**Пояснительная записка к проекту: “Создание игры на языке программирования Python, при использовании модуля Pygame’**

**Работа учащихся:**

**Дунаева Глеба,**

**Шафоростовой**

**Елизаветы**

**г. Москва, 2025**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| I. | Введение | | | |  | |
| II. | Основная часть | | | |  | |
| 2.1 | Теоретическая часть | | | |  | |
|  | 2.1.1 | Планирование работы | | |  | |
| 2.2 | Практическая часть | | | |  | |
|  | 2.2.1 | | Структуры БД | |  | |
|  | 2.2.2 | | Классовая структура | |  | |
|  | 2.2.3 | | Отрисовка карты | |  | |
|  | 2.2.4 | | Создание местности и локаций | |  | |
|  | 2.2.5 | | Создание НПС | |  | |
|  | 2.2.6 | | Сюжет игры | |  | |
| III. | Заключение | | |  | |
| IV. | Приложение | | |  | |

**Оглавление**

1. **Введение**

Создание развлекательного проекта уже подразумевает под собой уникальность продукта фактически это продукт творческой деятельности. В нашем проекте мы реализуем игру, в которой игрок должен бегать по карте и изучать сюжет, взаимодействуя с объектами и НПС. Этим обусловлена актуальность проекта.

**Цель проекта**: создать игру.

**Задачи**:

1. Создание ТЗ;
2. Создание сюжета и проработка персонажей;
3. Создание графической составляющей;
4. Создание кодовой (функциональной части);
5. Попытка создать аудио составляющею;
6. **Основная часть**
   1. **Теоретическая часть**

**2.1.1 Планирование работы**

В первую очередь было произведено планирование всей последующей деятельность. Были изучены различные игры. Их интерфейс. Позже был составлен список более приоритетных технологий и интерфейсов.

Далее было сформулировано описание проекта, из которого был составлен примерный план действий:

1. Создание git-репозитория;
2. Периодичное создание requirements.txt;
3. Создание структуры каталогов для удобного распределения файлов;
4. Создание БД структуры;
5. Создание классов и их функций, последующая доработка;
6. Создание .py файлов с инициализацией окна игры и обработки событий;
7. Создание графики и карты, последующее подключение; Примечание: все в ЧБ стиле.
8. Создание аудио составляющей с депрессивным, тревожны настроем;

Позже мы составили текстовое задание (ТЗ), который дает полное описание наших требований.

**2.2 Практическая часть**

**2.2.1 Структура БД**

На данном этапе проекта был заложен фундамент рабочей логики нашей игры. Было предложено сделать автоматизацию заполнения БД.

Всего вышло 8 таблиц:

1. Игроки;
2. Все активности – все, с чем можно взаимодействовать, нужно для отслеживания прогресса по прохождению;
3. Предметы, для хранения информации о всех предметах в игре;
4. Места, описание локаций в которых будет что-то происходить;
5. Персонажи, для хранения информации о НПС;
6. Тексты, для записи сюжетных текстов;
7. Сессии, для инициализации новых игровых процессов с запоминанием прогресса;
8. Рейтинг-доска, для хранения игроков, прошедших игру и для дальнейшего вывода на финальном экране;

Стоит упомянуть о том, что почти для всех таблиц созданы каталоги, реализующие работу с этими таблицами, к примеру, обработку данных, или построение диалога. Также есть нюанс, заключается он в том, что мы используем JSON файлы для хранения информации, требующей иерархичности.

**2.2.2 Классовая структура**

В данном разделе разъясняется классовая структура.

Во-первых, мы создаем классы с функциональными названиями, также как и функции. Они создаться для дальнейшего использования в других файлах и оптимизации процесса, также для красоты оформления кода.

Во-вторых, различаться два вида классов:

1. Service – класс, что, технически не вы выполняет своей классовой ноши, так как ы нем создаться функции которые взаимодействуют с одноименными таблицами БД.
2. Model – это самый обычный класс, который несет некоторый функционал, который требуется от данного объекта.
3. Sprite - класс для создания и регулирования поведения спрайтов.