|  |  |  |
| --- | --- | --- |
|  |  | УТВЕРЖДАЮ |
|  |  | Руководитель  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.  М.П. |

ТЕХНИЧЕСКИЙ ПРОЕКТ

Игра «ALEXITIMYA»

2024

Содержание

[1 Уточнение структур данных 2](#_Toc181651779)

[2 Архитектура проекта 3](#_Toc181651780)

[3 Формы и их представления 4](#_Toc181651781)

[4 Разработка алгоритма решения задачи 5](#_Toc181651782)

[5 Определения языка 10](#_Toc181651783)

[6 Структура программы и требования к техническим средствам 11](#_Toc181651784)

# Уточнение структур данных

У разрабатываемой игры «ALEXITIMYA» нету необходимости в создании БД, на замену ей приходит вспомогательная переменная PlayerPerf, которая хранит в себе все сохраненные данные о игровом процессе. Необходимость в сборе и обработке персональных данных отсутствует, так как не несет в себе никакой физической цели. Ниже будет представлен алгоритм сохранения игрового прогресса пользователя (См. Рис 1).



Рисунок 1 – Алгоритм выполнения сохранения игрового прогресса

# Архитектура проекта

Для разработки игры был выбран шаблон «First Person» внутри раздела «Видеоигры», данных шаблон предоставляет персонажа с готовым контроллером, при необходимости, данный контроллер можно дополнять игровыми механиками.

Описание использования основных технологий Unreal Engine 5, таких как:

* Nanite: Виртуализированная система микрополигональной геометрии, позволяющая напрямую импортировать и реплицировать многомиллионные полигональные сетки без потери производительности.
* Lumen: Полностью динамическая глобальная иллюминация и система отражений, позволяющая адаптировать непрямое освещение к изменениям прямого освещения или геометрии в реальном времени.
* Virtual Shadow Maps: Система, улучшающая качество теней и их производительность.
* Temporal Super Resolution (TSR): Система повышения разрешения, позволяющая рендерить кадры в более низком разрешении, но с сохранением качества, аналогичного кадрам в более высоком разрешении.

Работа с Активами и Материалами

* Описание процесса импорта и настройки активов, включая модели, текстуры и материалы. Использование инструментов như Quixel для создания и управления материалами.
* Организация объектов по слоям, что упрощает применение материалов к соответствующим объектам.

# Формы и их представления

Макет главного меню представлен на рисунке 2.



Рисунок 2 – Макет главного игрового меню

Данные макет показывает пример того, как будет выглядеть игровое меню, оно содержит в себе такие кнопки как, «Новая игра», она позволяет начать игру с самого начала, сбросив прошлый игровой процесс. Кнопка «Продолжить» запускает игру с последнего места, где было произведено сохранение, кнопка «Настройки» позволяет открыть меню настроек, и выставить приятные для пользователя пресеты графики или же подстроить под себя управление, кнопка «Выход» производит выход из главного меню

Прототип загрузочного экрана представлен на рисунке 3.

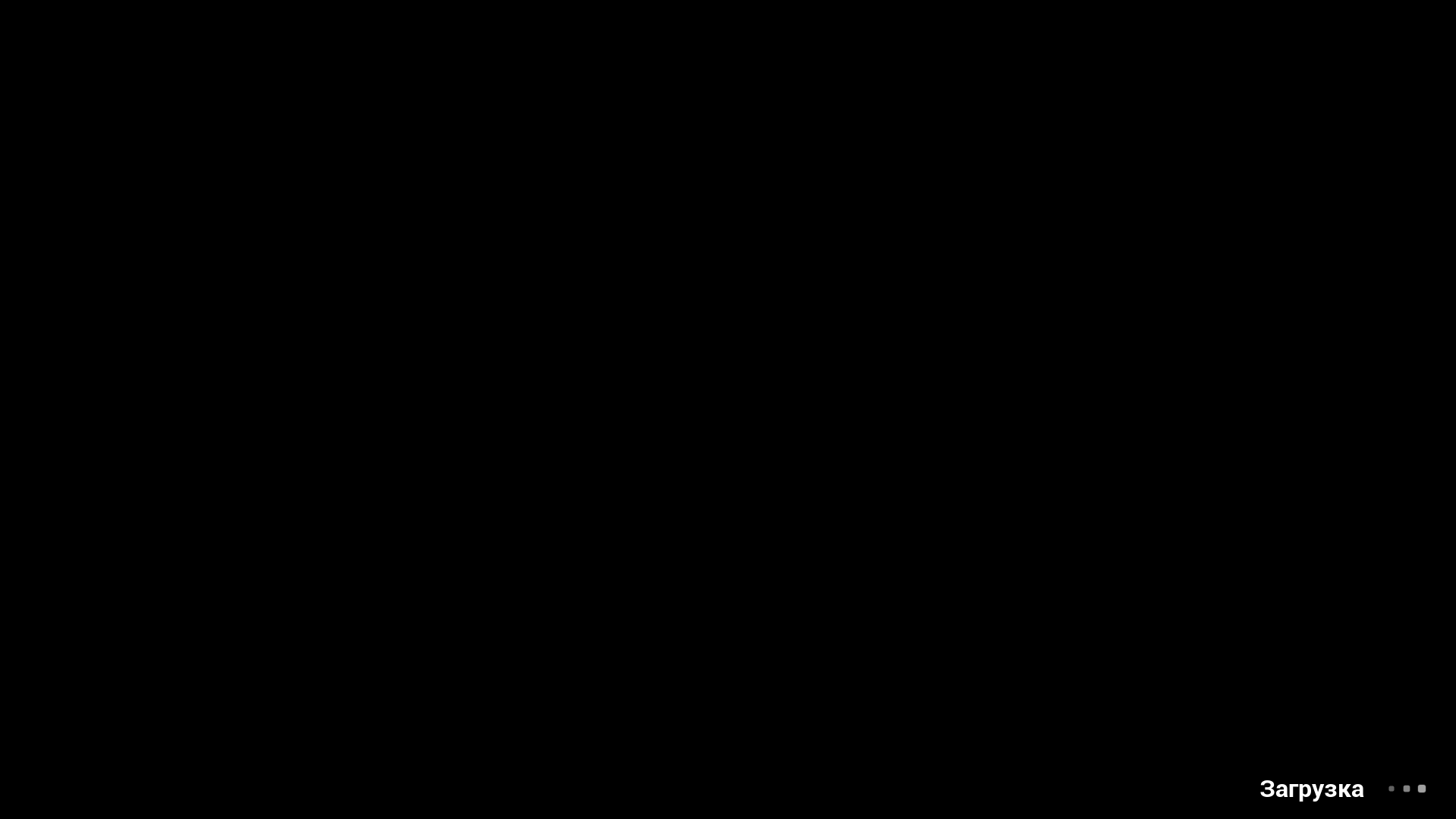


Рисунок 3 – Прототип загрузочного экрана

Загрузочный экран представляет из себя переходную фазу между игровыми уровнями, он может видоизменяться в зависимости от игрового прогресса.

# Разработка алгоритма решения задачи

Алгоритм «**Начало новой игры»**

Для лучшего понимания алгоритма «Начало новой игры» была создана схема алгоритма, которая изображена на рисунке 4.

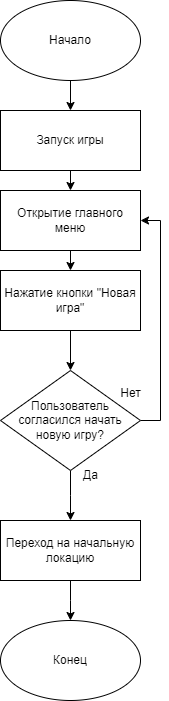


Рисунок 4 – Схема алгоритма «Начало новой игры»

Алгоритм «**Выполнение сохранения»**

Для лучшего понимания алгоритма «Выполнение сохранения» была создана схема алгоритма, которая изображена на рисунке 5.



Рисунок 5 – Схема алгоритма «Выполнение сохранения»

Так же для понимания как реагирует система был составлен пользовательский сценарий, который указан в таблице 1.

Таблица 1 – Пользовательские сценарий

|  |  |  |
| --- | --- | --- |
| **№** | **Пользователь** | **Система** |
| 1 | Нажатие кнопки «Новая игра» в главном меню | * 1. Активация BluePrint класса виджета главного меню.   2. Активируется задержка для проигрывания анимации нажатой кнопки.  3. После задержки и анимации происходит удаление виджета главного меню и происходит создание нового виджета с уточнением, хотел ли игрок начать новую игру или нет |
| 2 | Нажатие кнопки «Выход» в главном меню | * 1. Активация BluePrint класса виджета главного меню.   2. Активируется задержка для проигрывания анимации нажатой кнопки.  3. После задержки и анимации происходит удаление виджета главного меню и происходит создание нового виджета с уточнением, хотел ли игрок выйти из игры или нет |
| 3 | Нажатие клавиш «WASD» на клавиатуре | 1. Инициализация нажатия привязанных к перемещению клавиш  2. Получение значений векторов «Right vector» и «Forward vector» для актора в лице игрового персонажа  3. Перемещение игрового персонажа |
| 4 | Нажатие клавиши «E» на клавиатуре | 1. Активация BluePrint класса объекта, с которым происходит взаимодействие  2. Отображение созданного для этого объекта виджета  3. Отключение перемещения персонажа и осмотра мышью  4. За счет повторного нажатие данной клавиши происходит реверсивное действие за счет ноды FlipFlop. |
| 5 | Нажатие клавиши «ESC» | 1.Активация BluePrint класса BP\_FirstPersonCharacter  2.Отображения виджета меню паузы  3. Установка игры в состояние паузы  4. Отображение курсора, для взаимодействия с меню  5. При повторном нажатии данной кнопки происходят реверсивные действия за счет ноды FlipFlop |
| 6 | Вхождение в BoxTriger | 1. Идентификация столкновения игрока с коллизией объекта BoxTriger.  2. Выполнение запрограммированного действия внутри BluePrint класса. |
| 7 | Нажатие кнопки «Выход» в меню паузы | * 1. Активация BluePrint класса виджета меню паузы.   2. Закрытие игры за счет выполнения ноды «Quit game» |
| 8 | Нажатие кнопки «Продолжить» в меню паузы | * 1. Активация BluePrint класса виджета меню паузы.   2. Снятие игры с паузы  3. Удаление курсора с игрового экрана  4.Уничтожение виджета меню паузы |
| 9 | Нажатие кнопки «Продолжить» в главном меню | 1.Активация BluePrint класса главного меню  2. Получение значений положения игрока и его игрового прогресса из глобальной переменной  3. Запуск последнего игрового места |

# Определения языка

Для реализации объекта используется такой игровой движок как Unreal Engine 5 и визуальная система программирования BluePrint-ы.

Unreal Engine — это игровой движок от Epic Games, одной из крупнейших американских компаний по разработке игр и программного обеспечения. Движок предназначен в первую очередь для создания видеоигр, но также подходит для производства неигровых проектов в области архитектуры, строительства, автомобильной промышленности, медицины, кинематографа, анимации и других сфер.

BluePrint - Система визуального программирования в игровых движках Unreal Engine 4 и Unreal Engine 5. Основана на нодах с данными: событиями и функциями. Их можно связывать между собой и формировать элементы геймплея

# Структура программы и требования к техническим средствам

Функциональные требования:

* Пользователь должен иметь возможность управлять персонажем с помощью различных входных устройств (клавиатура, мышь, геймпад).
* Возможность сохранять и загружать игровые данные, включая прогресс, уровни и настройки.
* Возможность настройки графики и управления под потребности пользователя.
* Реализация реалистичной системы звука, включая эхо и другие звуковые эффекты.
* Реализация различных игровых механик, таких как движение персонажа и взаимодействие с объектами

Нефункциональные требования:

* Интуитивно понятный и простой интерфейс, адаптивный дизайн для различных устройств.