

МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет компьютерных наук

Техническое задание в соответствии с ГОСТ 34.602-89

Исполнители

\_\_\_\_\_ Богачев-Воевудский А.А.

\_\_\_\_\_ Веремеев В.

\_\_\_\_\_ Габелко М.

\_\_\_\_\_ Штукатуров Д.

\_\_\_\_\_ Елфимов А.

\_\_\_\_\_ Дубровин Д.

Заказчик

\_\_\_\_\_ Тарасов В.С.

Воронеж 2025

## **Содержание**

## Термины и определения

**Десктоп** – компьютер.

**Пиксель-арт** – графика, стилизованная под старые игры, где графика состояла из крупных пикселей.

**RPG** (Role Play Game) – тип игр, в котором игрок отыгрывает определенного персонажа и ему предоставлен выбор действий.

**FPS** (Frame Per Second) – количество кадров в секунду, наглядно отображающее производительность на устройстве.

**Юзабилити** – удобство / пригодность использования приложения.

**Логирование** — это фиксация событий в работе приложения, помогающая его разработчикам выявлять баги системы.

**User stories** – пользовательские сценарии, использующиеся для планирования сессии пользователя приложения.

**Acceptance Criteria** – критерии приемки, важная практика для улучшения коммуникации между разработчиками и заказчиками, а также неотъемлемая часть создания качественных пользовательских сценариев.

**UI Kit** - набор готовых решений пользовательского интерфейса.

**Иконографика** — это графический способ подачи информации, данных и знаний, целью которого является быстро и чётко преподнести сложную информацию.

**Эндпоинт** – конечная точка сервиса, к которой клиентское приложение обращается для выполнения определённых операций или получения данных.

**Микросервис** — это набор небольших модулей, на основе которых выполняется непрерывная поставка и развертывание больших и сложных приложений.

**Backend** - программно-аппаратная часть приложения, которую не видят пользователи: они не знают о работе внутренних процессов и не могут на них влиять.

## **1 Общие положения**

### **1.1 Анализ рынка**

- Рынок мобильных и десктопных игр в жанре "коллекционирование" активно развивается.
- Пиксель-арт привлекает игроков, ценящих ретро-стиль.
- Игры с элементами RPG (улучшение инвентаря, открытие новых локаций) пользуются популярностью.

### **1.2 Целевая аудитория**

- Возраст: 8–12 лет.
- Интересы: коллекционирование, рыбалка, пиксель-арт, RPG-элементы.

### **1.3 Основные конкуренты**

- "Stardew Valley" (хотя это фермерская игра, но содержит механику рыбной ловли и схожая рисовка).
- "Fishing Planet" (серьезная симуляция, но без пиксель-арта).
- "Fishdom" (фокус на декорировании аквариума).

## 2 Цели и задачи проекта

### 2.1.1 Цели

- Создать увлекательную игру в жанре "рыбалка", сочетающую коллекционирование, исследование и улучшение инвентаря.
- Реализовать пиксель-арт стиль для уникального визуального опыта.

### 2.1.2 Задачи

- Разработать 3 локации для рыбалки с разными условиями и типами рыб.
- Добавить магазины для продажи рыбы и покупки наживки.
- Реализовать бои с боссами для перехода между локациями.
- Создать базу данных для хранения прогресса игрока.

### 2.1.3 Критерии успешности

- Удержание игроков на 30 минут – 1 час в день.
- Позитивные отзывы о графике и игровом процессе.
- Монетизация через микротранзакции (например, уникальные наживки или косметические предметы).

## 2.2 Функциональные требования

- **Сохранение прогресса:** при повторном заходе игрока нужно дать ему возможность продолжить игру там, где он остановился. Прогресс игрока будет сохраняться через создание аккаунта внутри игры и внесением этой информации в базу данных.
- **Алгоритм изменения сложности:** в процессе игры от нескольких факторов будет зависеть сложность игрового процесса.
- **Возможность начать ловлю рыбы с любой части водоема:** игроку не нужно искать одну точку взаимодействия на весь водоем.

- **Выбор наживки:** можно будет купить наживку для ловли рыбы.
- **Различные характеристики у рыб:** вероятность поймать рыбу от наживки, погодных условий и прокачки удочки.
- **Улучшение удочки:** игрок может исследовать мир и находить улучшения для удочки, влияющие на баланс.
- **Продажа рыбы:** можно будет продать рыбу для покупки наживки или улучшения удочки.
- **Исследование мира:** для исследования мира есть сундуки, разбросанные по локации и содержащие в себе улучшения, которые нельзя купить в магазине.
- **Инвентарь:** игрок сам выбирает какую рыбу он хочет продать.
- **Достижения:** за выполнение определенных условий, не всегда видимых игроку, выдается достижение.
- **Генерация истории мира на записках:** по локации будут расставлены сундуки с улучшениями для удочки и иногда в них будут лежать записки, текст на который генерируется при старте новой игры.

### 2.3 Нефункциональные требования

- **Масштабируемость:** возможность добавить новые локации и предметы.
- **Производительность:** должно быть не менее 30 FPS в игре и возможность включить 60 FPS.
- **Юзабилити:** игроку должно быть интуитивно понятно, как работают механики в игре.
- **Локализация:** возможность игры на русском и английском языке.

- **Логирование данных:** нужно логировать данные для сбора статистики об игроках.

## **2.4 Пользовательские сценарии (User Stories)**

### **2.4.1 User Story: Начало игры**

- **Как:** Новый игрок.
- **Что хочу:** начать играть и выбрать первую локацию.
- **Почему:** чтобы начать процесс коллекционирования.
- **Acceptance Criteria:**
  - Игрок попадает на стартовый экран.
  - Игрок может выбрать первую локацию.

### **2.4.2 User Story: Ловля рыбы**

- **Как:** Игрок.
- **Что хочу:** ловить рыбу с использованием удочки.
- **Почему:** чтобы собирать коллекцию и получать деньги.
- **Acceptance Criteria:**
  - Игрок может выбирать место для рыбалки.
  - Процесс ловли включает мини-игру.
  - Игрок получает рыбу после успешной ловли.

### **2.4.3 User Story: Улучшение удочки**

- **Как:** Игрок.
- **Что хочу:** найти улучшения для удочки.
- **Почему:** чтобы сделать процесс ловли проще.
- **Acceptance Criteria:**

- Игрок может исследовать карту.
- На карте есть точки с улучшениями.
- Игрок может применить найденные улучшения.

#### **2.4.4 User Story: Бои с боссами**

- **Как:** Игрок.
- **Что хочу:** победить рыбу-босса.
- **Почему:** чтобы открыть новую локацию.
- **Acceptance Criteria:**
  - Игрок может вызвать босса после выполнения условий.
  - Бой включает специальные механики.
  - После победы открывается новая локация.

### **2.5 Перечень основных функциональных блоков системы**

#### **1. Система управления игроком**

- Перемещение по локации
- Взаимодействие с интерактивными объектами (водоем, магазин, сундуки)

#### **2. Система рыбалки**

- Мини-игра при взаимодействии с водоемом
- Искусственный интеллект у рыб

#### **3. Система улучшений**

- Поиск и применение улучшений для удочки.

#### **4. Система магазина**

- Покупка наживки



- Продажа рыбы
- Выдача заданий игроку

#### 5. Система боссов

- Бой с боссами в виде усложненной мини-игры
- Открытие новых локаций
- Мини-игра перед боем

#### 6. Система сохранения

- Сохранение прогресса игрока.
- Возможность стереть сохранение
- Загрузка прогресса при входе

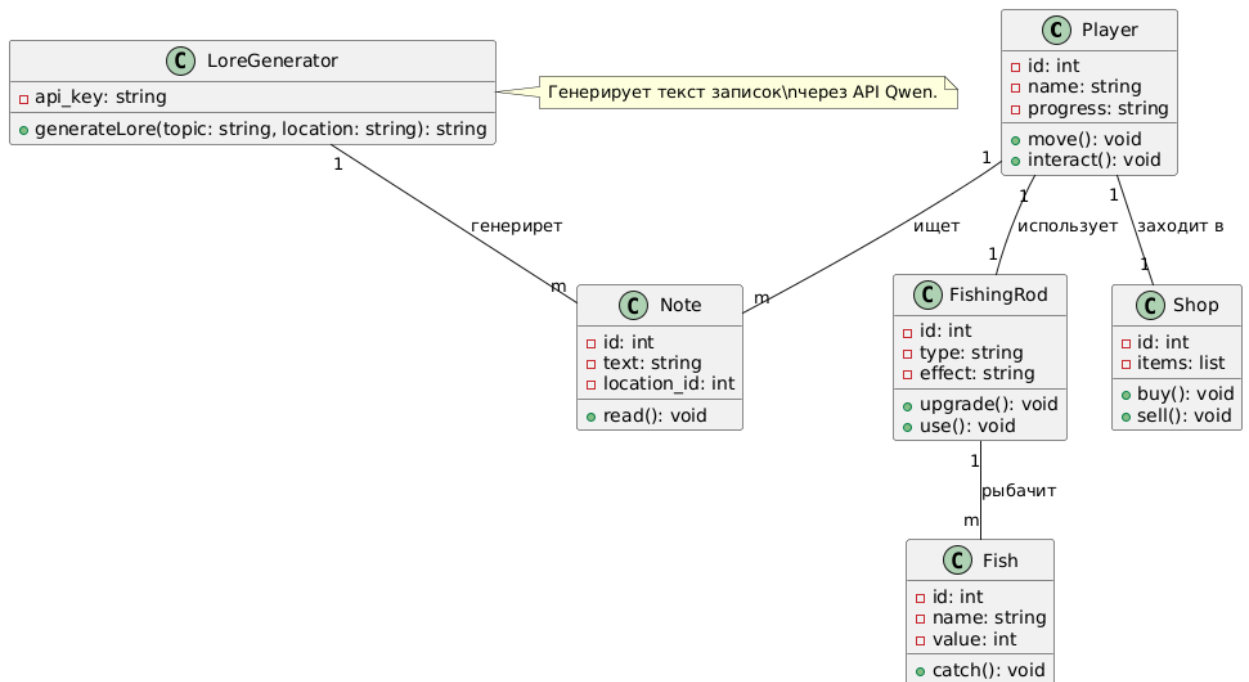
#### 7. Система коллекционирования

- Отображение собранной коллекции рыб

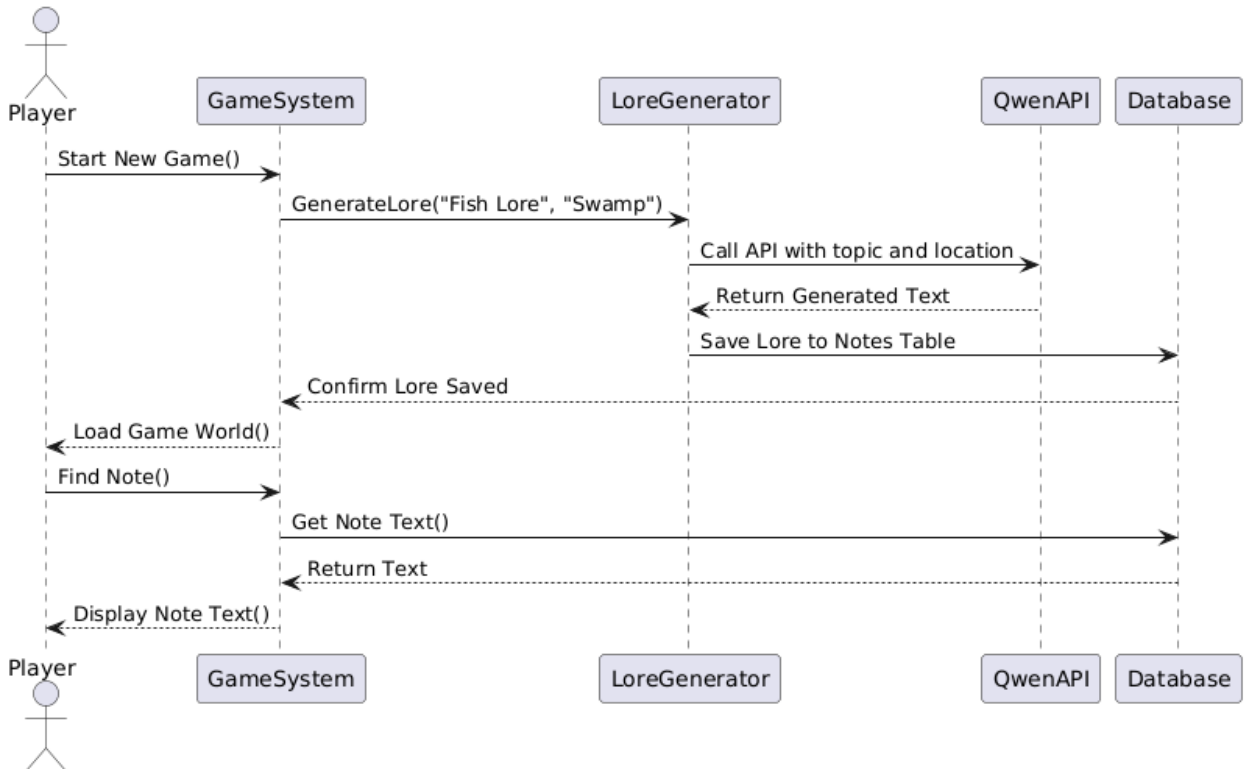
## 3 Начальная архитектура

### 3.1 UML диаграммы

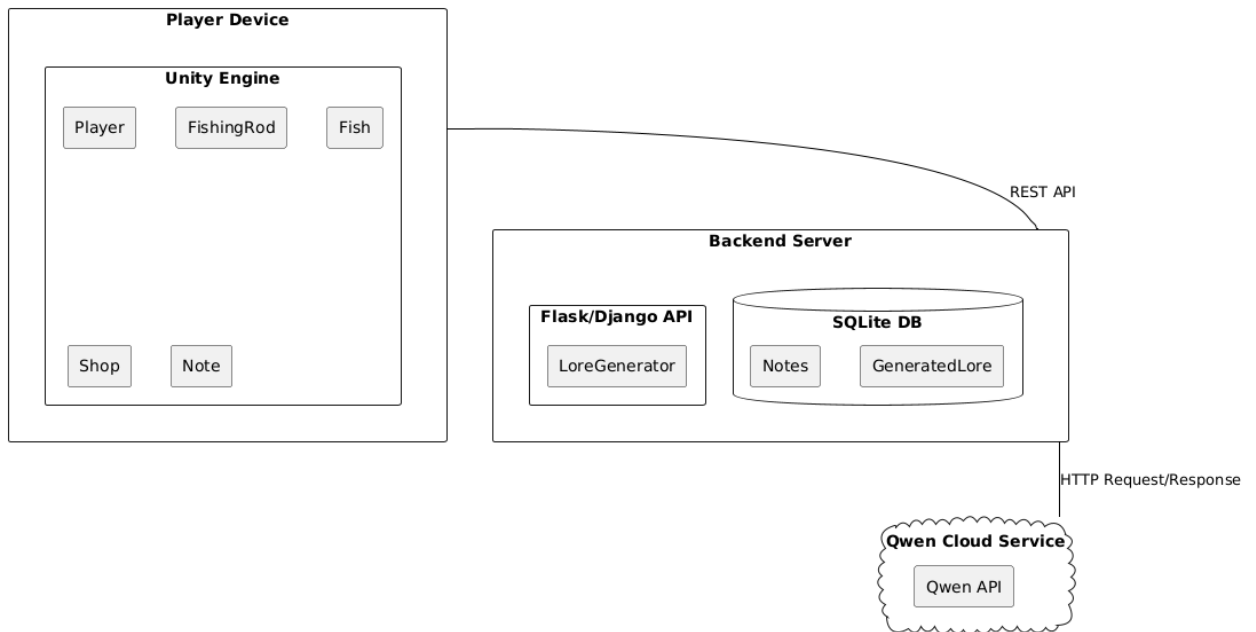
#### 3.1.1 Диаграмма классов



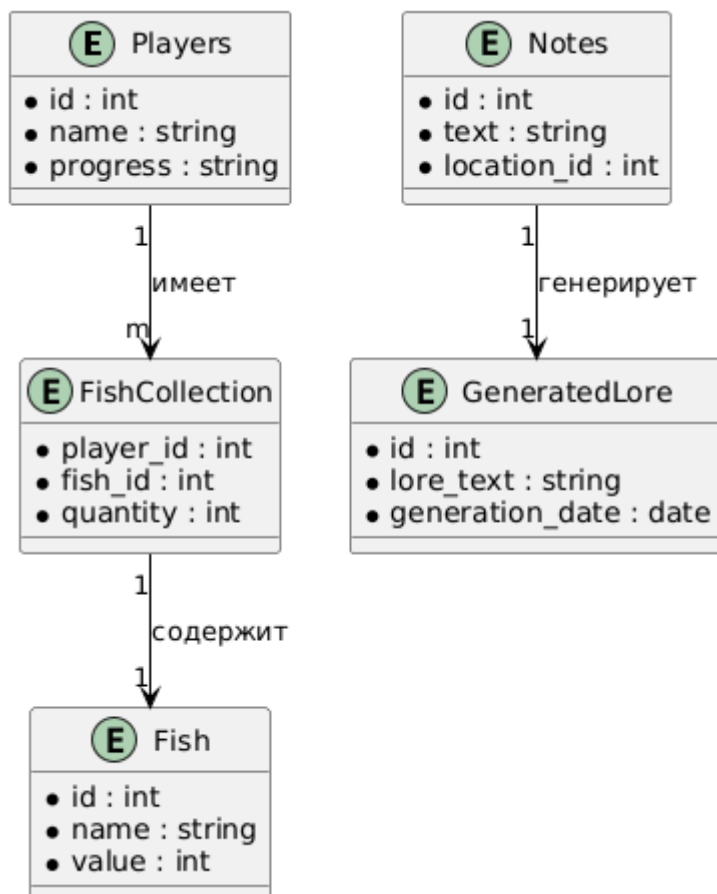
#### 3.1.2 Диаграмма последовательностей



### 3.1.3 Диаграмма развертывания



### 3.2 ER диаграмма



### **3.3 Схема API**

#### **3.3.1 Описание основных эндпоинтов**

##### **1. Аутентификация и пользователи**

POST /api/auth/register

Регистрация: создание аккаунта (логин, пароль, язык).

POST /api/auth/login

Авторизация: получение JWT-токена.

POST /api/auth/refresh

Обновление токена.

##### **2. Прогресс игрока**

POST /api/progress

Загрузка данных: локация, инвентарь, улучшения, валюта

POST /api/auth/login

Сохранение прогресса

POST /api/auth/refresh

Обновление токена.

##### **3. Магазин (игровой торговец)**

GET /api/shop/items

Получение списка товаров (наживка, улучшения).

POST /api/shop/buy

Покупка товара (проверка валюты, обновление инвентаря).

POST /api/shop/sell

Продажа рыбы (конвертация в валюту).

#### 4. Достижения

GET /api/achievements

Список разблокированных достижений.

POST /api/achievements/unlock

Разблокировка (например, при поимке 10 рыб).

#### 5. Логирование

POST /api/logs

Отправка данных о действиях игрока (ловля рыбы, переход между локациями).

#### 6. Монетизация

POST /api/payments/purchase

Обработка внутриигровых покупок

### 3.3.2 Взаимодействие компонентов

Клиент (Unity):

- Отправляет запросы на бэкенд через REST API.
- Обрабатывает ответы (обновление UI, анимации).
- Локально кэширует данные для оффлайн-доступа (с последующей синхронизацией).

Бэкенд:

- Игровой сервер (Node.js/Python/.NET Core):
  - Обрабатывает логику перемещения, взаимодействия, торговли.
  - Интегрируется с базой данных.
- Сервис генерации лора (отдельный микросервис):
  - Использует Qwen для создания уникальных записок.
- Аналитика:
  - Логирует действия игроков (ловля рыбы, покупки).

Базы данных:

- SQLite

- Таблицы: Players, Inventory, Locations, Fish, Merchants, Chests.

### **3.4 Предварительный выбор стека технологий**

#### **3.4.1 Backend**

- Язык: Python (FastAPI)
- База данных: SQLite.
- Аутентификация: JWT-токены.
- LLM-интеграция: Микросервис на Python (использует Qwen через API).

#### **3.4.2 Frontend (Игра):**

- Движок: Unity (C#).
- Локализация: Unity Localization Package.
- UI: что используется для анимаций (Unity).

#### **3.4.3 Хранение данных:**

- Локальные: SQLite
- Серверные: свой сервер

## 4 Дизайн макеты

### 4.1 UI Kit

- Цветовая палитра:
  - Основные цвета: Синий, зеленый, коричневый, оранжевый, красный, белый.
  - Дополнительные цвета: Фиолетовый, темно-синий, ярко-желтый.
  - Цветовые акценты: Использование градиентов, контрастные цвета.
- **Шрифты:** Pixelated шрифты
- Компоненты:
  - Кнопки:
    - Основные кнопки: прямоугольная форма с закругленными углами, фон — градиент между коричневым и оранжевым, текст белый.
    - Второстепенные кнопки: более светлые тона (бежевый или светло-зеленый), текст черный.
    - Активные кнопки: добавление эффекта свечения (ярко-желтый контур).
  - Меню:
    - Главное меню: фоновое изображение озера с плавающими рыбами, кнопки расположены вертикально по центру.
    - Настройки: слайдеры для регулировки громкости, переключатели для выбора режимов игры.

## 4.2 Брендбук

- Логотип:
  - Описание логотипа: Изображение кота в пиксель-арт стиле, держащего рыбу в лапах.
  - Кот выполнен в розовых, рыба — в синих.
- Фирменный стиль:
  - Общий стиль:
    - Фэнтэзи-ретро в пиксельном стиле с акцентом на природу и рыбалку.
    - Элементы дизайна вдохновлены старыми 8-битными играми, но с современной адаптацией для удобства.
  - Иконографика:
    - Все иконки выполнены в пиксель-арт стиле: удочка, рыбы, монеты.
    - Иконки имеют четкие границы и минимальное количество деталей для сохранения читаемости.
- Анимации:
  - Легкие анимации для кнопок.
  - Анимации передвижения персонажа.
- Звуки:
  - 8ми-битные мелодии для фоновой музыки.
  - Звуковые эффекты: плеск воды, кот мяукает при взаимодействии, звуки открывания дверей.
- Дополнительные элементы:



- Фирменные паттерны:
  - Пиксельные текстуры для создания объема и наложения теней.
- Элементы декора:
  - Пиксельные деревья, горы, камни, кусты, лед, трава, дома, предметы интерьера для создания атмосферы.
  - Декоративные рамки для окон и меню, выполненные в стиле деревянных конструкций.

## **5 Настройка гита**

### **5.1 Добавление в гит**

Добавить в гитхаб ссылки на документацию, таск-трекер и репозиторий с самой игрой.