых объектов, скриптов, анимации, а также предоставляет возможность мгновенного запуска и отладки разрабатываемого приложения или игры [6]. Все ресурсы и объекты могут перемещаться при помощи Drag-and-Drop методов.

В качестве инструмента для написания программного кода была выбрана среда Visual Studio 2019. В данной работе Visual Studio используется как лучшая альтернатива MonoDevelop, которая в игровом движке Unity представлена по умолчанию.

При будущей модернизации проекта возможен перенос всего проекта на язык C++ с использованием в качестве среды проектирования Unreal Engine, из-за большей ресурсоемкости и скорости работы языка программирования C++ и последующее сравнение скорости работы. [4]

Для реализации многопользовательского соединения было принято решение использовать встроенную библиотеку Unity «Photon», так как данная библиотека

является единственной библиотекой, подходящей для многопользовательского подключения для игр, созданных на платформе Unity. [7]

3.2 Описание используемых функций.

Для работы скриптов Unity существуют обязательные функции, которые обновляются на каждый кадр в зависимости от времени работы программы.

Функция «Awake» вызывается при запуске экземпляра скрипта. Данная функция применяется для инициализации переменных и состояния игры до того, как игра будет загружена. Так как эта функция вызывается всего один раз после запуска, а также в случайном порядке среди других скриптов, стоит отказаться от ее применения для передачи информации или установки связи между скриптами и использовать вместо нее функцию «Start», которая разрешает упорядочить инициализацию скриптов.

Функция «Update» вызывается перед отрисовкой кадра игры и перед прорисовкой анимации объектов, используется для анимирования и просчета кадров статичных объектов. Для не статичных объектов существует отдельная функция «FixedUpdate», которая просчитывает физику и анимацию объекта и обновляется фиксированными по времени шагами. [3]

Одним из главных скриптов в игре, является скрипт передвижения персонажа. На рисунке 3.1 представлен фрагмент кода, отвечающий за основные действия персонажа.

```
ссылка: 0
private void FixedUpdate()
{
    //до того пока не сделано действий - бездействие
    State = CharacterState.Idle;
    //передает в переменную данные о нажатой кнопке
    float v = Input.GetAxis("Vertical");

    if (Input.GetButton("Horizontal")) // ЕСЛИ входные данные - кнопки A или D
    {
        // ТО
        RunLR(); // Движение влево/вправо
    }
    if (Input.GetButton("Vertical")) // ЕСЛИ же входные данные - кнопки W или S
    {
        // ТО
        RunUD(v); // Движение вверх/вниз
    }

    if (Input.GetKeyDown(KeyCode.E) && ready)
    {
        Application.LoadLevel(Application.loadedLevel + 1);
    }
}
```

Рисунок 3.1 – Скрипт персонажа

Для противников в игре была написана функция поиска и преследования персонажа, при сближении противник начинает атаковать. Ниже представлены фрагменты кода данных функций. (см. Рисунок 3.2 и 3.3)

```
ССЫЛКА: 1
void Follow()
{
    if (face == Mode.Player) direction = player.right; else direction = (Mouse() - player.position).normalized;
    Vector3 position = player.position + direction * offset;
    position.z = transform.position.z;
    position = MoveInside(position, new Vector3(min.x, min.y, position.z), new Vector3(max.x, max.y, position.z));
    transform.position = Vector3.Lerp(transform.position, position, smooth * Time.deltaTime);
}
```

Рисунок 3.2 – Скрипт преследования

```
ССЫЛКА: 2
public void FindPlayer()
{
    player = GameObject.FindGameObjectWithTag("Player").transform;
    if (player) transform.position = new Vector3(player.position.x, player.position.y, transform.position.z);
}
```

Рисунок 3.3 – Скрипт поиска

Остальные функции представлены в полном листинге программы в ПРИЛОЖЕНИИ В.

3.3 Тестирование программного продукта

После написания программного кода в результате были реализованы основные функции, определенные в предыдущих главах. Этот этап является завершающим этапом создания программного продукта, который позволяет выявить проблемы и в последующем решить их при помощи отладки.

Тестирование программы проводилось с помощью группы тестировщиков и учащихся НИУ «ВШЭ-Пермь», добровольно согласившихся на участие, а также проводилось тестирование по критериям черного и белого ящиков (см. ПРИЛОЖЕНИЕ Б).

В ходе тестирования были выявлены ошибки, связанные с прохождением через элементы окружения из-за неправильного расположения данных элементов в игровом пространстве.

По окончании разработки системы, было составлено руководство программиста в ПРИЛОЖЕНИИ Е. Листинг программы приведен в ПРИЛОЖЕНИИ Б.

3.4 Выводы по главе

Исходя из спроектированных моделей и графического интерфейса, игровая логика, полученная во второй главе, была реализована в виде программного кода в системе разработки Unity. В данной главе приводится описание механизма взаимодействия пользователя с игровым пространством. Разработанные классы описывают основную часть функционала игры.

Заключение

По результату проделанной работы, была разработана многопользовательская игра в жанре «RPG». Для этого были исследованы основы работы в программном обеспечении Unity и рассмотрена литература на эту тему. Были изучены особенности жанра «RPG» и выявлены основные элементы предоставленного жанра. Был проведен обзор игр-аналогов для выявления плюсов и минусов, для выдвижения требований к создаваемой игре.

При проектировании были использована библиотека C#, необходимая для, написания кода приложения. Графический интерфейс был спроектирован и разработан благодаря инструментам Unity, с помощью Visual Studio.

Программу возможно усовершенствовать, добавив дополнительное количество уровней, дорисовав текстуры и усложнив игровой процесс. Так же можно добавить поддержку разных языков.

В течении работы над программой были улучшены навыки ООП и познания в сфере разработки игр, а также усовершенствован навык проектирования программного продукта и его анализа.

Положительным итогом проведенной работы является усовершенствование навыков написания программ на объектно-ориентированном языке программирования C# и создания игр под Windows и MacOS.

Список сокращений и условных обозначений

1. RPG – жанр компьютерных игр с упором ролевую игру с большим количеством диалогов, свободой в выборе путей решения различных задач, проработанным миром и сюжетом.
2. Очки здоровья (hp) – здоровье персонажа.
3. Движок - (graphics engine) – промежуточное программное обеспечение, программный движок, основной задачей которого является визуализация (рендеринг) двухмерной или трёхмерной компьютерной графики.
4. UML – унифицированный язык моделирования (Unified Modeling Language) система обозначений, которую можно применять для объектно-ориентированного анализа и проектирования
5. UI – Интерфейс пользователя, он же пользовательский интерфейс (user interface) – интерфейс, обеспечивающий передачу информации между пользователем-человеком и программно-аппаратными компонентами компьютерной системы
6. ПО – программное обеспечение.

Библиографический список

1. Арсаж, Ж. Программирование игр и головоломок/Ж. Арсаж, -М.: Книга по Требованию, 2012.
2. Бочкарев Н. А., Молотов Р. С. Подходы к трансформации объектов виртуальных пространств в среде Unity //Вестник Ульяновского государственного технического университета. – 2016. – №. 3 (75).
3. Никулин Е.А. Компьютерная графика. Модели и алгоритмы. СПб: издательство "Лань". - 708 с. (2017).
4. Официальная документация по Unity3D [Электронный ресурс] //URL:<http://unity3d.com/support/documentation>
5. Создание проектов и схем моделирования UML [Электронный ресурс] //URL:<https://msdn.microsoft.com/ru-ru/library/dd409445.aspx?f=255&MSPPErr=-2147217396>
6. Торн, Алан. Искусство создания сценариев в Unity : [рус.]. — СПб : ДМК, 2016. — 362 с.
7. Хокинг, Джозеф. Unity — в действии. Мультиплатформенная разработка на C# : [рус.]. — 2. — СПб : Питер, 2016. — 336 с.
8. Blow J. Game development: Harder than you think //Queue. – 2004. – Т. 1. – №. 10.
9. Diveky, Marko; Bielikova, Maria (September 29–October 2, 2009). "Generating Educational Interactive Stories in Computer Role-Playing Games". Learning in the Synergy of Multiple Disciplines: 4th European Conference on Technology Enhanced Learning, Proceedings, Nice, France: Springer.
10. Claypool K., Claypool M. Teaching software engineering through game design //ACM SIGCSE Bulletin. – 2005. – Т. 37. – №. 3.
11. Jeff Craighead, Jenny Burke, Robin Murphy. “Using the Unity Game Engine to Develop SARGE: A Case Study”. Proceedings of the 2008 Simulation Workshop at the International Conference on Intelligent Robots and Systems (IROS 2008). September 2008
12. Hejlsberg A., Wiltamuth S., Golde P. The C# programming language. – 2006.

13. McNaughton, M.; Schaeffer, J.; Szafron, D.; Parker, D.; Redford J. Code Generation for AI Scripting in Computer Role-Playing Games (PDF). American Association for Artificial Intelligence (2004).
14. Stuart, Keith. 3D games enter a new generation, The Observer, London: The Guardian (19 сентября 2010).

ПРИЛОЖЕНИЕ А

Пользовательская документация

Данная игра является курсовой работой студента первого курса федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский университет «Высшая школа экономики» факультета экономики, менеджмента и бизнес-информатики направления программная инженерия группы ПИ-18-2 Чепокова Е.С

Документация описана по ГОСТ РД 50-34.698-90.

Введение

Компьютерная игра «Untitled game» – многопользовательская ролевая онлайн игра для платформ Windows и MacOS. Игра используется в развлекательных целях

Помимо данной пользовательской документации не предусмотрено эксплуатационной документации, необходимой к ознакомлению.

Назначение и условия применения

Данный вариант игры подготовлен для выявления ошибок и их последующего исправления. Документация написана с целью информирования о способах установить игру и способах связаться с разработчиком.

Подготовка к работе

1. Скачивание игры

Скачивание файла с игрой осуществляется на сайте игры bloodiesproject.gq. Для скачивания нужно нажать на кнопку скачать. После чего вас направит на сайт Яндекс диска, откуда и можно скачать сжатый файл расширения .zip с игрой.

2. Установка

Клиент для Windows представляет собой в zip-архив. Распакуйте клиент средствами операционной системы или утилитами архивирования.

Начало игры

1. Для запуска игры используйте исполняемый файл Game.exe, расположенный в папке игры.
2. После запуска последует данное окно (рисунок А.1), в нем нужно выбрать разрешение и монитор для отображения

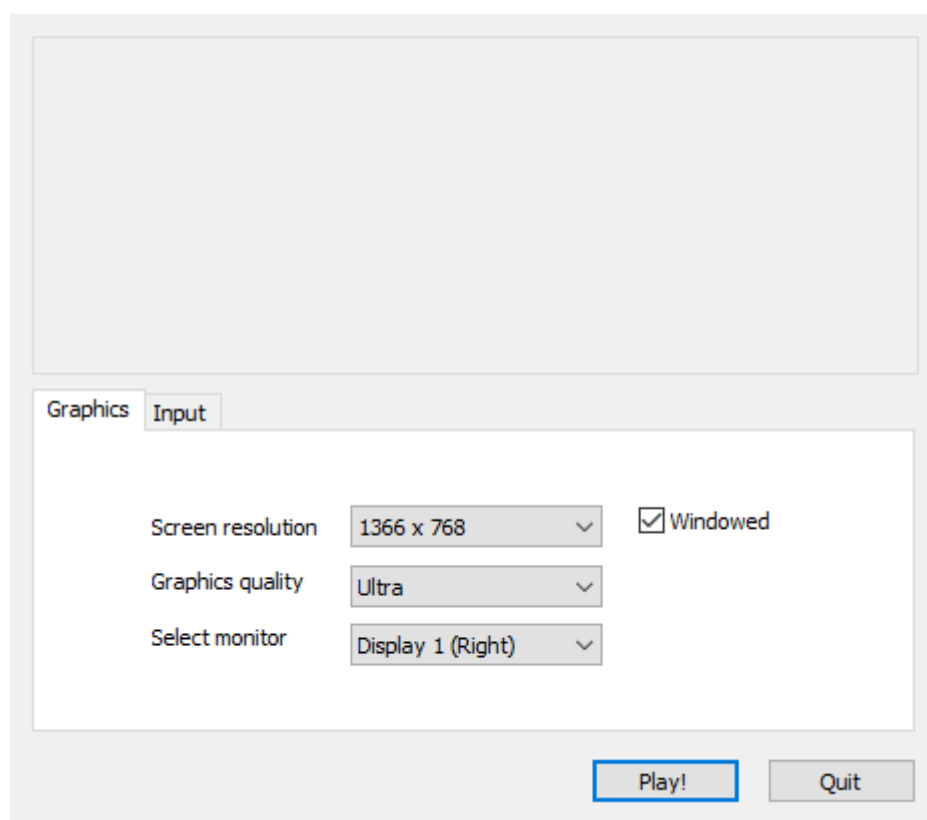


Рисунок А.1. Конфигурация игры

3. После запуска, в главном меню выбираете кнопку «Играть» и начинаете играть

Для экстренных ситуаций

Действия по восстановлению программы при обнаружении ошибок в данных
При неуспешной попытке выполнить какую-либо функцию рекомендуется перезапустить приложение.

При любых аварийных ситуациях писать разработчику, с описанием проблемы в любой форме на почту или в сообщения «ВКонтакте»:

bloodiesco@yandex.ru

<https://vk.com/bloodiesprod>

ПРИЛОЖЕНИЕ Б

Тестирование по критериям черного ящика

Таблица Б.1. Чёрный ящик

№ Теста	Действие	Ожидаемый результат	Реальный результат
T1	Нажатие кнопки «Продолжить» без сохранения	Кнопка не нажимается	Кнопка не нажимается
T2	Нажатие кнопки «Продолжить» с сохранением	Загрузка сохраненного уровня	Загрузка сохраненного уровня
T3	Нажатие кнопки «Новая игра»	Загрузка первого уровня	Загрузка первого уровня
T4	Нажатие кнопки «Настройки»	Открытие панели настроек	Открытие панели настроек
T5	Нажатие кнопки «Выход»	Сообщение о выходе	Сообщение о выходе
T6	Изменение настроек аудио и кнопка «Принять изменения»	Изменения принимаются	Изменения принимаются
T7	Изменение настроек видео и кнопка «Принять изменения»	Изменения принимаются	Изменения принимаются
T8	Изменение настроек игры и кнопка «Принять изменения»	Изменения принимаются	Изменения принимаются
T9	Изменение настроек аудио без кнопки «Принять изменения»	Изменения не принимаются	Изменения принимаются
T10	Изменение настроек видео и кнопки «Принять изменения»	Изменения не принимаются	Изменения не принимаются
T11	Изменение настроек игры и кнопки «Принять изменения»	Изменения не принимаются	Изменения не принимаются
T12	Движение персонажем через стены	Персонаж упирается в стену	Персонаж упирается в стену
T13	Беспорядочное нажатие на все кнопки	Сообщение об ошибке	Залипание клавиш
T14	Подключение геймпада	С него можно управлять	С него можно управлять
T15	Закрытие через диспетчер задач	Сохранение было произведено	Сохранение было произведено

ПРИЛОЖЕНИЕ В

Полный листинг скриптов игры

Скрипт для персонажа:

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Character : MonoBehaviour
{
    [SerializeField]
    private int life = 1;
    [SerializeField]
    private float speed = 3.0f;

    new private Rigidbody rigidbody;
    private Animator animation;
    private SpriteRenderer sprite;
    private bool ready = true;

    private CharacterState State
    {
        get
        {
            return (CharacterState)animation.GetInteger("State");
        }
        set
        {
            animation.SetInteger("State", (int)value);
        }
    }

    private void OnCollisionEnter(Collision collision)
    {
        if (collision.gameObject.tag == "EndLevel")
        {
            ready = true;
        }
    }

    private void Awake()
    {
        rigidbody = GetComponent<Rigidbody>();
        animation = GetComponent<Animator>();
        sprite = GetComponentInChildren<SpriteRenderer>();
        Cursor.lockState = CursorLockMode.Locked;
    }

    private void FixedUpdate()
    {
        State = CharacterState.Idle;
        float v = Input.GetAxis("Vertical");

        if (Input.GetButton("Horizontal"))
        {
            RunLR();
        }
        if (Input.GetButton("Vertical"))
        {
            RunUD(v);
        }
    }
}
```

```

        if (Input.GetKeyDown(KeyCode.E) && ready)
        {
            Application.LoadLevel(Application.loadedLevel + 1);
        }
    }
    private void RunLR()
    {
        Vector3 HorizontDirection = transform.right * Input.GetAxis("Horizontal");

        transform.position = Vector3.MoveTowards(transform.position, transform.position +
        HorizontDirection, speed * Time.deltaTime);

        sprite.flipX = HorizontDirection.x < 0.0F;

        State = CharacterState.Movement_right;
    }
    private void RunUD(float vertical)
    {
        Vector2 VertDirection = transform.forward * Input.GetAxis("Vertical");

        transform.Translate(new Vector2(0, speed * vertical * Time.fixedDeltaTime)); //
        движение по лестнице

        if (Input.GetKey(KeyCode.S) || Input.GetKey(KeyCode.DownArrow) &&
        !(Input.GetKey(KeyCode.W) || Input.GetKey(KeyCode.UpArrow)))
        {
            State = CharacterState.Movement_down;
        }
        else
        {
            State = CharacterState.Movement_up;
        }
        if ((Input.GetKey(KeyCode.W) || Input.GetKey(KeyCode.UpArrow)) &&
        !(Input.GetKey(KeyCode.S) || Input.GetKey(KeyCode.DownArrow)))
        {
            State = CharacterState.Movement_up;
        }
        else
        {
            State = CharacterState.Movement_down;
        }
    }
}

public enum CharacterState
{
    Idle,
    Movement_right,
    Movement_left,
    Movement_up,
    Movement_down
}

```

Скрипт для изменения клавиш:

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

```

```

public class ChangeKeyboard : MonoBehaviour {

    public CustomInput customInput;
    public Button[] button;

    public bool canChangeKey;
    public KeyCode newKey;
    private Event eventKey;
    public int indexChange=0;

    // Use this for initialization
    void Start () {
        canChangeKey = false;
        SetKeys ();
        UpdateTextUI ();
    }

    // Update is called once per frame
    void Update () {
    }

    void OnGUI(){
        button [0].onClick.AddListener (delegate {
            ChangeKeyButton (0);
        });

        button [1].onClick.AddListener (delegate {
            ChangeKeyButton (1);
        });

        button [2].onClick.AddListener (delegate {
            ChangeKeyButton (2);
        });

        button [3].onClick.AddListener (delegate {
            ChangeKeyButton (3);
        });

        button [4].onClick.AddListener (delegate {
            ChangeKeyButton (4);
        });

        button [5].onClick.AddListener (delegate {
            ChangeKeyButton (5);
        });

        ChangeKey ();
        if (Input.anyKeyDown) {
            if (canChangeKey == true) {
                if (indexChange == 0) {
                    customInput.forward = newKey;
                } else if (indexChange == 1) {
                    customInput.back = newKey;
                } else if (indexChange == 2) {
                    customInput.right = newKey;
                } else if (indexChange == 3) {
                    customInput.left = newKey;
                } else if (indexChange == 4) {
                    customInput.crouch = newKey;
                } else if (indexChange == 5) {
                    customInput.jump = newKey;
                }
                SetKeyDefault ();
                UpdateTextUI ();
            }
        }
    }
}

```

```

        button [indexChange].enabled = true;
        canChangeKey = false;
    }
}

public void ChangeKeyButton(int indexKey){
    indexChange = indexKey;
    canChangeKey = true;
    button [indexKey].GetComponentInChildren<Text> ().text = "WAIT FOR KEY";
    button [indexKey].enabled = false;
}

void ChangeKey(){
    eventKey = Event.current;
    newKey = eventKey.keyCode;
}

void UpdateTextUI(){
    button [0].GetComponentInChildren<Text> ().text =
customInput.forward.ToString ();
    button [1].GetComponentInChildren<Text> ().text = customInput.back.ToString
();
    button [2].GetComponentInChildren<Text> ().text =
customInput.right.ToString ();
    button [3].GetComponentInChildren<Text> ().text = customInput.left.ToString
();
    button [4].GetComponentInChildren<Text> ().text =
customInput.crouch.ToString ();
    button [5].GetComponentInChildren<Text> ().text = customInput.jump.ToString
();
}

public void SetKeyDefault(){
    customInput.forwardDefaultKey = customInput.forward.ToString ();
    customInput.backDefaultKey = customInput.back.ToString ();
    customInput.rightDefaultKey = customInput.right.ToString ();
    customInput.leftDefaultKey = customInput.left.ToString ();
    customInput.crouchDefaultKey = customInput.crouch.ToString ();
    customInput.jumpDefaultKey = customInput.jump.ToString ();
}

public void SetKeys(){
    customInput.forward = (KeyCode) System.Enum.Parse(typeof(KeyCode),
customInput.forwardDefaultKey);
    customInput.back = (KeyCode) System.Enum.Parse(typeof(KeyCode),
customInput.backDefaultKey);
    customInput.right = (KeyCode) System.Enum.Parse(typeof(KeyCode),
customInput.rightDefaultKey);
    customInput.left = (KeyCode) System.Enum.Parse(typeof(KeyCode),
customInput.leftDefaultKey);
    customInput.crouch = (KeyCode) System.Enum.Parse(typeof(KeyCode),
customInput.crouchDefaultKey);
    customInput.jump = (KeyCode) System.Enum.Parse(typeof(KeyCode),
customInput.jumpDefaultKey);
}
}

```

Скрипт для изменения языка:

```

using System.Collections;
using System.Collections.Generic;
using System.IO;
using UnityEngine;
using UnityEngine.UI;

```

```

public class LanguageControl : MonoBehaviour {

    //[Header("JSON File Names")]
    //public string mainScreenLanguage = "MainScreenLanguage";
    //public string optionsScreenLanguage = "OptionsScreenLanguages";

    public Text[] main_titles;
    public Text[] buttons_options;
    public Text[] options_video;
    public Text[] options_audio;
    public Text[] options_game;

    public WS_MAIN_LANGUAGE ws_main_language;
    public WS_OPTIONS_LANGUAGE ws_options_language;

    private MenuControl _menuControl;
    private OptionsControl _optionsControl;

    //Editor
    [HideInInspector]
    public int currentTab;
    [HideInInspector]
    public int currentTabTwo;
    [HideInInspector]
    public int previousTab = -1;
    [HideInInspector]
    public int previousTabTwo = -1;

    // Use this for initialization
    void Start () {
        ws_main_language = new WS_MAIN_LANGUAGE ();
        ws_options_language = new WS_OPTIONS_LANGUAGE ();
        _menuControl = this.GetComponent<MenuControl> ();
        _optionsControl = this.GetComponent<OptionsControl> ();

    }

    // Update is called once per frame
    void Update () {

    }

    // SET ALL CHANGES
    public void SetLanguageInGame(){
        SetMainLanguage ();
        SetOptionsLanguage ();
    }// END

    // MAIN MENU
    public void SetMainLanguage(){
        if (_optionsControl != null && _optionsControl._gameConfig != null &&
ws_main_language != null) {
            if (_optionsControl._gameConfig.language == 0) { // ENGLISH
                //Debug.Log ("Change To English!");
                if (_menuControl.inGame == false)
                {
                    main_titles[0].text = ws_main_language.play_en;
                    main_titles[1].text = ws_main_language.options_en;
                    _menuControl.menu[0].SetText(ws_main_language.continue_en);
                    _menuControl.menu[1].SetText(ws_main_language.play_en);
                    _menuControl.menu[2].SetText(ws_main_language.options_en);
                    _menuControl.menu[3].SetText(ws_main_language.credits_en);
                    _menuControl.menu[4].SetText(ws_main_language.exit_en);

                }
            }
        }
    }
}

```



```

        options_game [1].text =
ws_options_language.vericalSensitivity_en;
        options_game [2].text = ws_options_language.difficulty_en;
        options_game [3].text = ws_options_language.language_en;
        options_game [4].text = ws_options_language.forward_en;
        options_game [5].text = ws_options_language.back_en;
        options_game [6].text = ws_options_language.left_en;
        options_game [7].text = ws_options_language.right_en;
        options_game [8].text = ws_options_language.crouch_en;
        options_game [9].text = ws_options_language.jump_en;
        options_game [10].text = ws_options_language.tips_en;

    } else if (_optionsControl._gameConfig.language == 1) { // RU
        // BUTTONS
        buttons_options[0].text = ws_options_language.video_ptbr;
        buttons_options[1].text = ws_options_language.audio_ptbr;
        buttons_options[2].text = ws_options_language.game_ptbr;
        buttons_options[3].text = ws_options_language.apply_ptbr;
        buttons_options[4].text = ws_options_language.return_ptbr;

        // VIDEO
        options_video[0].text = ws_options_language.displayMode_ptbr;
        options_video[1].text =
ws_options_language.targetDisplay_ptbr;
        options_video[2].text = ws_options_language.resolution_ptbr;
        options_video[3].text =
ws_options_language.graphicsQuality_ptbr;
        options_video[4].text = ws_options_language.antiAliasing_ptbr;
        options_video[5].text = ws_options_language.vsync_ptbr;

        // AUDIO
        options_audio [0].text =
ws_options_language.masterVolume_ptbr;
        options_audio [1].text = ws_options_language.musicVolume_ptbr;
        options_audio [2].text =
ws_options_language.effectsVolume_ptbr;
        options_audio [3].text = ws_options_language.voiceVolume_ptbr;
        options_audio [4].text = ws_options_language.micVolume_ptbr;
        options_audio [5].text =
ws_options_language.soundBackground_ptbr;

        // GAME
        options_game [0].text =
ws_options_language.horizontalSensitivity_ptbr;
        options_game [1].text =
ws_options_language.vericalSensitivity_ptbr;
        options_game [2].text = ws_options_language.difficulty_ptbr;
        options_game [3].text = ws_options_language.language_ptbr;
        options_game [4].text = ws_options_language.forward_ptbr;
        options_game [5].text = ws_options_language.back_ptbr;
        options_game [6].text = ws_options_language.left_ptbr;
        options_game [7].text = ws_options_language.right_ptbr;
        options_game [8].text = ws_options_language.crouch_ptbr;
        options_game [9].text = ws_options_language.jump_ptbr;
        options_game [10].text = ws_options_language.tips_ptbr;
    }
} // END
}

using UnityEngine;

public class WS_MAIN_LANGUAGE {

    // ENGLISH

```

```

        public string continue_en = " CONTINUE";
        public string play_en = " PLAY";
        public string options_en = " OPTIONS";
        public string credits_en = " CREDITS";
        public string exit_en = " QUIT";

        public string resume_en = "RESUME";

        // RUS
        public string continue_ptbr = " Продолжить";
        public string play_ptbr = " Новая игра";
        public string options_ptbr = " Настройки";
        public string credits_ptbr = " Титры";
        public string exit_ptbr = " Выйти";

        public string resume_ptbr = "Продолжить";
    }

    using UnityEngine;

    public class WS_OPTIONS_LANGUAGE {

        //***** ENGLISH
        *****\\
        public string video_en = "VIDEO";
        public string audio_en = "AUDIO";
        public string game_en = "GAME";
        public string apply_en = "APPLY";
        public string return_en = "RETURN";

        // VIDEO
        public string displayMode_en = " DISPLAY MODE";
        public string targetDisplay_en = " TARGET DISPLAY";
        public string resolution_en = " RESOLUTION";
        public string graphicsQuality_en = " GRAPHICS QUALITY";
        public string antialiasing_en = " ANTIALIASING";
        public string vsync_en = " V-Sync";

        // AUDIO
        public string masterVolume_en = " MASTER VOLUME";
        public string musicVolume_en = " MUSIC VOLUME";
        public string effectsVolume_en = " EFFECTS VOLUME";
        public string voiceVolume_en = " VOICE VOLUME";
        public string micVolume_en = " MIC VOLUME";
        public string soundBackground_en = " SOUND BACKGROUND";

        // GAME
        public string horizontalSensitivty_en = " HORIZONTAL SENSITIVY";
        public string vericalSensitivty_en = " VERTICAL SENSITIVY";
        public string difficulty_en = " DIFFICULTY";
        public string language_en = " LANGUAGE";
        public string forward_en = " UP";
        public string back_en = " DOWN";
        public string left_en = " LEFT";
        public string right_en = " RIGHT";
        public string crouch_en = " CROUCH";
        public string jump_en = " JUMP";
        public string tips_en = " TIPS";

        //***** RUS
        *****\\
        public string video_ptbr = "Видео";
        public string audio_ptbr = "Аудио";
        public string game_ptbr = "Игра";

```



```

public string apply_ptbr = "Принять изменения";
public string return_ptbr = "Назад";

// VIDEO
public string displayMode_ptbr = " Дисплей";
public string targetDisplay_ptbr = " Монитор";
public string resolution_ptbr = " Разрешение";
public string graphicsQuality_ptbr = " Качество графики";
public string antialiasing_ptbr = " Сглаживание";
public string vsync_ptbr = " V-Sync";

// AUDIO
public string masterVolume_ptbr = " Общая громкость";
public string musicVolume_ptbr = " Громкость музыки";
public string effectsVolume_ptbr = " Громкость эффектов";
public string voiceVolume_ptbr = " Громкость голоса";
public string micVolume_ptbr = " Громкость микрофона";
public string soundBackground_ptbr = " Громкость окружения";

// GAME
public string horizontalSensitivity_ptbr = " Горизонтальная чувствительность";
public string verticalSensitivity_ptbr = " Вертикальная чувствительность";
public string difficulty_ptbr = " Сложность";
public string language_ptbr = " Язык";
public string forward_ptbr = " Вверх";
public string back_ptbr = " Вниз";
public string left_ptbr = " Налево";
public string right_ptbr = " Направо";
public string crouch_ptbr = " Присесть";
public string jump_ptbr = " Прыжок";
public string tips_ptbr = " Атака";
}

```

Скрипт для меню:

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.Events;

[System.Serializable]
public class Menu : MonoBehaviour {

    [Header("Menu Settings")]
    public bool active;

    // Responsible variable if the
    button is active or not

    public int maxFontSize = 22;
    // Maximum font size

    public int minFontSize = 18;
    // Minimum font size

    public Color mouseEnter;
    // MenuItem color when the mouse enters

    public Color mouseExit;
    // MenuItem color when the mouse
    exits

    public Color mousePressed;
    // MenuItem color when the mouse pressed /
    click

```

```

public Color deactivatedColor;
// MenuItem color when disabled

[Header("Line Settings")]
public bool enableLine;
// Enable underline
public bool enableLineEffect;
// Enable effect when mouse enters
public float widthLine=200f;
// Width underline
public float heightLineMin=1f;
// Minimum height underline
public float heightLineMax=2f;
// Maximum height underline

[Header("Animation Settings")]
public bool initAnim;
// Initial animation
public float timerInitAnim;
// Time for animation starts

public float menuXStart = -233f;
// Initial X-axis
public float menuXEnd = 115f;
// End X-axis
public float speedAnim = 400f;
// Speed animation

[Header("Events")]
[SerializeField]
private UnityEvent Enter = new UnityEvent ();

[SerializeField]
private UnityEvent Exit = new UnityEvent ();

[SerializeField]
private UnityEvent Click = new UnityEvent ();

// Variables that the user does not need to change
private MenuControl _menuc;
// Menu Control Component
private Image _effectSelected;
// Underline Component
private Text _text;
// Text Component
private Vector3 _initPos;
// Initial position
private RectTransform _rect;
// RectTransform component

// Use this for initialization
void Start () {
    getComponents ();
    basicSettings ();
}

// Update is called once per frame
void Update () {
    if (initAnim == true) {
        updateAnimation ();
    }
}

```

```

//-----
START METHODS MENUITEM-----
-\\
    // Get the components
    void getComponents(){
        _rect = this.GetComponent<RectTransform> ();
        // Get the RectTransform component of this object
        _menuc = FindObjectOfType<MenuControl> ();
        // Get the Control Menu (there should only be
one)
        _text = this.GetComponent<Text> ();
        // Get the Text component of children
        _effectSelected = this.GetComponentInChildren<Image> ();
        // Get the Image component of children
    }// END

    // Basic and necessary settings
    void basicSettings(){
        _initPos = _rect.localPosition;
        // Get initial position

        if (initAnim == true) {
            // If the initial animation is true
            _rect.localPosition = new Vector3(menuXStart, _initPos.y,
_initPos.z);
            // Arrow the position of the object to the X axis of the
variable "menuXStart"
        }

        //Set Default Color
        _text.color = mouseExit;
        _effectSelected.color = mouseExit;

        // If the button is not active
        if (active == false) {
            _text.color = deactivatedColor; // Set color to "deactivatedColor"
        }

        // If the underline is false it defines the effect of the underline as
false too
        if (enableLine == false) {
            _effectSelected.enabled = false;
        }

    }// END

    // Update initial animation
    void updateAnimation(){
        // If the time to start the animation is over
        if (timerInitAnim <= 0) {
            _rect.transform.localPosition = Vector2.MoveTowards
(_rect.transform.localPosition, new Vector2 (menuXEnd, _initPos.y), speedAnim *
Time.deltaTime); // Starts Animation
        }
        if (timerInitAnim >= 0) {timerInitAnim -= Time.deltaTime;} // If the
animation time is greater than zero (ie not started) it starts to go down 1 second
    }// END
//-----END
METHODS MENUITEM-----\\

//-----
START METHODS ON/OFF-----
-\\
    // Method by setting the default menuItem again (mouseExit)
    public void menuDisable(){

```

```

        if (active == true) {
            // If the
button is active
            _text.color = mouseExit;
            // Sets the default
color
            _effectSelected.color = mouseExit;
            // Sets the default
color (underline)
            if (enableLineEffect==true) {
                // If the underline
effect is active
                _effectSelected.rectTransform.sizeDelta = new Vector2
(widthLine, heightLineMin); // Set a new size
            }
            _text.fontSize = minFontSize;
            // Sets the default
font size
        }
    }// END

    // Method by setting the active menuItem (mouseenter)
    public void menuEnable(){
        if (active == true) {
            // If the
button is active
            _text.color = mouseEnter;
            // Sets new color
(mouseEnter)
            _effectSelected.color = mouseEnter;
            // Sets new color for
underline (mouseenter)
            if (enableLineEffect==true) {
                // If the underline
effect is active
                _effectSelected.rectTransform.sizeDelta = new Vector2
(widthLine, heightLineMax); // Set a new size
            }
            _text.fontSize = maxFontSize;
            // Sets font size to
max font size
        }
    }// END

    // Set underline is active or no
    public void setMenuLine(bool value){
        enableLine = value;
        // Set value
        _effectSelected.gameObject.SetActive (value);
        // Set value
    }// END

    // Activate an object and mask it
    public void enableObject(GameObject obj){
        obj.SetActive (true);
        // Active the
object
        if (_menuc.inGame == false)
        {
            _menuc.mask.SetActive(true);
            // Active the mask
            _menuc.setAlphaMask(0.5f);
            // Set alpha of mask to 0.5f
        }
    }// END

```

```

//-----END
METHODS ON/OFF-----\\

    public void CallTheEvent(int index){
        if (index == 0) {
            Enter.Invoke ();
        } else if (index == 1) {
            Exit.Invoke ();
        } else if (index == 2) {
            Click.Invoke ();
        }
    }
} // END

    public void SetText(string newText){
        this.GetComponent<Text> ().text = newText;
    }

    public void showMessageInConsole(string s){
        Debug.Log (s);
    }
}

```

Скрипт для контроллера меню:

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using System.IO;
using UnityEngine.SceneManagement;

public class SettingsButtonManager : MonoBehaviour
{
    public GameObject Canvas;

    public bool isActiveCanvas = false; //Открыто ли меню

    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
        if (Input.GetKeyDown(KeyCode.Escape))
        {
            ShowHideMenu();
        }
    }

    public void Show_panel(GameObject obj)
    {
        obj.SetActive(true);
    }

    public void Hide_panel(GameObject obj)
    {
        obj.SetActive(false);
    }

    public void ShowHideMenu()
    {

```

```

        isActiveCanvas = !isActiveCanvas;
        if (isActiveCanvas == false)
        {
            Cursor.SetCursor(null, Vector2.zero, CursorMode.Auto);
            Time.timeScale = 1;
            Cursor.lockState = CursorLockMode.Locked;
        }
        else
        {
            Cursor.lockState = CursorLockMode.None;
            Cursor.SetCursor(null, Vector2.zero, CursorMode.ForceSoftware);
            Time.timeScale = 0;
        }
        //GetComponent<Canvas>().enabled = isOpened; //Включение или отключение Canvas.
Ещё тут можно использовать метод SetActive()
        Canvas.SetActive(isActiveCanvas);
    }

    public void QuitGame()
    {
        Application.Quit(); //Заккрытие игры. В редакторе, конечно, она закрыта не будет,
поэтому для проверки можно использовать Debug.Log();
    }

    public void GoToMain()
    {
        SceneManager.LoadScene("Main_screen"); //Переход на сцену с названием Menu
    }

    public void GoBack()
    {
        ShowHideMenu();
    }
}

```

Скрипт для подключения:

```

using UnityEngine;
using UnityEngine.SceneManagement;
using UnityStandardAssets.CrossPlatformInput;
using UnityStandardAssets.Cameras;

public class WorkerInGame : Photon.MonoBehaviour
{
    public FreeLookCam Camera;

    public void Awake()
    {
        Camera = GameObject.FindObjectOfType<FreeLookCam> ();

        // in case we started this demo with the wrong scene being active, simply load the
menu scene
        if (!PhotonNetwork.connected)
        {
            SceneManager.LoadScene(WorkerMenu.SceneNameMenu);
        }
    }
}

```

```

        return;
    }

    Vector3 position = new Vector3( 1000, 50, 300 );
    // we're in a room. spawn a character for the local player. it gets synced by using
    PhotonNetwork.Instantiate
        GameObject    newPlayerObject=PhotonNetwork.Instantiate("UMARPG",    position,
    Quaternion.identity, 0);
        Camera.SetTarget(newPlayerObject.transform);
    }

    public void OnGUI()
    {
        if (GUILayout.Button("Return to Lobby"))
        {
            PhotonNetwork.LeaveRoom(); // we will load the menu level when we successfully
left the room
        }
    }

    public void OnMasterClientSwitched(PhotonPlayer player)
    {
        Debug.Log("OnMasterClientSwitched: " + player);

        string message;
        InRoomChat chatComponent = GetComponent<InRoomChat>(); // if we find a InRoomChat
component, we print out a short message

        if (chatComponent != null)
        {
            // to check if this client is the new master...
            if (player.isLocal)
            {
                message = "You are Master Client now.";
            }
            else
            {
                message = player.name + " is Master Client now.";
            }

            chatComponent.AddLine(message); // the Chat method is a RPC. as we don't want
to send an RPC and neither create a PhotonMessageInfo, lets call AddLine()

```

```

    }
}

public void OnLeftRoom()
{
    Debug.Log("OnLeftRoom (local)");

    // back to main menu
    SceneManager.LoadScene(WorkerMenu.SceneNameMenu);
}

public void OnDisconnectedFromPhoton()
{
    Debug.Log("OnDisconnectedFromPhoton");

    // back to main menu
    SceneManager.LoadScene(WorkerMenu.SceneNameMenu);
}

public void OnPhotonInstantiate(PhotonMessageInfo info)
{
    Debug.Log("OnPhotonInstantiate " + info.sender);    // you could use this info to
store this or react
}

public void OnPhotonPlayerConnected(PhotonPlayer player)
{
    Debug.Log("OnPhotonPlayerConnected: " + player);
}

public void OnPhotonPlayerDisconnected(PhotonPlayer player)
{
    Debug.Log("OnPlayerDisconnected: " + player);
}

public void OnFailedToConnectToPhoton()
{
    Debug.Log("OnFailedToConnectToPhoton");

    // back to main menu
    SceneManager.LoadScene(WorkerMenu.SceneNameMenu);
}
}

```


Скрипт для конфигураций:

```
using UnityEngine;
public class GameConfig
{
    // VIDEO \\
    public int displayMode = 0;
    public int targetDisplay = 0;
    public int resolutionId = 0;
    public int graphicsQuality = 0;
    public int antialiasing = 0;
    public int vsync = 0;
    public bool toggleTest = false;

    // AUDIO \\
    public float masterVolume = 1f;
    public float musicVolume = 0.5f;
    public float effectsVolume = 0.5f;
    public float voiceVolume = 0.5f;
    public float micVolume = 0.5f;
    public bool soundBackground = true;

    // GAME \\
    public float horizontalSensitivity = 1f;
    public float verticalSensitivity = 1f;
    public int difficulty = 0;
    public int language = 0;
    public bool tips = true;

    // INPUT \\
    public string forward = "W";
    public string back = "S";
    public string left = "A";
    public string right = "D";
    public string crouch = "LeftControl";
    public string jump = "Space";
}
```

Скрипт для загрузки уровня:

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.Events;
using System.IO;
using UnityEngine.SceneManagement;

public class LoadLevel : MonoBehaviour
{
    public int Level;
    private bool ready_b = false;

    // Start is called before the first frame update
    void Start()
    {
    }
}
```

```

// Update is called once per frame
void Update()
{

}
private void PressedButton ()
{
    if (Input.GetKeyDown(KeyCode.E))
    {
        ready_b = true;
    }
}

private void OnTriggerEnter2D(Collider2D collision)
{
    if ((collision.tag == "Player"))
    {
        SceneManager.LoadScene(Level);
        //Application.LoadLevel(Level);
    }
}
}

```

ПРИЛОЖЕНИЕ Г

Пользовательское соглашение

МЫ НАСТОЯТЕЛЬНО РЕКОМЕНДУЕМ ВАМ ВНИМАТЕЛЬНО ОЗНАКОМИТЬСЯ С ЭТИМИ УСЛОВИЯМИ. УСТАНОВИВ ДАННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, ВЫ СОГЛАШАЕТЕСЬ С ОБЯЗАТЕЛЬСТВАМИ, ПРЕДУСМОТРЕННЫМИ ЭТИМ ДОКУМЕНТОМ. ЕСЛИ ВЫ НЕ СОГЛАСНЫ СО ВСЕМИ УСЛОВИЯМИ, ВЫ ДОЛЖНЫ ОТКАЗАТЬСЯ ОТ УСТАНОВКИ.

Чепокон Елизар Сергеевич благодарит вас за загрузку и установку данного программного обеспечения. Этот документ является Пользовательским соглашением ("Соглашение"), которое содержит условия, регулирующие доступ к нашему программному обеспечению и его использование.

Принимая условия настоящего Соглашения, вы подтверждаете, что являетесь дееспособным в соответствии с законодательством места вашего проживания, вам не запрещается получать игровые услуги в Интернете, и в законодательстве вашей страны нет других ограничений, в том числе возрастных ограничений для пользования такими услугами. Для лиц младше возраста совершеннолетия (правоспособность), Соглашение считается заключенным с их законными представителями (родителями, опекунами).

Основные термины и определения

"Веб-сайт" - совокупность, размещенных в сети Интернет веб-страниц, объединенных общей темой и / или дизайном, и единым адресным пространством домена survarium.com, предоставляющий Пользователю доступ к Сервису, в том числе для участия в Игре.

"Форум" - часть Веб-сайта, предназначенная для дополнительной связи пользователей и обсуждения событий, связанных с Игрой.

"Сервис" - совокупность таких компонентов, как Веб-сайт, Игра, функциональность, контент, программное обеспечение и услуги, предлагаемые через Веб-сайт.

"Игра" - многопользовательская ролевая игра от третьего лица (RPG) "Untitled game", которая представляет собой компьютерную программу, включающую

клиентскую часть Игры, компоненты, функции, игровые ресурсы, включая все дополнения и обновления к игре, а также сервер часть программного обеспечения Игры. Участие пользователя в Игре осуществляется в интерактивном режиме (онлайн), путем установки Клиентской части Игры на персональном компьютере пользователя и путем подключения Пользователя через Интернет к игровым ресурсам Компании. Принцип функционирования Игры основывается на общеизвестной модели Free-To-Play (бесплатная игра), что означает предоставление Пользователю права использовать Игру путем участия в Игре без внесения абонентской платы и без каких-либо иных обязательных платежей, которые необходимы для принятия участия в Игре и/или продвижения в игровом процессе. Компания обладает необходимыми правами на Игру, на все ее элементы, взятые отдельно и в целом.

"Клиентская часть Игры" - это программное обеспечение, которое требуется для вашего участия в Игре и должно быть установлено на вашем персональном компьютере. Клиентская часть Игры устанавливается на ваш персональный компьютер самостоятельно. Клиентская часть Игры распространяется Компанией и / или уполномоченными ею лицами посредством сети Интернет.

"Материалы" - Веб-сайт, Игра, Программное обеспечение и любая часть Сервиса, компоненты, контент, включая, но не ограничиваясь, любые Аккаунты Пользователей, названия, компьютерные коды, файлы, игровое программное обеспечение, клиентское и серверное программное обеспечение, аудиодорожки, обновления, картинки, темы, объекты, персонажи, имена, слоганы, местоположение, концепция, иллюстрации, рисунки, графика, изображения, видео, анимация, звуки, музыка, композиции, звукозаписи, аудиовизуальные эффекты, информация, данные, документация, апплеты, протоколы чатов, информация о персонажах, игровой процесс, сценарии, записи.

"Политика конфиденциальности" - неотъемлемая часть Соглашения, регулирующего предоставление, обработку и использование персональных данных Пользователя.

"Бета-версия" - версия программного продукта до коммерческого выпуска программного продукта. Предоставляется на особых условиях для тестирования и выявления ошибок.

"Законодательство" - законодательство страны, являющейся страной регистрации (учреждения) Компании, а также законодательством страны гражданства (регистрации) или постоянного места нахождения Пользователя.

"Правила форума" - неотъемлемая часть Соглашения, регулирующего правила участия и поведения Пользователя на Форуме, ограничения на действия Пользователя и ответственность Пользователя за нарушение этих правил и ограничений.

"Пользователь" и слова "вы / ваш" относятся к физическому лицу, посетителю Веб-сайта и / или игроку игры. Пользователь является стороной Соглашения.

1. Общие положения

1.1. Настоящее Соглашение регулирует отношения между Компанией и Пользователем, это официальный документ, который определяет правила использования Веб-сайта, Сервиса, прав и обязанностей Компании, прав и обязанностей Пользователей Веб-сайта.

1.2. В тот момент, когда вы завершили свою регистрацию на Веб-сайте или в момент начала использования Сервиса, вы подтверждаете, что являетесь участником настоящего Соглашения, и что вы принимаете все условия без каких-либо оговорок и исключений, и принимаете их как имеющие юридическую силу. Вся деятельность Веб-сайта и информация об этом регулируется настоящим Соглашением.

1.3. Соглашение может быть изменено без предварительного уведомления. Новое Соглашение вступает в силу с момента его опубликования в Интернете, если иное не предусмотрено в пересмотренном Соглашении. Вы соглашаетесь регулярно читать условия Соглашения, чтобы иметь представление обо всех нововведениях, изменениях и дополнениях. Продолжая использовать любую часть Сервиса после внесения изменений в Соглашение, вы подтверждаете признание обновленных условий. Если вы не согласны с измененными условиями, вы должны прекратить использование Сервиса и / или его элементов.

1.4. Настоящее Соглашение не отменяет никаких дополнительных условий, которые могут применяться при использовании определенной части Сервиса. Если положение настоящего Соглашения противоречит одному или нескольким положениям любого другого соглашения, условиям, преимущество имеют положения настоящего Соглашения.

2. Описание взаимоотношений

2.1. Мы предоставляем зарегистрированным Пользователям доступ к Игре на основе модели "free-to-play", а также к связанным с ней компонентам, функциям и услугам, доступ к которым осуществляется через Веб-сайт, это может быть как форум сообщества, так и другие интерактивные службы. Чтобы войти в Игру, вам может потребоваться загрузить и / или установить Клиентскую часть Игры, что влечет за собой необходимость использования определенного оборудования и программного обеспечения, а также доступа к Интернету.

2.2. Мы предоставляем вам бесплатный базовый доступ к Сервису, но некоторые дополнительные функции и компоненты Игры или Сервиса в целом, направленные на улучшение игрового процесса, могут оставаться недоступными до тех пор, пока вы не выполните определенные условия для предоставления доступа к функциям и компонентам.

2.3. Принимая настоящее Соглашение, вы подтверждаете и понимаете, что любые дополнительные функции и компоненты являются добавочными и связаны с доступом к Сервису. Вы не можете их использовать, и ваши права на использование таких функций будут ограничены, если по какой-либо причине вы не сможете получить доступ к Сервису. Вы соглашаетесь с тем, что Компания или связанные с ней стороны не могут нести ответственность за отсутствие доступа к функциям и компонентам и не обязаны возмещать ущерб или убытки, понесенные по этой причине, и вы понимаете, что несете единоличную ответственность за соблюдение всех требований, необходимых для использования Сервиса, а также дополнительных функций и компонентов.

2.4. Вы соглашаетесь с тем, что мы не несем ответственность за оборудование, программное обеспечение или ваш доступ к Интернету, скорость платежей, которые возникают при получении прав на использование премиальных функций и компонентов Сервиса. В некоторых случаях для использования определенной части В противном случае элементы таких Сервисов не будут доступны вам.

3. Создание и использование Аккаунта

3.1. Чтобы вы могли использовать некоторые ресурсы Веб-сайта, вы должны пройти процесс регистрации, в результате которого будет создан ваш Аккаунт.

3.2. Чтобы создать свой Аккаунт, вам должно быть не менее 13 лет. Если вы достигли 13-летнего возраста, но не достигли совершеннолетия для подписания контрактов в юрисдикции, в которой вы проживаете и / или возраста 18 лет, важно, чтобы вы прочитали Соглашение в присутствии своих родителей или законных опекунов / попечителей.

3.3. При создании Аккаунта вы соглашаетесь предоставлять достоверную, точную, актуальную и полную информацию, которая должна быть введена в соответствующие поля, и поддерживать ее актуальность, достоверность, точность и полноту.

3.4. После того, как вы создали Аккаунт, вы несете ответственность за него. Если характеристики Аккаунта нарушают ту или иную часть условий настоящего Соглашения, мы можем немедленно отключить его без предварительного уведомления или вашего согласия. В исключительных случаях, когда официальное Соглашение не нарушается, но, на наш взгляд, имя Аккаунта необходимо изменить, мы можем сделать это без вашего явного согласия.

3.5. Вы соглашаетесь и понимаете, что вы несете ответственность за сохранение конфиденциальности паролей, связанных с Аккаунтом, который вы используете для доступа к Сервису. Вы будете нести исключительную ответственность перед Компанией за все действия, которые происходят в вашем Аккаунте. Если вам стало известно о любом несанкционированном использовании вашего Аккаунта, вы соглашаетесь немедленно уведомить нашу службу поддержки Пользователей.

3.6. Мы можем прекратить действие вашего Аккаунта по нашему собственному усмотрению, если вы его не использовали в течение двенадцати месяцев после даты его создания или в течение непрерывного периода в двенадцать месяцев. По окончании действия вашего Аккаунта мы уведомим вас по электронной почте. В течение одного месяца после получения уведомления вы можете сообщить нам о своем намерении сохранить свой Аккаунт, и по нашему собственному усмотрению мы не можем его удалить.

4. Право собственности. Право интеллектуальной собственности

4.1. Компания владеет правами на интеллектуальную собственность на Веб - сайт, Форум, Игру и другие компоненты Сервиса (включая, но не ограничиваясь,

любые Аккаунты пользователей, названия, компьютерные коды, файлы, игры, программное обеспечение, клиентское и серверное программное обеспечение, инструменты, звуковые дорожки, обновления, темы, объекты, персонажи, их имена, слоганы, местоположение, концепция, иллюстрации, рисунки, графика, изображения, видео, анимация, звуки, музыка, композиции, звуковые записи, аудиовизуальные эффекты, информацию, данные, документацию, апплеты, протоколы чатов, информацию о персонажах, игровом процессе и сценарии, записи, вопросы, связанные с игрой, действия в игре), которые защищены действующим законодательством, включая законы об авторском праве.

4.2. Вы можете использовать элементы Веб - сайта только для личного, некоммерческого использования, при условии сохранения всех знаков охраны авторских прав, смежных прав, торговых марок (товарных знаков, знаков для товаров и услуг), в том числе при условии получения согласия от нашей службы поддержки Пользователей.

4.3. Вы гарантируете, что все права на интеллектуальную собственность на материалы, размещенные вами на Веб - сайте, принадлежат вам на законных основаниях. Вы несете полную ответственность за размещение материалов, распространение которых запрещено законом.

4.4. Вы гарантируете, что в случае претензий к Компании и / или судебных процессов, связанных с нарушением гарантий, предусмотренных в настоящем Соглашении, вы берете на себя обязательства по их урегулированию от вашего имени и несете ответственность за эти претензии и / или судебные процессы самостоятельно и за свой счет. Если судебное разбирательство приведет к удовлетворению претензий и / или судебных процессов третьих сторон против нас, вы соглашаетесь возместить все убытки, понесенные нами, и другие расходы, вызванные таким нарушением.

4.5. Если вы планируете переделывать, модифицировать или создавать производные работы любого рода на основе наших Материалов, которые являются нашей собственностью, вы соглашаетесь с тем, что у вас нет прав на результаты таких изменений и что все права на них принадлежат исключительно Компании.

4.6. Вы передаете нам на безвозмездной основе с момента создания без каких-либо ограничений и компенсаций все права, которые вы можете иметь на результаты таких изменений и переработок. Эта передача включает, но не ограничивается, авторские права на территории стран всего мира на результаты таких изменений и право быть истцом в случаях нарушений, совершенных в прошлом или, которые будут совершены в будущем.

4.7. Принимая настоящее Соглашение, вы предоставляете Компании эксклюзивные, бессрочные, действующие по всему миру, абсолютные права создавать, использовать, продавать, предлагать для продажи, импорта или использования иным образом, публикации, цитирования, адаптации, перевода, хранения, воспроизведения, создавать производные работы, интегрировать, лицензировать, печатать, сублицензировать, распространять, транслировать, передавать в прямой эфир или любой другой способ публичной демонстрации результатов таких изменений или любой их части любым способом в любой форме без уведомления, вознаграждения или компенсации для вас. Вы отказываетесь и соглашаетесь не требовать никаких прав, которые могут возникнуть у вас относительно результатов таких изменений, если это возможно в соответствии с применимым законом.

5. Другие сайты и сторонний контент

5.1. Веб-сайт может содержать ссылки на другие сайты, статьи, фотографии, иллюстрации, графику, информацию и другой контент, принадлежащий третьим лицам.

5.2. Мы не несем ответственности за информацию, размещенную на сторонних сайтах, к которым у вас есть доступ через Веб-сайт.

5.3. Ссылка на любой веб-сайт, продукт, услугу или любую информацию коммерческого или некоммерческого характера, размещенную на Веб-сайте, не означает, что мы одобряем или рекомендуем эти продукты и услуги с нашей стороны.

5.4. Если вы обращаетесь к сторонним сайтам, вы делаете это на свой страх и риск, и это Соглашение не действует для сторонних сайтов.

6. Права и обязанности Компании

6.1. Компания имеет право:

6.1.1. изменять дизайн Веб-сайта и / или его содержимого в любое время с уведомлением пользователей или без него;

6.1.2. удалять без каких-либо оснований и без уведомления любой контент, Материалы, комментарии, любую информацию, которая по нашему усмотрению нарушает и / или может нарушать действующее законодательство, это Соглашение, Политику конфиденциальности, Правила Форума, права других Пользователей или третьих лиц, или наносит им вред, или угрожает их безопасности;

6.1.3. удалить ваш Аккаунт в случаях, предусмотренных разделом 9 настоящего Соглашения;

6.1.4. отправлять сообщения (в том числе посредством электронной почты, смс и т. д.), которые содержат рекламную информацию о Сервисе, Игре или материалах, тематически связанных с ними;

6.1.5. полностью или частично уменьшить функциональность Веб - сайта, Материалов по техническим, превентивным или иным причинам по нашему собственному усмотрению.

6.1.6. использовать ваши данные в соответствии с Политикой конфиденциальности, размещенной на Веб - сайте.

6.2. Компания обязана:

6.2.1. соблюдать настоящее Соглашение, Политику конфиденциальности, Правила Форума.

7. Права и обязанности Пользователя

7.1. Пользователь имеет право:

7.1.1. настроить Аккаунт, изменить имя пользователя и пароль для доступа к нему;

7.1.2. публиковать, добавлять и редактировать свой Аккаунт и информацию в профиле Пользователя о себе;

7.1.3. публиковать сообщения на Форуме;

7.1.4. осуществлять иные, не запрещенные законом или настоящим Соглашением, действия, касающиеся вашего использования Веб - сайта или Сервиса.

7.2. Пользователь обязан:

- 7.2.1. указать достоверную информацию на момент регистрации;
- 7.2.2. принять необходимые меры для обеспечения конфиденциальности учетных данных (логина и пароля), используемых для доступа к Аккаунту;
- 7.2.3. не использовать Веб-сайт в коммерческих целях без предварительного разрешения Компании;
- 7.2.4. соблюдать настоящее Соглашение, Политику конфиденциальности, Правила Форума;
- 7.2.5. не использовать какое-либо программное обеспечение, ошибки, дефекты в целях получения несанкционированного доступа к Сервису или получения неконкурентных преимуществ, в том числе над другими пользователями;
- 7.2.6. не предпринимать следующие действия:
 - (а) продавать, перепродавать, передавать, отдавать в кредит, использовать в любой другой форме Сервис, Веб-сайт или Материалы, принадлежащие Компании, с коммерческой целью;
 - б) распространять, публично исполнять или публично показывать любые Материалы, принадлежащие Компании, за исключением обычного использования Игры;
 - в) изменять, адаптировать, модифицировать, улучшать Игру, содержимое Игры или другие Материалы, принадлежащие Компании, или их части;
 - г) использовать программное обеспечение для сбора или удаления информации;
 - д) загружать (кроме кеша отображения страницы) любую часть Материалов, принадлежащих Компании, или любую содержащуюся в них информацию, за исключением случаев, явно предусмотренных предполагаемой функцией Веб-сайта;
 - е) использовать Сервис или Материалы способами, которые отличаются от традиционно принятых;
 - ё) вводить в заблуждение других Пользователей относительно вашей личности;
 - ж) копировать, переводить, публиковать, декомпилировать, выполнять технический анализ, дизассемблировать или конвертировать Веб-сайт (исходный код Веб-сайта) или осуществлять попытку получить доступ к исходному коду Веб-

сайта для создания производных работ на основе исходного кода Веб-сайта или иным образом;

з) загружать, хранить, публиковать, распространять, предоставлять доступ или иным образом использовать любую информацию, инструменты и / или материалы, которые:

1. содержат угрозы, дискредитируют, оскорбляют, порочат личную или деловую репутацию, или нарушают конфиденциальность других Пользователи или третьих лиц;

2. являются вульгарными или непристойными, содержат нецензурную лексику или порнографические изображения и тексты или сцены сексуального характера с участием несовершеннолетних;

3. содержат сцены насилия или бесчеловечного обращения с животными;

4. пропагандируют и / или разжигают расовую, религиозную, этническую ненависть или вражду, или защищают идеологию фашизма, коммунизма или расового превосходства, содержат отрицательные заявления о любой стране / народе / религии / политике, унижают национальное достоинство, в том числе по признаку национального происхождения или политических убеждений;

5. содержат насильственные материалы;

6. содействуют преступной деятельности или содержат советы, инструкции или указания о совершении преступных действий;

7. содержат ограниченную информацию;

8. содержат рекламу;

9. содержат мошенническую информацию;

10. содержат личные данные, изображения третьих лиц без их предварительного согласия, за исключением случаев, предусмотренных законодательством;

11. содержат вирусы или другие компьютерные коды, файлы или программы, предназначенные для прерывания, уничтожения или ограничения функциональности любого компьютерного или телекоммуникационного оборудования или программ для несанкционированного доступа, а также серийные номера для коммерческих программных продуктов и программ, которые генерируют

их, пароли и другие способы получения несанкционированного доступа к платным ресурсам в Интернете, а также размещение ссылок на такую информацию;

12. нарушают нормальную работу Веб - сайта и / или Сервиса;

13. нарушают права интеллектуальной собственности Компании и / или третьих лиц.

8. Бета тестирование

8.1. После подачи заявки на участие в бета-тестировании вам может быть предоставлена такая возможность. Ваше участие в качестве бета-тестера должно соответствовать нижеприведенным условиям.

8.2. Принимая участие в бета-тестировании, вам известно о том, что компоненты Игры и / или Веб-сайта могут содержать определенные ошибки, могут не работать должным образом или не выполнять все необходимые функции и / или могут противоречить функционированию другого программного обеспечения и может вызывать ошибки, потерю данных или другие трудности.

8.3. Вы проведете тестирование лично и не будете предоставлять доступ к бета-версии Игры любому другому лицу.

8.4. Мы приглашаем вас играть в Игру как бета-тестер, только для оценки характеристик Игры и обнаружения ошибок. Никакие положения, содержащиеся в настоящем Соглашении или не представленные на Веб - сайте или через Сервис, не могут быть истолкованы как те, которые предоставляют вам какое-либо право или привилегию любого рода в отношении последующих версий Игры или контента, которые вы найдете.

8.5. Бета-версия Игры предоставляется для тестирования "как есть". Мы не даем вам абсолютно никаких гарантий относительно Игры. Вам известно, что вы играете в бета-версию Игры на свой страх и риск.

8.6. Вы подтверждаете, что бета-версия Игры может включать известные или неизвестные ошибки и что мы не берем на себя никаких обязательств относительно бета-версии Игры. Играя в бета-версию Игры, вы имеете право зарабатывать в Игре валюту (золото, серебро, очки опыта, экипировку или другие игровые достижения).

8.7. Вы должны знать, что такие данные могут быть в любой момент сброшены во время процесса тестирования, когда Игра полностью пройдет определенный этап тестирования. В этом случае все файлы журналов и данные

Пользователя могут быть удалены, и Пользователь вернется к исходному состоянию.

9. Основания для удаления Аккаунта

9.1. Мы имеем право заблокировать или удалить ваш Аккаунт в следующих случаях и по нашему собственному усмотрению без предварительного уведомления или согласия пользователя.

Основные причины для удаления или блокировки вашего Аккаунта:

- вы предоставили неверную информацию о себе или у нас есть основания полагать, что предоставленная вами информация является неполной или некорректной;

- вы нарушили какое-либо положение Соглашения, Политики конфиденциальности, Правил форума или Правил магазина премиум-класса;

- вы не использовали свой Аккаунт в течение 12 месяцев подряд.

9.2. Вы можете закрыть свой Аккаунт, отправив нам уведомление. Такие действия будут рассматриваться как прекращение действия настоящего Соглашения, и Пользователь соглашается больше не использовать Веб-сайт и / или связанные услуги.

9.3. Если действие вашего Аккаунта прекращено, вы не получаете никакой компенсации. Мы не конвертируем валюту в Игре наличными, и мы не предоставляем никакой другой компенсации. Ваш доступ к вашему Аккаунту будет прекращен.

10. Ответственность

10.1. Мы не несем ответственность за возможные сбои и перерывы в работе Веб - сайта и / или Сервиса и потерю информации в результате. Мы не несем ответственность за какой-либо ущерб вашему компьютеру, мобильным устройствам, любому другому оборудованию или программному обеспечению Пользователей.

10.2. Мы не несем ответственность за любые скомпрометированные пароли третьих лиц и любые действия, совершенные с использованием вашего Аккаунта.

10.3. Вы несете ответственность за любые незаконные действия, которые происходят в вашем Аккаунте, а также в связи с размещением Материалов с использованием вашего Аккаунта на Веб-сайте.

10.4. Вы несете личную ответственность за любую материальную или другую информацию, которую вы загружаете на Веб-сайт или иным образом публикуете через Сервис.

10.5. Вы готовы возместить Компании или любой третьей стороне любой ущерб, причиненный в результате ваших действий, включая, помимо прочего, нарушение настоящего Соглашения, прав интеллектуальной собственности или других прав.

11. Ограничение ответственности

Ни при каких обстоятельствах мы не несем ответственность за какие-либо прямые, косвенные или случайные убытки, включая, но не ограничиваясь, упущенную выгоду, простой, потерю данных в связи с использованием или невозможностью использования Сервиса или любого другого контента, а также любые функции, компоненты, услуги или материалы третьих сторон, которые представлены Сервисом или через него, включая, но не ограничиваясь, любые убытки, возникшие в результате или вызванные использованием информации, полученной от нас или наших связанных сторон или партнеров или в результате ошибок, задержек, вирусов, дефектов или задержек при передаче или сбое, независимо от того, связаны ли они со сбоем, кражей или несанкционированным доступом к Сервису.

12. Отказ от гарантий

12.1. Вы понимаете и соглашаетесь с тем, что Сервис и его элементы предоставляются на основе "как есть" без каких-либо гарантий.

12.2. Компания, дочерняя компания Компании, наши сотрудники, акционеры, связанные стороны, сотрудники или агенты отказываются от всех других гарантий, в том числе, но не ограничиваясь:

- гарантиями КОММЕРЧЕСКОЙ ЦЕННОСТИ, пригодности;
- гарантиями соответствия вашим ожиданиям;
- гарантиями отсутствия ошибок, вирусов или других вредоносных компонентов или дефектов;
- гарантиями бесперебойности и регулярности работы.

13. Заключительные положения

13.1. Настоящее Соглашение вступает в силу со дня его принятия и действует в течение неопределенного времени.

13.2. В случае нарушения условий настоящего Соглашения Компания имеет право отказаться от исполнения настоящего Соглашения в одностороннем порядке.

13.3. Вопросы, не урегулированные в настоящем документе, определяются в соответствии с законодательством России. Споры, связанные с заключением, исполнением и прекращением действия настоящего Соглашения, подлежат рассмотрению соответствующим судом России в соответствии с материальным и процессуальным правом России.

13.4. Если окончательное решение суда компетентной юрисдикции заявляет, что какое-либо из положений настоящего Соглашения является недействительным, остальная часть Соглашения остается в силе.

13.5. Права и обязанности Сторон, вытекающие из формулировок предыдущих версий Соглашения, которые в настоящее время недействительны, рассматриваются в соответствии с положениями активной (фактической) редакцией Соглашения, если из характера отношений не вытекает иное между Сторонами.

БЛАГОДАРИМ ВАС, ЧТО НАШЛИ ВРЕМЯ ОЗНАКОМИТЬСЯ С ДАННЫМ СОГЛАШЕНИЕМ. ЕСЛИ ВЫ НЕ СОГЛАСНЫ С УСЛОВИЯМИ НАСТОЯЩЕГО СОГЛАШЕНИЯ, ПОЖАЛУЙСТА, НЕМЕДЛЕННО ПОКИНЬТЕ ВЕБ-САЙТ. ЕСЛИ У ВАС ЕСТЬ ВОПРОСЫ ИЛИ ПРЕДЛОЖЕНИЯ - ПОЖАЛУЙСТА, ОБРАЩАЙТЕСЬ К СЛУЖБЕ ПОДДЕРЖКИ: bloodiesco@yandex.ru.

ПРИЛОЖЕНИЕ Д

Техническое задание

1. Введение

1.1. Наименование программы

«Реализация многопользовательской компьютерной игры в жанре РПГ».

1.2. Краткая характеристика области применения

Программный продукт представляет собой компьютерную многопользовательскую ролевую игру. Данная система создается как развлекательное программное обеспечение для юридических или частных лиц.

2. Основания для разработки

Программа создается в рамках написания курсовой работы по теме «Реализация многопользовательской компьютерной игры в жанре РПГ» (факультет экономики, менеджмента и бизнес-информатики), в соответствии с учебным планом подготовки бакалавров по направлению подготовки 09.03.04 «Программная инженерия». Приказ № 8.2.2.1-38-25/04 от 07.10.2019.

3. Назначение разработки

Данный проект является компьютерной игрой, вследствие чего предусматривается одна категория пользователей - игроки. В процессе работы приложения пользователь является непосредственным участником игрового процесса и оказывает непосредственное влияние на него.

Эксплуатационное назначение: программный продукт должен эксплуатироваться в развлекательных целях.

4. Требования к программе

4.1. Требования к функциональным характеристикам

4.1.1. Состав выполняемых функций:

Графический функционал:

1. Выбор разрешения экрана;
2. Выбор качества графики;
3. Выбор полноэкранного или оконного режима;

Звуковой функционал:

1. Регулировка общей громкости;

2. Регулировка громкости музыки;
3. Регулировка громкости внутриигровых звуков;

Внутриигровой функционал:

1. Система поиска пути;
2. Система взаимодействия игровых объектов;
3. Боевая система;

Интерфейс пользователя:

1. Переходные сцены (вступительная, финальная, экран загрузки);
2. Главное меню;
3. Графический интерфейс пользователя.

4.1.2. Организация входных и выходных данных:

Входными данными в компьютерной игре являются игровые настройки пользователя (разрешение экрана, громкость звуков, качество графики и т.д.), а также непосредственное управление во время игрового процесса с помощью компьютерной мыши и клавиатуры. Проект относится к играм в реальном времени, где в отличие от пошаговых - действия игрока незамедлительно оказывают влияние на игровой процесс.

Выходными данными являются графическая интерпретация игрового процесса на мониторе игрока и звук, сопровождающий его. Действия игрока влияют на игровой процесс и текущее состояние игровой сцены. Игрок контролирует игрового персонажа с помощью интерфейса пользователя.

4.2. Требования к обеспечению надежного функционирования программы

В программе должна присутствовать проверка входной информации на соответствие типов, принадлежность диапазону допустимых значений и соответствие структурной корректности. В случае возникновения ошибок предусмотреть возможность вывода информативных диагностических сообщений. В программе реализована система логов, а также возможность анализа дампа приложения в случае некорректного завершения.

5. Условия эксплуатации

Для управления системой необходима установка устройств и подключение их к контроллеру мастером. Для эксплуатации после установки достаточно мобильного

телефона с доступом к сети «Интернет» или любого другого устройства с доступом к сети «Интернет».

5.1. Требования к составу и параметрам технических средств

Минимальные системные требования:

- ОС (операционная система): Windows XP/Vista/7/8;
- Процессор: Intel Core 2 Duo @ 3.0 Ghz / AMD Athlon 64 X2 6000+;
- Оперативная память: 1 Gb;
- Жесткий диск: 10 Gb свободно;
- Видео память: 512 Mb;
- Видео карта: nVidia GeForce 9800 / AMD Radeon HD 4870;
- Звуковая карта: Совместимая с DirectX;
- DirectX 9.0c;
- Клавиатура, Мышь.

Рекомендуемые системные требования:

- ОС (операционная система): Windows Vista/7/8;
- Процессор: Intel Core i5 @ 3.2 GHz / AMD Phenom II X4 @ 3.6 GHz;
- Оперативная память: 2 Gb;
- Жесткий диск: 10 Gb свободно;
- Видео память: 1 Gb;
- Видео карта: nVidia GeForce GTX 460 / AMD Radeon HD 5850;
- Звуковая карта: Совместимая с DirectX;
- DirectX 11;
- Клавиатура, Мышь.

5.2. Требования к информационной и программной совместимости

5.2.1. Требования к информационным структурам и методам решения

Программа должна функционировать под управлением ОС семейства Windows следующих версий: Windows XP, Vista, 7, 8.

5.2.2. Требования к исходным кодам и языкам программирования

В приложении используются библиотеки платформы.NET Framework.

5.2.3. Требования к программным средствам, используемым программой

Требуется установленный DirectX 9.0с или более поздней версии.

5.3. Требования к маркировке и упаковке

Требования к маркировке и упаковке не предъявляются.

5.4. Требования к транспортированию и хранению

Специальные требования к транспортировке и хранению не предъявляются.

6. Требования к программной документации

Требования устанавливаются на основе правил написания и оформления курсовых работ студентов образовательной программы бакалавриата «Программная инженерия» по направлению подготовки 09.03.04 Программная инженерия, Пермь 2019.

7. Техничко-экономические показатели

В рамках данной работы расчёт экономической эффективности не предусмотрен.

8. Стадии и этапы разработки

Предусматривается:

1. Анализ
2. Проектирование
3. Разработка
4. Тестирование

9. Порядок контроля и приемки

9.1. Виды испытаний

Производится проверка корректного выполнения программой заложенных в нее функций.

9.2. Общие требования к приемке работы

Общие требования к проверке работ указаны в правилах написания и оформления курсовых работ студентов образовательной программы бакалавриата «Программная инженерия» по направлению подготовки 09.03.04 Программная инженерия, Пермь 2019.

ПРИЛОЖЕНИЕ Е

Руководство разработчика

1. Общие сведения

1. Документация: программная.
2. Предмет: многопользовательская компьютерная игры в жанре РПГ.
3. Аудитория: C# программисты, программисты Unity.
4. Задачи: обеспечить программисту возможность решать задачи, которые могут быть перед ним поставлены в отношении программы.
5. Стандарты: ГОСТ 19.504-79

2. Цели и задачи

Программный продукт по своему основному назначению является многопользовательской компьютерной ролевой игрой, разрабатываемый для развлечения.

3. Назначение и условия применения программы

3.1. Назначение программы

Программный продукт создан для развлечения.

3.2. Функции, выполняемые программой

Программа работает с удаленным брокером по принципу издатель-подписчик MQTT протокола. Программный продукт позволяет изменять, добавлять или удалять данные устройств, подключенных к нему.

3.3. Условия, необходимые для выполнения программы

Минимальные системные требования:

- ОС (операционная система): Windows XP/Vista/7/8;
- Процессор: Intel Core 2 Duo @ 3.0 Ghz / AMD Athlon 64 X2 6000+;
- Оперативная память: 1 Gb;
- Жесткий диск: 10 Gb свободно;
- Видео память: 512 Mb;
- Видео карта: nVidia GeForce 9800 / AMD Radeon HD 4870;
- Звуковая карта: Совместимая с DirectX;
- DirectX 9.0c;
- Клавиатура, Мышь.

Рекомендуемые системные требования:

- ОС (операционная система): Windows Vista/7/8;
- Процессор: Intel Core i5 @ 3.2 GHz / AMD Phenom II X4 @ 3.6 GHz;
- Оперативная память: 2 Gb;
- Жесткий диск: 10 Gb свободно;
- Видео память: 1 Gb;
- Видео карта: nVidia GeForce GTX 460 / AMD Radeon HD 5850;
- Звуковая карта: Совместимая с DirectX;
- DirectX 11;
- Клавиатура, Мышь.

4. Характеристика программы

4.1. Описание основных характеристик программы

Скорость работы игры зависит от характеристик персонального компьютера.

4.2. Особенности программы

Программа должна функционировать под управлением ОС семейства Windows следующих версий: Windows XP, Vista, 7, 8.

5. Обращение к программе

Запуск программы осуществляется после установки и включения программы пользователем.

6. Входные и выходные данные

Входные данные производятся с помощью клавиатуры и мыши.

7. Сообщения

При возникновении ошибок система оповещает о причине ошибки.