ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук

Департамент программной инженерии

СОГЛАСОВАНО	УТВЕРЖДАЮ
Научный руководитель, приглашенный преподаватель департамента программной инженерии	Академический руководитель образовательной программы «Программная инженерия», кандидат технических наук
_____ Н.И. Веселко	_____ В.В. Шилов
«15» апреля 2022 г.	«15» апреля 2022 г.

Подп. и дата	
№ Инв. № дубл.	
Взам. инв.	
Подп. и дата	
Инв. № подл	

VR-ИГРА «ДИЗАЙНЕР ПЛЯЖЕЙ» НА ДВИЖКЕ UNREAL ENGINE 4

Техническое задание

ЛИСТ УТВЕРЖДЕНИЯ

RU.17701729.05.03-01 ТЗ 01-1-ЛУ

Исполнитель

студентка группы БПИ198

_____ / М. А. Шалаева /

«15» апреля 2022 г.

Москва 2022

УТВЕРЖДЕН

RU.17701729.05.03-01 ТЗ 01-1-ЛУ

Подп. и дата	
№ Инв. № дубл.	
Взам. инв.	
Подп. и дата	
Инв. № подл	

VR-ИГРА «ДИЗАЙНЕР ПЛЯЖЕЙ» НА ДВИЖКЕ UNREAL ENGINE 4

Техническое задание

RU.17701729.05.03-01 ТЗ 01-1-ЛУ

Листов 32

Москва 2022

Содержание

АННОТАЦИЯ **4**

1 ВВЕДЕНИЕ **6**

1.1 Наименование программы 6

1.2 Краткая характеристика области применения программы 6

2 ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ **7**

3 НАЗНАЧЕНИЕ РАЗРАБОТКИ **8**

3.1 Функциональное назначение 8

3.2 Эксплуатационное назначение 8

4 ТРЕБОВАНИЯ К ПРОГРАММЕ **9**

4.1 Требования к функциональным характеристикам 9

4.1.1 Требования к составу выполняемых функций 9

4.1.2 Организация входных данных 17

4.1.3 Организация выходных данных 18

4.2 Требования к временным характеристикам 21

4.3 Требования к интерфейсу 21

4.4 Требования к надежности 22

4.5 Условия эксплуатации 22

4.5.1 Климатические условия эксплуатации 22

4.5.2 Требования к видам обслуживания 22

4.5.3 Требования к численности и квалификации персонала 22

4.5.4 Требования к окружающей среде 22

4.6 Требования к составу и параметрам технических средств 22

4.7 Требования к информационной и программной совместимости 23

4.7.1 Требования к исходным кодам и языкам программирования 23

4.7.2 Требования к программным средствам, используемым программой 23

4.7.3 Требования к защите информации и программ 23

4.8 Требования к маркировке и упаковке 23

4.9 Требования к транспортированию и хранению 23

5 ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ 24

5.1 Состав программной документации 24

5.2 Специальные требования к программной документации 24

6 ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ 25

6.1 Ориентировочная экономическая эффективность 25

6.2 Предполагаемая потребность 25

6.3 Экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами 25

7 СТАДИИ И ЭТАПЫ РАЗРАБОТКИ 26

8 ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ 28

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 29

ПРИЛОЖЕНИЕ 1 1

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ 2

АННОТАЦИЯ

Техническое задание – это основной документ, определяющий цель, набор требований, структуру, свойства, методы и порядок создания программного продукта, в соответствии с которым производится разработка программы, ее тестирование и приемка, исключающий двусмысленность толкования разными исполнителями.

Настоящее Техническое задание на разработку «VR-игры ‘‘Дизайнер пляжей’’ на движке Unreal Engine 4» содержит следующий ряд разделов:

1) «Введение»;
2) «Основания для разработки»;
3) «Назначение разработки»;
4) «Требования к программе»;
5) «Требования к программным документам»;
6) «Технико-экономические показатели»;
7) «Стадии и этапы разработки»;
8) «Порядок контроля и приемки»;
9) «Список использованных источников»;
10) приложения.

В разделе «Введение» указано наименование и краткая характеристика области применения программы.

В разделе «Основания для разработки» указан документ, на основании которого ведется разработка, и наименование темы разработки.

В разделе «Назначение разработки» указано функциональное и эксплуатационное назначение программного продукта.

Раздел «Требования к программе» содержит основные требования к функциональным характеристикам, к надежности, к условиям эксплуатации, к составу и параметрам технических средств, к информационной и программной совместимости, к маркировке и упаковке, к транспортировке и хранению, а также специальные требования.

Раздел «Требования к программным документам» содержит предварительный состав программной документации и специальные требования к ней.

Раздел «Технико-экономические показатели» содержит ориентировочную экономическую эффективность, предполагаемую годовую потребность, экономические преимущества разработки программы.

Раздел «Стадии и этапы разработки» содержит стадии разработки, этапы и содержание работ.

В разделе «Порядок контроля и приемки» указаны общие требования к приемке работы.

В разделе «Список использованных источников» указаны ссылки на источники, которые были использованы при разработке.

Настоящий документ разработан в соответствии с требованиями:

1) ГОСТ 19.101–77 Виды программ и программных документов [8];
2) ГОСТ 19.102–77 Стадии разработки [9];
3) ГОСТ 19.103–77 Обозначения программ и программных документов [10];
4) ГОСТ 19.104–78 Основные надписи [11];
5) ГОСТ 19.105–78 Общие требования к программным документам [12];
6) ГОСТ 19.106–78 Требования к программным документам, выполненным печатным способом [13];
7) ГОСТ 19.201–78 Техническое задание. Требования к содержанию и оформлению [14].

Изменения к техническому заданию оформляются согласно ГОСТ 19.603–78 [16], ГОСТ 19.604–78 [17].

Перед тем, как приступить к изучению данного документа, предполагается предварительное ознакомление с приведенной в Приложении 1 терминологией.

1 ВВЕДЕНИЕ

1.1 Наименование программы

Наименование темы разработки: «VR-игра ‘‘Дизайнер пляжей’’ на движке Unreal Engine 4»

Наименование темы разработки на английском языке: «‘‘Beach Designer’’ Unreal Engine 4 VR-game»

1.2 Краткая характеристика области применения программы

«VR-игра ‘‘Дизайнер пляжей’’ на движке Unreal Engine 4» – прикладная программа, разрабатываемая в развлекательных целях и представляющая собой симулятор дизайнера пляжей, занимающегося уборкой мусора, приобретением и расстановкой пляжного инвентаря.

Приложение предназначено для использования игроками, имеющими в распоряжении VR-очки Oculus Quest 2, в возрасте от 6 до 25 лет и направлено на удовлетворение эстетической потребности посредством декорирования территории, экологической потребности [20] посредством уборки и облагораживания территории, а также на формирование следующих мировоззренческих ценностей: важность сохранения окружающей среды, забота о ней, ответственное отношение к уборке мусорных отходов.

2 ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ

Основанием для разработки является учебный план подготовки бакалавров по направлению 09.03.04 «Программная инженерия» и утвержденная академическим руководителем тема курсового проекта.

Наименование темы разработки: «VR-игра ‘‘Дизайнер пляжей’’ на движке Unreal Engine 4».

Условное обозначение темы разработки: «Beach Designer».

3 НАЗНАЧЕНИЕ РАЗРАБОТКИ

3.1 Функциональное назначение

Разрабатываемое приложение дает пользователю возможность примерить на себя роль специалиста по благоустройству пляжей, который может взаимодействовать с объектами на картах, убирать территорию, покупать и расставлять пляжный инвентарь по своему усмотрению на нескольких пляжах Америки. За выполнение задания пользователь будет получать определенную сумму денежной валюты, которую можно будет использовать для приобретения более дорогих товаров на последующих уровнях.

3.2 Эксплуатационное назначение

Основными конечными потребителями разрабатываемого приложения являются люди в возрасте от 12 до 25 лет, которым хочется удовлетворить свою эстетическую потребность и которым интересно примерить на себя роль пляжного дизайнера.

Для корректного использования данной программы пользователю будут нужны очки виртуальной реальности Oculus Quest 2, ПК, отвечающий требованиям из пункта 4.6, и кабель Oculus Link для подключения очков к ПК (если не используется AirLink).

4 ТРЕБОВАНИЯ К ПРОГРАММЕ

4.1 Требования к функциональным характеристикам

4.1.1 Требования к составу выполняемых функций

1) Разрабатываемая программа должна визуализировать шесть игровых локаций:
   - офис;
   - локация, реализованная на основе пляжа парка Wai'anapanapa, Гавайи, США;
   - локация, реализованная на основе Papakōlea Green Sand Beach, Гавайи, США.

2) Офис — стартовая комната и связующее звено между всеми остальными локациями, перечисленными в пункте 1 данного списка (далее — пляжными уровнями). В офисе должны располагаться планшет и кот (см. пункт 8 списка). Из офиса должна происходить навигация между остальными уровнями с помощью планшета (см. пункт 10 списка).

3) Каждый пляжный уровень должен ставить перед пользователем задачу убрать территорию и украсить ее в определенной тематике.

4) Каждый пляжный уровень должен процедурно заполняться мусором во время первой загрузки. Мусорные объекты должны генерироваться в доступных для пользователя местах и должны находиться на расстоянии не менее 300 сантиметров друг от друга. Доступные места определяются границами перемещения пользователя.

   Пользователь должен иметь возможность поднять любой сгенерированный мусорный объект. В случае, если пользователь его выкинет, должна быть возможность вернуть все существующие на данный момент мусорные объекты (в том числе и тот, что был выкинут пользователем) на исходное место с помощью соответствующей кнопки в меню паузы.

   Каждый пляжный уровень включает в себя следующий перечень последовательных заданий:
   - Уборка мусора со всей доступной территории. В панели «Инструменты» → «Уборка» пользователь должен иметь возможность нажать на кнопку, создающую мусорный бак. Помимо этого должна быть кнопка, перемещающая мусорной бак к пользователю, а также кнопка уничтожения мусорного бака. Должна быть возможность брать мусорный бак в руки, перемещать, отпускать. Для утилизации мусорного объекта пользователю необходимо поместить мусорный объект в корзину (см. пункт 11 списка).

     По мере утилизации каждого мусорного объекта будет увеличиваться счетчик найденных объектов в штуках в формате «⟨счетчик⟩ / ⟨количество всех мусорных объектов на

уровне⟩» и в процентах в формате «⟨процент найденных объектов⟩%» и выводиться в панели «Инструменты» во вкладке «Уборка».

У каждого мусорного объекта должен быть тег CatFindable, чтобы кот мог его найти (см. пункт 8 списка);

- Декорирование уровня с помощью товаров, которые пользователь приобретает во вкладке «Магазин» из панели «Инструменты» (см. пункт 11 списка).

  Размер оплаты за конкретный уровень должен рассчитываться на основании того, насколько расставленные на уровне товары подходит к тематике задания. Определить, подходит товар или нет, можно по хештегам в его описании во вкладке «Магазин» в панели «Инструменты».

Описание всех пляжных уровней должно находиться в планшете пользователя. Описание последующих уровней и возможность начать их выполнение должны быть недоступны, пока не завершен предыдущий уровень.

5) Одна и та же локация может быть использована в разных уровнях с разными заданиями.

6) В офисе пользователь должен иметь возможность:
   - Кормить кота (см. пункт 8 списка);
   - Взаимодействовать с планшетом (см. пункт 10 списка).

   На любом пляжном уровне пользователь должен иметь возможность:
   - Передвигаться в пространстве с помощью джойстика на левом контроллере;
   - Телепортироваться в пространстве с помощью последовательного нажатия на джойстик на любом из контроллеров, его вращения и отпускания;
   - Поднимать и бросать предметы, с которыми можно взаимодействовать;
   - Открывать меню паузы путем нажатия на кнопку B правого контроллера (см. пункт 14 списка);
   - Входить в режим «отделки» путем нажатия на кнопку A правого контроллера и с его помощью взаимодействовать с предметами декора (см. пункт 12 списка);
   - Входить в режим «продажи» путем нажатия на кнопку X левого контроллера и с его помощью продавать предметы декора.

   На любом игровом уровне (как в офисе, так и на пляжном уровне) пользователь должен иметь возможность:
   - Поворачивать камеру на 45° с помощью джойстика на правом контроллере;

- Осматривать окружающую местность с помощью изменения поворота VR-очков в реальном мире;
- Изменять свои координаты в игровом пространстве в случае перемещения VR-очков в реальном мире;
- Открывать меню «Инструменты» путем нажатия на кнопку Y левого контроллера (см. пункт 11 списка);
- Гладить кота (см. пункт 8 списка).

7) Передвижение с помощью джойстика и телепортации должно быть ограничено допустимой зоной (Nav Mesh Bounds Volume [3]). Она должна быть как на берегу моря, так и на его дне, чтобы пользователь мог зайти в воду.

8) Разрабатываемое приложение должно реализовывать кота в качестве домашнего питомца пользователя:
   - Во время первого обучения программа должна предоставлять пользователю возможность выбрать одну из трех окрасок кота (белый с серым, черно-белый или рыжий). После этого функция изменения окраса кота должна быть недоступна;
   - Кот должен представлять из себя искусственный интеллект, поведение которого описывается с помощью Behavior Tree и Blackboard [1];
   - У кота должно быть три основных состояния поведения: «бездействие», «следование за пользователем» и «следование к месту объекта с тегом CatFindable»;
   - В состоянии бездействия в офисе кот должен уметь находиться в четырех состояниях: стоять, сидеть, спать и есть — и менять их в последовательности «стоять — сидеть — стоять — спать»; режим «есть» включается в случае, если пользователь дал коту еду, и тогда ключевая последовательность «стоять — сидеть — стоять — спать» должна быть прервана после перехода в состояние «стоять» и переключена на состояние «есть». На пляжных уровнях в состоянии бездействия кот должен уметь только стоять. Длительность каждого режима должна быть случайной в определенных границах:
     - режим «сидеть» — от 4 до 12 секунд (8 ± 4 секунды);
     - режим «стоять» — от 4 до 10 секунд (7 ± 3 секунды);
     - режим «спать» — от 14 до 20 секунд (17 ± 3 секунды);
     - режим «есть» — 7 секунд;
   - В состоянии бездействия пользователь должен иметь возможность гладить кота, если

кот не находится в состоянии «есть»;

- Когда пользователь гладит кота, у кота должны воспроизводиться звук мурлыканья и включаться соответствующая анимация;
- В состоянии «есть» и у кота должны воспроизводиться соответствующий звук и включаться соответствующая анимация;
- В состоянии «следование за пользователем» кот должен следовать к случайной точке в радиусе 150 сантиметров от игрока;
- В состоянии «следование к месту объекта с тегом CatFindable» кот должен издавать звуки мяуканья и направляться к точке расположения объекта, у которого есть тег CatFindable. В данном состоянии над котом должна быть надпись «Предмет», чтобы пользователь знал о том, что кот что-то «нашел», даже если играет без звука. Данное состояние должно включаться, когда пользователь попадает в Collision Box предмета. Кот должен находиться около этого объекта до тех пор, пока пользователь его не поднимет. При этом кот должен находиться в состоянии бездействия и мяукать с интервалом в пять секунд, привлекая к себе внимание;
- Кот не должен уметь заходить в воду: около нее он должен останавливаться и ждать выхода пользователя из воды.

9) Взаимодействие с виджетами в игре должно происходить с помощью наведения невидимого во время игры луча, исходящего от контроллера, на нужный элемент (компонент Widget Interaction). При пересечении луча с активным виджетом должна появляться сфера взаимодействия, являющаяся аналогом компьютерного курсора. Эквивалентом «клика» должно являться нажатие на кнопку триггера (Trigger) активного контроллера. Для смены активного контроллера нужно нажать на кнопку триггера на неактивном контроллере.

10) В офисе должен быть планшет:
    - Пользователь должен иметь возможность брать планшет в любую руку с помощью нажатия на кнопку захвата предмета на контроллере (Right & Left Grip);
    - При выпускании планшета из рук гаджет должен телепортироваться в исходную точку пространства;
    - К поверхности планшета должен быть привязан виджет, с элементами которого пользователь должен иметь возможность взаимодействовать;
    - При нажатии на кнопку с заданием должна выводиться информация о задании, о пля-

же, на который отправится пользователь, и о минимальном размере гонорара, который можно за задание получить;

- При открытии задания на планшете должна появляться кнопка, по нажатию на которую уровень переключится на соответствующий пляжный уровень.

11) На любом уровне пользователь должен иметь возможность открыть панель «Инструменты»:
    - Внутри должны быть две глобальные вкладки: «Действия» и «Магазин»;
    - Во вкладке «Действия» должен присутствовать список того, что пользователь может использовать или сделать:
        - Покормить кота. Данное действие должно быть доступно только в офисе. При нажатии на данный пункт панель должна закрываться, а перед котом должна появляться еда. Спустя секунду после того, как кот перейдет в состояние «стоять», должны включаться анимация принятия пищи котом и соответствующий звук, спустя 5 секунд они должны выключиться. За секунду до завершения анимации и воспроизведения звука объект еды должен быть уничтожен. Пункт должен быть недоступен, пока кот ест;
        - Мусорный бак. Он должна быть доступен только на пляжных уровнях на стадии уборки территории. После нажатия на соответствующую кнопку в панели корзина должна появиться перед пользователем, а панель должна закрыться. Теперь при открытии панели должны быть кнопки «Переместить корзину ко мне», которая поменяет местоположение корзины в пространстве так, что она окажется перед пользователем, и «Уничтожить корзину», которая уничтожит корзину. Пользователь должен иметь возможность брать корзину за края, поднимать, перемещать, бросать и ставить на землю любой из рук. При попадании любого мусорного объекта в корзину: объект должен уничтожаться, счетчик найденных мусорных объектов увеличиваться на единицу, из корзины должна плавно подниматься надпись, которая исчезнет через 5 секунд, с суммой, которую пользователь получил за утилизацию данного мусорного объекта;
        - Включение и выключение режима «отделки» и «продажи». Данные кнопки должны быть доступны только на этапе декорирования пляжа. При наведении активного контроллера на объект при активном режиме «отделки» или «продажи». В режиме «продажи» контур объекта должен изменить цвет, над объектом должна

быть выведена надпись с ценой, которую можно получить при продаже данного объекта. При нажатии на правый триггер должна быть выведена надпись, запрашивающая подтверждение на продажу товара.

- Во вкладке «Магазин» должны находиться товары, которые пользователь может поставить на пляжный уровень. Данная вкладка должна быть недоступна на стадии уборки уровня. Цвет некоторых товаров можно поменять, если приобрести для них соответствующую краску. У каждого товара должны быть название, описание, стоимость и категория.

  При нажатии на товар панель «Инструменты» должна закрываться, а товар должен появиться на карте — произойдет переход в режим «покупки» (то же, что и режим отделки, только после выхода из режима у пользователя списывается сумма покупки).

12) На пляжных уровнях должен быть предусмотрен режим «отделки»:
    - Режим должен быть доступен в случае, если пользователь уже приступил к декорированию территории;
    - Должен включаться и выключаться путем нажатия на кнопку A правого контроллера;
    - Должен автоматически включаться при покупке товара в магазине и его появлении на карте;
    - В данном режиме для игрока должны быть недоступны телепортация и передвижение с помощью джойстика контроллера;
    - Поиск предмета, находящегося на пути направления контроллера игрока, должен происходить по методу бегущих лучей;
    - Предмет должен выделяться по нажатию на триггер активного контроллера;
    - В данном режиме пользователь должен иметь возможность переместить выделенный объект на любое подходящее место с помощью джойстика левого контроллера;
    - В данном режиме пользователь должен иметь возможность вращать выделенный объект по оси Z с помощью джойстика правого контроллера.

13) На пляжных уровнях должен быть предусмотрен режим «продажи»:
    - Режим должен быть доступен в случае, если пользователь уже приступил к декорированию территории;
    - Должен включаться и выключаться путем нажатия на кнопку X левого контроллера;
    - В данном режиме для игрока должны быть недоступны телепортация и передвижение

с помощью джойстика контроллера;

- Поиск предмета, находящегося на пути направления контроллера игрока, должен происходить по методу бегущих лучей;
- Предмет должен выделяться по нажатию на триггер активного контроллера;
- В данном режиме пользователь должен иметь возможность продать выделенный объект или отказаться от покупки.

14) В приложении должно быть реализовано обучение, в ходе которого пользователю расскажут об элементах управления, назначении панели «Инструменты», о работе с планшетом и о взаимодействии с котом. Обучение должно автоматически воспроизводиться при первом входе в игру, а также воспроизводиться повторно при нажатии на соответствующую кнопку в меню паузы. Должна быть возможность обучение пропустить.

15) Должно быть реализовано меню паузы для пляжного уровня, вызывающееся нажатием на кнопку B правого контроллера в случае, если пользователь находится на пляжном уровне. При его открытии блокируются все остальные возможности в игре (изменение положения предметов, взаимодействие с котом, открытие панели «Инструменты», вход в режим «отделки» или «продажи» и выход из них). Во время активного меню паузы у пользователя должна быть возможность:
    - Вернуть первоначальное положение всех существующих на текущем уровне мусорных объектов;
    - Сгенерировать уровень заново;
    - Открыть меню настроек, в котором можно будет изменить громкость звуков, изменить громкость музыки, переключить текущий трек, начать игру заново (стереть текущее сохранение и начать с нуля);
    - Воспроизвести обучение;
    - Посмотреть управление;
    - Сохранить игру;
    - Вернуться с уровня в офис (при этом текущее состояние уровня нужно сохранить);
    - Покинуть игру.

16) На пляжном уровне генерация меню паузы, объекта с всплывающим сообщением и панели «Инструменты» в пространстве должна происходить следующим образом:
    - Получение текущего положения камеры пользователя (далее — camera);

- Получение вектора направления камеры с помощью функции Get Forward Vector (далее — vector);
- Присвоение координатам X, Y, Z объекта паузы значений $\langle camera.X + 120 \cdot vector.X \rangle$, $\langle camera.Y + 120 \cdot vector.Y \rangle$, $\langle camera.Z$ соответственно;
- Нахождение вектора поворота с помощью функции Find Look At Rotation, где стартовые координаты X, Y, Z — camera.X, camera.Y, camera.Z, финальные координаты X, Y, Z — координата X объекта паузы, координата Y объекта паузы, 0 соответственно;
- Присвоение значения координаты Z, полученной с помощью функции Find Look At Rotation на предыдущем шаге, в поворот объекта паузы по Z;
- Присвоение значения 1.0 в параметр Scale по X, Y, Z.

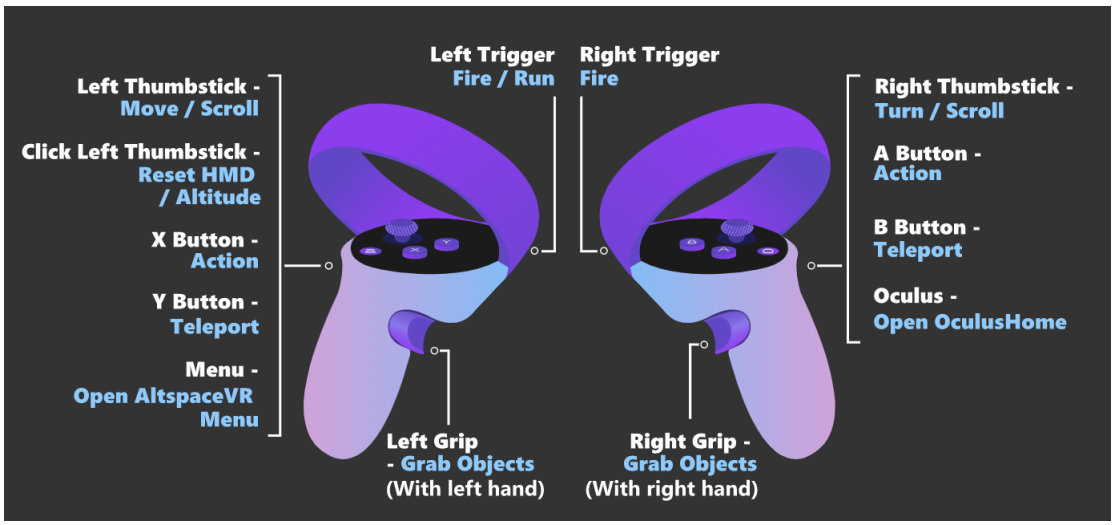
17) В офисе кнопка B должна бездействовать. Альтернативой паузы в офисе выступают кнопки, перманентно располагающиеся на стене перед пользователем. При нажатии на соответствующие кнопки происходят следующие действия:
    - Перезагрузка текущего уровня;
    - Открытие меню настроек, в котором можно будет изменить громкость звуков, изменить громкость музыки, переключить текущий трек, начать игру заново (стереть текущее сохранение и начать с нуля);
    - Воспроизведение обучения;
    - Просмотр управления;
    - Сохранение игры;
    - Выход из игры.
18) На каждом игровом уровне должна воспроизводиться музыка.
19) При бросании пользователем объектов в воду должен воспроизводиться соответствующий звук.
20) Должно быть реализовано сохранение, которое будет хранить информацию о текущем внешнем виде и состоянии всех начатых игровых уровней, окраске кота, пройденных уровнях, количестве денег, имени пользователя, выбранном на пляжном уровне треке, громкости звуков и музыки. Сохранение должно вызываться после выбора окраса кота, перед выходом из пляжного уровня и перед выходом из игры. Также у пользователя должна быть возможность сохраниться в любой момент времени с помощью кнопки в меню паузы.

4.1.2 Организация входных данных

Входными данными являются:

1) Поворот VR-очков Oculus Quest 2;
2) Перемещение VR-очков Oculus Quest 2 в пространстве;
3) Перемещение контроллеров Oculus Quest 2 в пространстве.

Также входными данными являются взаимодействия со следующими элементами управления контроллеров Oculus Quest 2 (см. рисунок) [4]:



1) Вращение аналогового джойстика на левом контроллере Oculus Quest 2 (Left Thumstick);
2) Нажатие на аналоговый джойстик на левом контроллере Oculus Quest 2 (Click Left Thumstick);
3) Прикосновение к аналоговому джойстику на левом контроллере Oculus Quest 2 (Touch Left Thumstick);
4) Нажатие на кнопку X левого контроллера Oculus Quest 2;
5) Прикосновение к кнопке X левого контроллера Oculus Quest 2;
6) Нажатие на кнопку Y левого контроллера Oculus Quest 2;
7) Прикосновение к кнопке Y левого контроллера Oculus Quest 2;
8) Нажатие на кнопку захвата предмета (под средним пальцем) на левом контроллере Oculus Quest 2 (Left Grip);
9) Нажатие на кнопку триггера (под указательным пальцем) на левом контроллере Oculus Quest 2 (Left Trigger);
10) Прикосновение к кнопке триггера (под указательным пальцем) на левом контроллере Oculus Quest 2 (Touch Left Trigger);

11) Вращение аналогового джойстика на правом контроллере Oculus Quest 2 (Right Thumstick);
12) Нажатие на аналоговый джойстик на правом контроллере Oculus Quest 2 (Click Right Thumstick);
13) Прикосновение к аналоговому джойстику на правом контроллере Oculus Quest 2 (Touch Right Thumstick);
14) Нажатие на кнопку A правого контроллера Oculus Quest 2;
15) Прикосновение к кнопке A правого контроллера Oculus Quest 2;
16) Нажатие на кнопку B правого контроллера Oculus Quest 2;
17) Прикосновение к кнопке B правого контроллера Oculus Quest 2;
18) Нажатие на кнопку «Окулус» правого контроллера Oculus Quest 2;
19) Нажатие на кнопку захвата предмета (под средним пальцем) на правом контроллере Oculus Quest 2 (Right Grip);
20) Нажатие на кнопку триггера (под указательным пальцем) на правом контроллере Oculus Quest 2 (Right Trigger).
21) Прикосновение к кнопке (под указательным пальцем) на правом контроллере Oculus Quest 2 (Touch Right Trigger).

4.1.3 Организация выходных данных

1) Выходными данными при вращении VR-очков Oculus Quest 2 в реальном мире является соответствующее вращение камеры в игре;
2) Выходными данными при перемещении VR-очков Oculus Quest 2 в пространстве является соответствующее изменение положения игрока в игровом мире;
3) Выходными данными при перемещении любого из двух контроллеров Oculus Quest 2 в пространстве является соответствующее изменение положения контроллера в игровом мире;
4) Выходными данными при вращении аналогового джойстика на левом контроллере Oculus Quest 2 (Left Thumstick) является плавное передвижение игрока в игровом мире в соответствующую направлению джойстика сторону;
5) Выходными данными при нажатии на аналоговый джойстик на левом контроллере Oculus Quest 2 (Click Left Thumstick) является появление параболообразной линии со стрелкой на конце в случае, если возможен телепорт в место, к которому повернут контроллер;
6) Выходными данными при вращении аналогового джойстика на левом контроллере Oculus Quest 2 (Left Thumstick) после нажатия на него (Click Left Thumstick) является изменение направ-

ления стрелки телепорта, если телепорт возможен;

7) Выходными данными при отпускании аналогового джойстика на левом контроллере Oculus Quest 2 (Left Thumstick) является телепортация к выбранному месту на карте с изменением поворота камеры, соответствующем повороту стрелки;
8) Выходными данными при прикосновении к аналоговому джойстику на левом контроллере Oculus Quest 2 (Touch Left Thumstick) является выделение красным цветом соответствующего элемента на изображении контроллеров в случае, если панель просмотра управления открыта;
9) Выходными данными при нажатии на кнопку X левого контроллера Oculus Quest 2 является активация режима «Продажи» в случае, если он был неактивен, и деактивация в случае, если был активен;
10) Выходными данными при прикосновении к кнопке X левого контроллера Oculus Quest 2 является выделение красным цветом соответствующего элемента на изображении контроллеров в случае, если панель просмотра управления открыта;
11) Выходными данными при нажатии на кнопку Y левого контроллера Oculus Quest 2 является открытие является активация панели «Инструменты» в случае, если она был неактивна, и деактивация в случае, если была активна;
12) Выходными данными при прикосновении к кнопке Y левого контроллера Oculus Quest 2 является выделение красным цветом соответствующего элемента на изображении контроллеров в случае, если панель просмотра управления открыта;
13) Выходными данными при нажатии на кнопку захвата предмета (под средним пальцем) на левом контроллере Oculus Quest 2 (Left Grip) является изменение анимации руки и захват предмета, находящегося в зоне контроллера в случае неактивности панели просмотра управления; иначе — выделение цветом соответствующего элемента на изображении контроллеров в панели просмотра управления;
14) Выходными данными при нажатии на кнопку триггера (под указательным пальцем) на левом контроллере Oculus Quest 2 (Left Trigger) является активация левого контроллера в случае, если контроллер был неактивен; «клик» на объект в случае, если контроллер активен; выделение объекта, на который наведен контроллер в данный момент, в режимах «Отделки» и «Продажи», если контроллер активен;
15) Выходными данными при прикосновении к кнопке триггера (под указательным пальцем) на левом контроллере Oculus Quest 2 (Touch Left Trigger) является выделение красным цветом

соответствующего элемента на изображении контроллеров в случае, если панель просмотра управления открыта;

16) Выходными данными при вращении аналогового джойстика на правом контроллере Oculus Quest 2 (Right Thumstick) является перемещение камеры игрока в игровом мире в соответствующую направлению джойстика сторону на 45°;
17) Выходными данными при нажатии на аналоговый джойстик на правом контроллере Oculus Quest 2 (Click Right Thumstick) является появление параболообразной линии со стрелкой на конце в случае, если возможен телепорт в место, к которому повернут контроллер;
18) Выходными данными при вращении аналогового джойстика на правом контроллере Oculus Quest 2 (Right Thumstick) после нажатия на него (Click Right Thumstick) является изменение направления стрелки телепорта, если телепорт возможен;
19) Выходными данными при отпускании аналогового джойстика на правом контроллере Oculus Quest 2 (Right Thumstick) является телепортация к выбранному месту на карте с изменением поворота камеры, соответствующем повороту стрелки;
20) Выходными данными при прикосновении к аналоговому джойстику на правом контроллере Oculus Quest 2 (Touch Right Thumstick) является выделение красным цветом соответствующего элемента на изображении контроллеров в случае, если панель просмотра управления открыта;
21) Выходными данными при нажатии на кнопку A правого контроллера Oculus Quest 2 является активация режима «Отделки» в случае, если он был неактивен, и деактивация в случае, если был активен;
22) Выходными данными при прикосновении к кнопке A правого контроллера Oculus Quest 2 является выделение красным цветом соответствующего элемента на изображении контроллеров в случае, если панель просмотра управления открыта;
23) Выходными данными при нажатии на кнопку B правого контроллер Oculus Quest 2 является активация игрового меню паузы в случае, если она был неактивна, и деактивация в случае, если была активна;
24) Выходными данными при прикосновении к кнопке B правого контроллера Oculus Quest 2 является выделение красным цветом соответствующего элемента на изображении контроллеров в случае, если панель просмотра управления открыта;
25) Выходными данными при нажатии на кнопку «Oculus» (правый контроллер Oculus Quest 2)

является открытие меню игровых очков Oculus Quest 2;

26) Выходными данными при нажатии на кнопку захвата предмета (под средним пальцем) на правом контроллере Oculus Quest 2 (Right Grip) является изменение анимации руки и захват предмета, находящегося в зоне контроллера в случае неактивности панели просмотра управления; иначе — выделение цветом соответствующего элемента на изображении контроллеров в панели просмотра управления;
27) Выходными данными при нажатии на кнопку триггера (под указательным пальцем) на правом контроллере Oculus Quest 2 (Right Trigger) является активация левого контроллера в случае, если контроллер был неактивен; «клик» на объект в случае, если контроллер активен; выделение объекта, на который наведен контроллер в данный момент, в режимах «Отделки» и «Продажи», если контроллер активен;
28) Выходными данными при прикосновении к кнопке триггера (под указательным пальцем) на правом контроллере Oculus Quest 2 (Touch Right Trigger) является выделение красным цветом соответствующего элемента на изображении контроллеров в случае, если панель просмотра управления открыта.

4.2 Требования к временным характеристикам

Время отклика приложения при стабильном подключении ПК к электропитанию не должно превышать одной секунды.

4.3 Требования к интерфейсу

1) Программа должна визуализировать трехмерную реальность с реалистичными соотношениями размеров предметов с ростом человека;
2) Используемые в приложении модели должны быть Low-Poly и/или Mid-Poly;
3) Офис должен представлять из себя комнату размера 3 на 4 метра с одним окном. Источники света должны имитировать дневной свет из окна и свет от люстры внутри комнаты. В офисе должны располагаться следующие трехмерные модели: кровать, шкаф, дверь, стол, стул, оконная рама, штора, карниз, люстра, подставка для планшета, планшет, кот;
4) Локации пляжей должны быть основаны на внешнем виде соответствующих реальных местностей;
5) Количество денег пользователя должно выводиться в правом верхнем углу виджета паузы и во вкладке «Магазин» панели «Инструменты».

4.4 Требования к надежности

1) Приложение не должно аварийно завершаться при любом наборе входных данных (если не подразумевается отладка приложения);
2) Программа не должна давать сбой при некорректных действиях пользователя.

4.5 Условия эксплуатации

4.5.1 Климатические условия эксплуатации

Климатические условия эксплуатации, при которых должна обеспечиваться корректная работа программы, должны соответствовать требованиям, предъявляемым к техническим средствам, реализующим данный программный продукт.

4.5.2 Требования к видам обслуживания

Обслуживание не требуется.

4.5.3 Требования к численности и квалификации персонала

Для работы с приложением достаточно одного человека, знакомого с интерфейсом VR-очков Oculus Quest 2 и способного взаимодействовать с окружающей виртуальной средой.

Требования к квалификации пользователя не предъявляются.

4.5.4 Требования к окружающей среде

Следует избегать прямого попадания солнечных лучей на VR-очки.

Вокруг пользователя должно быть свободное пространство, которое позволит вытягивать руки во всех направлениях, ничего не задевая в режимах игры сидя и стоя.

4.6 Требования к составу и параметрам технических средств

В распоряжении пользователя должны быть VR-очки Oculus Quest 2, ПК, провод Oculus Link (опционально). Провод Oculus Link не нужен, если осуществлено подключение очков к ПК с помощью Air Link.

Минимальные требования, предъявляемые к hardware ПК пользователя для поддержки VR-очков Oculus Quest 2 [19]:

- Процессор: Intel i5-4590 / AMD Ryzen 5 1500X или лучше;

- Оперативная память: 8 GB и выше;
- Видеокарта: NVIDIA GeForce GTX 970 / AMD Radeon 400 Series или лучше;
- USB-порты: 1x USB-порт (если не используется Air Link).

4.7 Требования к информационной и программной совместимости

4.7.1 Требования к исходным кодам и языкам программирования

Программа должна быть разработана на движке Unreal Engine 4 версии 4.26.2 на языке программирования Blueprint.

4.7.2 Требования к программным средствам, используемым программой

Требование к software VR-очков — активный режим Air Link / Oculus Link, совместимость со SteamVR.

Минимальные требования, предъявляемые к software ПК для поддержки VR-очков Oculus Quest 2 [19]:

- Операционная система: Windows 10;
- Pixel шейдеры: 5.1;
- Vertex шейдеры: 5.1;
- Выделенная видео-память: 3 GB или больше;
- Установленный SteamVR;
- Установленное приложение Oculus с активным сопряжением с очками пользователя.

4.7.3 Требования к защите информации и программ

Требования к защите информации и программ не предъявляются.

4.8 Требования к маркировке и упаковке

Программа распространяется в закрытом режиме в виде электронного пакета, содержащего программную документацию, приложение (исполняемые файлы, набор готовых к работе кодов и прочие необходимые для работы файлы) и презентацию проекта.

4.9 Требования к транспортированию и хранению

Транспортировка программного продукта должна осуществляться без нарушения полноты комплекта, предоставленного разработчиком изначально.

Хранение продукта осуществляется с помощью репозитория на GitHub.

5 ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

5.1 Состав программной документации:

1) «VR-игра “Дизайнер пляжей” на движке Unreal Engine 4». Техническое задание (ГОСТ 19.201 –78);
2) «VR-игра “Дизайнер пляжей” на движке Unreal Engine 4». Пояснительная записка (ГОСТ 19.404–79);
3) «VR-игра “Дизайнер пляжей” на движке Unreal Engine 4». Программа и методика испытаний (ГОСТ 19.301–79);
4) «VR-игра “Дизайнер пляжей” на движке Unreal Engine 4». Руководство оператора (ГОСТ 19.505–79);
5) «VR-игра “Дизайнер пляжей” на движке Unreal Engine 4». Текст программы (ГОСТ 19.401–78).

5.2 Специальные требования к программной документации

Документы к программе должны быть выполнены в соответствии с ГОСТ 19.106–78 и ГОСТами к каждому виду документа (см. п. 5.1.).

Пояснительная записка должна быть загружена в систему Антиплагиат через SmartLMS «НИУ ВШЭ».

Лист, подтверждающий загрузку пояснительной записки, загружается в SmartLMS вместе со всеми материалами не позже, чем за три до защиты курсовой работы.

Титульные листы документов должны быть подписаны руководителем разработки.

Документация в формате .pdf или .docx и программа сдаются в электронном виде в архиве формата .zip или .rar.

15.04.2022 все материалы курсового проекта:

- техническая документация,
- программный проект,
- исполняемый файл,
- отзыв руководителя

должны быть загружены одним или несколькими архивами в проект дисциплины «Курсовой проект, 3 курс ПИ» в личном кабинете в SmartLMS «НИУ ВШЭ».

6 ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

6.1 Ориентировочная экономическая эффективность

Данный курсовой проект не предусматривает расчет экономической эффективности.

6.2 Предполагаемая потребность

Предполагаемая потребность обуславливается тем, что у пользователей существует необходимость в удовлетворении эстетической и экологической потребностей [20]. Данное приложение также будет востребовано среди людей, желающих развивать свои навыки дизайна открытых природных пространств.

6.3 Экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами

На данный момент не существует прямых аналогов разрабатываемого продукта. В таблице 1 приведена сравнительная характеристика с приложениями, наиболее близкими к теме разрабатываемого проекта.

Таблица 1 – Аналоги

	«House Flipper»	«Garden Flipper»	«Home Improvisation: Furniture Sandbox»	Home Designer — Living Room	Home Designer — Home Sweet Home	«Beach Designer»
Поддержка VR	–	–	+	–	–	+
Разнообразие используемых инструментов	+	–	+	–	–	–
Основная цель — декорирование	+	+	+	–	–	+
Основная цель — мини-игры	–	–	–	+	+	–
Работа с пляжами	–	–	–	–	–	+

7 СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

Таблица 2 – Стадии и этапы разработки

Стадии разработки	Этапы работ	Содержание работ	Сроки
1. Техническое задание	Обоснование необходимости разработки	Постановка задачи	01.11.21
		Сбор исходных теоретических материалов	01.11.21
	Научно-исследовательский этап разработки	Определение структуры входных и выходных данных	13.11.21
		Предварительный выбор методов решения задач	14.11.21
		Определение требований к техническим и программным средствам	16.11.21
		Обоснование возможности решения поставленной задачи	16.11.21
	Разработка и утверждение технического задания	Определение требований к программному продукту	10.01.22
		Выбор движка и языков программирования	10.01.22
		Разработка и согласование технического задания с научным руководителем	03.02.22
		Загрузка согласованного технического задания в SmartLMS	17.02.22
2. Рабочий проект	Разработка программы	Предварительная разработка структуры программы	20.02.22
		Программирование и отладка программы	10.03.22
	Разработка программной документации	Разработка программных документов в соответствии с требованиями ГОСТ 19 ЕСПД (Единой системы программной документации)	21.03.22
	Испытания программы	Разработка, согласование и утверждение порядка в методики испытаний	25.03.22
		Поведение испытаний программы в соответствии с утверждением порядком и методикой	28.03.22
		Корректировка программы и программной документации по результатам испытаний	30.03.22
3. Внедрение	Подготовка и передача программы	Подготовка программы и программной документации для презентации и защиты	01.04.22
		Представление разработанного программного продукта научному руководителю и получение отзыва	04.04.22

Продолжение таблицы 2

		Представление разработанного программного продукта научному руководителю и получение отзыва.	04.04.22
		Загрузка Пояснительной записки в систему Антиплагиат через Smart LMS НИУ ВШЭ.	11.04.22
		Загрузка материалов курсового проекта в Smart LMS, дисциплина «Курсовой проект, 3 курс ПИ»	15.04.22
		Загрузка презентации курсового проекта в Smart LMS, дисциплина «Курсовой проект, 3 курс ПИ»	17.04.22
		Защита программного продукта комиссии.	21.04.22

Разработка данного программного продукта должна быть завершена к 4 апреля 2022 г. Исполнитель – Шалаева Марина, студентка группы БПИ198 факультета компьютерных наук НИУ ВШЭ.

8 ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ

Контроль и приемка разработки осуществляются в соответствии с документом «Программа и методика испытаний» (ГОСТ 19.301–79) [15].

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] Behavior Trees [Электронный ресурс] / behavior trees. Режим доступа: https://docs.unrealengine.com/4.27/en-US/InteractiveExperiences/ArtificialIntelligence/BehaviorTrees/, свободный. (дата обращения: 01.02.2022).

[2] Lowpoly, Highpoly и вертекс нормали [Электронный ресурс] / Lowpoly. Режим доступа: https://dtf.ru/gamedev/73251-statya-3-7-pro-setku-lowpoly-highpoly-i-verteks-normali, свободный. (дата обращения: 28.01.2022).

[3] Navmesh Content Examples [Электронный ресурс] / navmesh. Режим доступа: https://docs.unrealengine.com/4.27/en-US/Resources/ContentExamples/NavMesh/, свободный. (дата обращения: 01.02.2022).

[4] Oculus Quest/Quest 2 controls and permissions [Электронный ресурс] / oculus controls. Режим доступа: https://docs.microsoft.com/en-us/windows/mixed-reality/altspace-vr/getting-started/oculus-controls, свободный. (дата обращения: 01.02.2022).

[5] Virtual Reality, VR [Электронный ресурс] / vr. Режим доступа: https://www.it.ua/ru/knowledge-base/technology-innovation/virtualnaja-realnost-vr, свободный. (дата обращения: 10.01.2022).

[6] Volume зоны [Электронный ресурс] / volume зоны. Режим доступа: https://uengine.ru/site-content/docs/actors-geometry/volumes, свободный. (дата обращения: 25.01.2022).

[7] Виртуальные очки [Электронный ресурс] / vr-очки. Режим доступа: https://www.testprom.ru/artprod/elektronika/sertifikacziya-virtualnyx-ochkov, свободный. (дата обращения: 24.10.2021).

[8] ГОСТ 19.101–77 Виды программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

[9] ГОСТ 19.102–77 Стадии разработки. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

[10] ГОСТ 19.103–77 Обозначения программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

[11] ГОСТ 19.104–78 Основные надписи. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

[12] ГОСТ 19.105–78 Общие требования к программным документам. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

[13] ГОСТ 19.106–78 Требования к программным документам, выполненным печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

[14] ГОСТ 19.201–78 Техническое задание. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

[15] ГОСТ 19.301–79 Программа и методика испытаний. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

[16] ГОСТ 19.603–78 Общие правила внесения изменений. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

[17] ГОСТ 19.604–78 Правила внесения изменений в программные документы, выполненные печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

[18] Неигровой персонаж [Электронный ресурс] / npc. Режим доступа: https://ru.wikipedia.org/wiki/Неигровой_персонаж, свободный. (дата обращения: 09.02.2022).

[19] Системные требования Oculus Quest 2 [Электронный ресурс] / oculus quest 2 requirements. Режим доступа: https://ru.pickgamer.com/games/oculus-quest-2/requirements, свободный. (дата обращения: 27.01.2022).

[20] Тимошенко, Т.В. Экология и потребности человека [Электронный ресурс]: Известия ЮФУ. Технические науки — Режим доступа: https://cyberleninka.ru/article/n/ekologiya-i-potrebnosti-cheloveka/viewer, свободный. (дата обращения: 31.01.2022).

[21] Что такое Oculus Quest 2 [Электронный ресурс] / Oculus Quest 2. Режим доступа: https://tech.onliner.by/2021/11/11/vpechatleniya-ot-oculus-quest-2, свободный. (дата обращения: 21.01.2022).

ПРИЛОЖЕНИЕ 1

ТЕРМИНОЛОГИЯ

Таблица 1 – Терминология

Термин	Значение
Виртуальная реальность (Virtual Reality, VR)	«Созданный техническими средствами мир, передаваемый человеку через его ощущения: зрение, слух, осязание и другие. Виртуальная реальность имитирует как воздействие, так и реакции на воздействие. Для создания убедительного комплекса ощущений реальности компьютерный синтез свойств и реакций виртуальной реальности производится в реальном времени» [5]
Очки виртуальной реальности (VR-очки, VR-шлем)	«Специализированный прибор, способный симулировать разнообразные аудиовизуальные трехмерные пространства. Состоит из пластикового (реже картонного) корпуса, экрана с перегородкой и асферических линз, которые фокусируют изображение» [7]
Контроллер	Устройство ввода информации, используемое в видеоиграх или развлекательных системах для получения входных данных. Обычно используется для управления объектом или персонажем в игре
Oculus Quest 2	«Полностью автономный VR-шлем, выпущенный компанией Oculus, которая входит в состав Meta» (Meta является признанной на территории России экстремистской организацией) [21]
NPC (Non-Player Character, Неигровой персонаж)	«Персонаж в играх, который не находится под контролем игрока» [18]
Low-Poly	Трёхмерная модель с малым количеством полигонов (примерно 5 — 10 тысяч) [2]
Mid-Poly	Трёхмерная модель со средним количеством полигонов
Nav Mesh	«Игровой объект, использующийся для определения зоны передвижения игрока и для вычисления навигационных путей для Искусственного Интеллекта на уровне» [6]
Nav Mesh Bounds Volume	«Игровой объект, использующийся для определения зоны расчета Nav Mesh на уровне» [6]

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

Лист регистрации изменений									
Номера листов (страниц)					Всего листов (страниц в докум.)	№ документа	Входящий № сопроводительного докум. и дата	Подп.	Дата
Изм.	Измененных	Замененных	Новых	Аннулированных					