

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





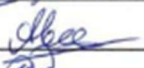


Факультет Компьютерных наук

Кафедра программирования и информационных технологий

Приключенческая игра с коллекционированием «Fishventure»

Курсовая работа

Направление: 09.03.02. Информационные системы и технологии

Зав. Кафедрой \_\_\_\_\_ д. ф.-м. н, доцент С.Д. Махортов  
Руководитель \_\_\_\_\_ ст. преподаватель В.С. Тарасов  
Руководитель практики  ассистент, М.А. Савин  
Обучающийся  А.А. Богачев-Воевудский, 3 курс, д/о  
Обучающийся  В.Е. Веремеев, 3 курс, д/о  
Обучающийся  Д.С. Штукатуров, 3 курс, д/о  
Обучающийся  А.М. Елфимов, 3 курс, д/о  
Обучающийся  Д.Ю. Дубровин, 3 курс, д/о  
Обучающийся  М.Д. Габелко, 3 курс, д/о

Воронеж 2025

# Содержание

<b>СОДЕРЖАНИЕ .....</b>	<b>2</b>
<b>ВВЕДЕНИЕ .....</b>	<b>3</b>
1.1 ПОСТАНОВКА ЗАДАЧИ .....	5
1.2 ЦЕЛЬ РАЗРАБОТКИ: .....	5
1.3 ЗАДАЧИ .....	5
1.4 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ .....	6
1.5 ОБЗОР АНАЛОГОВ .....	6
1.5.1 Stardew Valley .....	6
1.5.2 Fishing Planet .....	7
1.5.3 Fishdom .....	8
1.6 ТРЕБОВАНИЯ К ПРИЛОЖЕНИЮ .....	10
1.6.1 Функциональные .....	10
1.6.2 Нефункциональные .....	12
<b>2. ОПИСАНИЕ АРХИТЕКТУРЫ ПРИЛОЖЕНИЯ .....</b>	<b>13</b>
2.1 СУБД .....	13
2.2 СХЕМА API .....	14
2.2.1 Регистрация пользователя .....	14
2.2.2 Авторизация пользователя .....	15
2.2.3 Обновление токена .....	15
2.2.4 Отправка логов .....	16
2.2.5 Отправка метрик игрока .....	16
2.2.6 Генерация записок .....	17
2.3 ИГРОВОЙ ДВИЖОК .....	18
<b>3. РЕАЛИЗАЦИЯ .....</b>	<b>19</b>
3.1 ИГРА «FISHVENTURE» .....	19
3.1.1 Главное меню .....	19
3.1.2 Тайловая система карты .....	21
3.1.3 Мини-игра рыбалки .....	21
3.1.4 Инвентарь .....	22
3.1.5 Система наживок .....	23
3.1.6 Магазин .....	24
3.2 БЭКЕНД .....	26
3.2.1 API .....	26
3.2.2 Админ панель .....	27
3.3 ДОПОЛНИТЕЛЬНО .....	29
3.3.1 Тестирование .....	29
3.3.2 Метрики .....	29
<b>ЗАКЛЮЧЕНИЕ .....</b>	<b>30</b>

## Введение

В современном мире мобильные игры занимают одну из ведущих позиций в индустрии развлечений. Благодаря доступности и удобству использования, они привлекают аудиторию всех возрастов и интересов. Одним из популярных жанров являются приключенческие игры с элементами коллекционирования, позволяющие игроку погрузиться в увлекательный игровой процесс, исследовать мир, собирать уникальные предметы и развивать навыки персонажа.

Актуальность проекта обусловлена ростом популярности мобильных игр с несложным геймплеем, но богатым контентом. Среди множества существующих решений выделяются такие игры, как Stardew Valley , Fishing Planet и Fishdom , которые объединяют элементы рыбалки, исследования мира и развития персонажа. Однако рынок все еще открыт для новых решений, особенно если они используют современные технологии, такие как генерация уникального лора с помощью LLM (Large Language Models), что повышает вовлеченность и интерес к игре.

Целью данной работы является разработка мобильной приключенческой игры с коллекционированием под названием «Fishventure». Игра будет сочетать в себе механику рыбалки, исследование мира, улучшение оборудования и сбор уникальных видов рыб. Для реализации лора игрового мира планируется использовать языковую модель Qwen 2.5, что добавит глубины и вариативности игровому процессу. В качестве платформы выбраны мобильные устройства на базе Android, а для разработки используется игровой движок Unity.

В рамках курсовой работы будет проведено полное проектирование игры: от анализа целевой аудитории и конкурентов до разработки технического задания, архитектуры, дизайна интерфейса и организации

командной работы. Основные задачи включают создание мини-игры по ловле рыбы, системы улучшений, магазина, сохранения прогресса, коллекционирования, а также обеспечение взаимодействия с сервером через API. Особое внимание уделено юзабилити, производительности и масштабируемости, чтобы обеспечить высокое качество продукта и возможность его дальнейшего развития.

Результатом работы станет полноценная мобильная игра, которая может быть выпущена в магазине RuStore и протестирована среди целевой аудитории. Данный проект позволит применить современные подходы к разработке игр, а также углубить знания в области геймдева, дизайна интерфейсов и использования искусственного интеллекта в игровой индустрии.

## **1.1 Постановка задачи**

## **1.2 Цель разработки:**

В рамках разработки игры были поставлены следующие цели:

- Создать игру в жанре «рыбалка», сочетающую коллекционирование, исследование мира и улучшение инструментов.
- Использовать LLM для генерации уникального лора игрового мира для мотивирования игрока исследовать его.
- Монетизировать игру путем единоразовой покупки.

## **1.3 Задачи**

Для достижения поставленных целей были обозначены следующие задачи:

- Разработать игру на движке Unity, сочетающую рыбалку, коллекционирование, исследование мира.
- Реализовать механизм генерации лора мира игры в форме записок на основе LLM Qwen 2.5.
- Выпустить игру в качестве платного продукта в магазин мобильных приложений RuStore.

## **1.4 Анализ предметной области**

## **1.5 Обзор аналогов**

На рынке существует множество игры связанные с тематикой рыбалки. В рамках данной курсовой работы будут рассмотрены следующие игры:

— Stardew Valley

— Fishing Planet

— Fishdom

Эти игры пользуются популярностью. Далее они будут рассмотрены подробнее.

### **1.5.1 Stardew Valley**

Это игра про персонажа из города, который узнает о полученной в наследство от дедушки фермы. Ферма находится далеко от города в поселении Пеликан. Тут игрока ждут множество историй

В игре существуют множество занятий, одним из которых является рыбалка. Механика достаточно проста в идеи и приятно ощущается в игре. Но эта механика не является обязательной в игре Stardew Valley.

Игра вышла в 2019 году. Она не имеет внутриигровых покупок, только разовая оплата за покупку игры. Для игры характерен пиксель-арт и RPG элементы.

Stardew Valley вышла на ПК, но через время вышла и на мобильные операционные системы Android/iOS.

### 1.5.2 Fishing Planet

Fishing Planet — это реалистичный многопользовательский онлайн-симулятор рыбалки от первого лица, выполненный с акцентом на детализацию и имитацию настоящих условий ловли рыбы. Игрок погружается в атмосферу живого водоема, где важно учитывать множество факторов: время суток, погодные условия, силу и направление ветра, уровень освещенности и даже сезонные изменения в поведении рыбы.

Игровой процесс строится вокруг поиска подходящего места для рыбалки, выбора правильной снасти, наживки и тактики ловли в зависимости от вида целевой рыбы. Механика игры максимально приближена к реальности: игрок должен уметь читать поведение воды, правильно забрасывать приманку, чувствовать поклевку и грамотно вываживать трофей.

Особенностью Fishing Planet является динамическая система окружающей среды: вода реагирует на ветер, течения и глубину, что влияет на поведение рыб и эффективность используемых методов ловли. Это делает игру не просто развлечением, но и своеобразным тренажером для начинающих и опытных рыболовов.

Игра была выпущена изначально для ПК, а затем получила мобильную версию, доступную на платформах Android и iOS. Fishing Planet не предполагает внутриигровых покупок — игрок оплачивает игру единовременно и получает полный доступ ко всем локациям и возможностям.

Данный проект ориентирован на любителей спокойного, медитативного игрового процесса и ценителей реализма в симуляторах. Однако из-за высокой сложности и требовательности к знанию особенностей рыбалки, он может быть менее привлекателен для широкой аудитории, предпочитающей более простые и казуальные механики.

### 1.5.3 Fishdom

Fishdom — это игра, сочетающая в себе механику головоломок типа «три в ряд» и уход за аквариумом. Разработанная студией Playrix, она предлагает игрокам чередовать решение логических задач с возможностью украшать и развивать собственный подводный мир.

Игровой процесс состоит из двух основных частей: прохождение уровней и развитие аквариума. В первой части игрок решает головоломки, переставляя элементы на поле для создания комбинаций. Каждый уровень имеет свои цели и ограничения, что добавляет разнообразия и требует стратегического подхода. Во второй части, за успешное прохождение уровней, игрок получает монеты, которые можно потратить на покупку новых рыбок, декораций и украшений для аквариума. Также есть возможность кормить рыб, наблюдать за их поведением и даже получать от них бонусы для следующих уровней.

Монетизация игры построена на внутриигровых покупках и рекламе. Игрок может приобретать игровые монеты, специальные ускорители и уникальные предметы для аквариума. Доступны сезонные пропуски, предоставляющие дополнительные награды за выполнение заданий. Такая модель позволяет играть бесплатно, но одновременно стимулирует траты ради улучшенного игрового опыта.

Fishdom была выпущена как для ПК, так и для мобильных устройств на базе iOS и Android, что сделало её доступной широкому кругу пользователей. Игра позиционирует себя как казуальный продукт, ориентированный на расслабляющий геймплей и простоту освоения.

Однако со временем повторяемость уровней и зависимость от платной монетизации могут снижать интерес к игре. Тем не менее, Fishdom остаётся ярким примером успешного сочетания головоломки и симулятора ухода за



питомцами, что делает её интересной для анализа при разработке игровых механик, направленных на регулярное возвращение игроков.

## 1.6 Требования к приложению

### 1.6.1 Функциональные

В игре реализованы следующие требования:

- **Мини-игра рыбалки:** позволяет игроку ловить рыбу через выполнение интерактивной механики рыбалки, включающей: шкалу прогресса, последовательность стрелочек и таймер. Шкала прогресса отображает текущий прогресс ловли рыбы. Игрок должен повторять последовательность стрелочек в заданном порядке, выполнение каждой последовательности ограничено таймером (один цикл). Процент правильного выполнения последовательности влияет на заполнение или опустошение шкалы прогресса. Если шкала полностью заполняется, рыба считается пойманной; если она становится пустой, рыба срывается.
- **Параметры сложности рыбалки для каждой рыбы:** для каждого вида рыбы задаются параметры, определяющие сложность её ловли. Эти параметры включают штраф за провал цикла (на сколько уменьшается шкала прогресса при ошибке), очки прогресса за успешный цикл (на сколько заполняется шкала прогресса при правильном выполнении), время на выполнение цикла (длительность таймера) и длину последовательности стрелочек (количество действий, которые игрок должен выполнить).
- **Возможность начать ловлю рыбы с любой части водоёма:** возможность начать рыбалку с любого тайла суши, прилегающего к водоёму.

- **Выбор наживки:** возможность покупать разные виды наживки в магазине и выбирать их перед началом рыбалки. Выбор наживки влияет на вероятность ловли определённых видов рыб.
- **Продажа рыбы:** продавать пойманную рыбу в магазине за внутриигровую валюту. Цена рыбы будет зависеть от ее вида.
- **Исследование мира:** для мотивации игрока исследовать локации в них размещаются сундуки, содержащие уникальные улучшения и записки с историей мира (лор). Сундуки расставлены по локации в заранее определённых точках. Некоторые сундуки доступны сразу, а другие требуют решения головоломок или выполнения дополнительных условий. Чтобы открыть сундук, необходимо подойти на соседний от него тайл.
- **Система инвентаря:** в инвентаре персонажа хранятся все пойманные рыбы. Возможность продавать рыбу в магазине за внутриигровую валюту.
- **Механика тайловой карты:** карта реализуется в виде 2D-сетки тайлов с координатами (x, y), где каждый тайл содержит информацию о типе тайла: вода, суша и т.п. А также данные об интерактивных объектах, таких как сундуки или декорации. Тайлы используются для определения взаимодействий игрока с окружающей средой и обеспечивают базовую структуру для размещения объектов и управления игровым процессом.
- **Механика взаимодействия:** взаимодействие с объектами происходит через проверку соседних тайлов в направлении взгляда игрока. Радиус взаимодействия составляет один тайл от текущей позиции.

- **Система лора мира:** в сундуках будут лежать записки, текст которых генерируется при старте новой игры с применением LLM Qwen 2.5 через бесплатный API.

### 1.6.2 Нефункциональные

К данной игре были предъявлены следующие нефункциональные требования:

- **Масштабируемость:** начальная архитектура должна закладывать простоту масштабируемости. Добавление новых видов рыб и локаций должно быть возможно с незначительными изменениями кодовой базы.
- **Производительность:** игра должна обеспечивать работоспособность и минимальную частоту кадров в 30 FPS на устройствах с ОС Android версии 10+.
- **Безопасность:** реализация системы аутентификации с использованием токенов.

Использование токенов авторизации вместо пароль позволяет минимизировать передачу пароля по сети.

## 2. Описание архитектуры приложения

### 2.1 СУБД

В качестве СУБД используется MySQL — одна из самых популярных и устойчиво развивающихся реляционных СУБД в мире. Она широко используется как в малых, так и в крупных проектах благодаря своей надежности, производительности и поддержке множества платформ.

Одним из главных преимуществ MySQL является её высокая производительность при работе с большими объемами данных, что делает её подходящим выбором для серверной части мобильного приложения, где важна скорость обработки запросов. Также стоит отметить простоту установки и настройки, особенно в связке с такими фреймворками, как FastAPI, что ускоряет процесс разработки backend-части. MySQL имеет хорошо проработанную систему безопасности, включая гибкую настройку прав доступа и шифрование данных, что особенно важно для хранения пользовательских учетных записей и игровой статистики.

Кроме того, MySQL обладает широким сообществом разработчиков и большим количеством документации, что облегчает поиск решений при возникновении проблем. Также она поддерживает работу с транзакциями и обеспечивает целостность данных, что критично для корректной работы игрового сервера.

Однако у MySQL есть и недостатки. Например, по сравнению с более современными СУБД, такими как PostgreSQL, она имеет ограниченный набор типов данных, что может усложнить реализацию сложных логических структур. Также в некоторых случаях могут возникнуть трудности с масштабированием и оптимизацией запросов, особенно если не соблюдены лучшие практики проектирования базы данных. Некоторые механизмы оптимизации требуют глубокого знания внутреннего устройства СУБД, что может быть сложно для начинающих разработчиков.

Тем не менее, учитывая требования проекта, такие как хранение пользовательских данных, игровой статистики и логов, а также необходимость интеграции с LLM-сервисами, MySQL остаётся оптимальным выбором благодаря своей устойчивости, скорости и совместимости с используемыми технологиями.

## 2.2 Схема API

Для авторизации и регистрации игроков было разработано API. Оно также позволяет отправлять логи на сервер и сохранять статистику о игровых сеансах в качестве метрик.

Само API реализовано на языке программирования Python, на основе библиотеки FastAPI

### 2.2.1 Регистрация пользователя

Создание нового аккаунта пользователя с указанием логина, пароля и даты рождения:

— Эндпоинт: POST /api/v1/auth/register

— Заголовки: Content-Type: application/json

— Тело запроса: {"login": <string>, "password": <string>, "birth\_date": <string>}

— Ответы:

а) Если успешно: HTTP Status 201. JSON ответ: {"auth\_token": <string>}, если регистрация прошла успешно

б) Если с ошибками: HTTP Status 400.  
JSON ответ: {**“error”**: <message>}, если не получилось зарегистрировать

### 2.2.2 Авторизация пользователя

Для авторизации необходимо передать логин и пароль:

— Эндпоинт: POST /api/v1/auth/login

— Заголовки: Content-Type: application/json

— Тело запроса: {**“login”**: <string>, **“password”**: <string>}

— Ответы:

а) Если успешно: HTTP Status 200.  
JSON ответ: {**“auth\_token”**: <string>}, если авторизация прошла успешно

б) Если с ошибками: HTTP Status 403. JSON ответ: {**“error”**: <string>}, если не получилось авторизироваться

### 2.2.3 Обновление токена

Получение нового токена авторизации, необходимо передать действующий токен авторизации и логин пользователя.

— Эндпоинт: POST /api/v1/auth/refresh

— Заголовки: Content-Type: application/json

— Тело запроса: {"**auth\_token**": <string>, "**login**": <string>}

— Ответы:

а) Если успешно: HTTP Status 200.

JSON ответ: {"**auth\_token**": <string>}, если успешный

б) Если с ошибками: HTTP Status 403.

JSON ответ: {"**error**": <string>}, если токен или логин не действителен

## 2.2.4 Отправка логов

Для отправки записи лога на сервер

— Эндпоинт: POST /api/v1/logs

— Заголовки: Content-Type: application/json

— Тело запроса: {"**auth\_token**": <string>, "**login**": <string>, "**log\_text**": <string>}

— Ответы:

а) Если успешно: HTTP Status 200.

б) Если с ошибками авторизации: HTTP Status 403.

JSON ответ: {"**error**": <string>}, если токен или логин не действителен.

## 2.2.5 Отправка метрик игрока



Для отправки времени входа в игру и выхода из игры в БД на сервере.

— Эндпоинт: POST /api/v1/player\_logs

— Заголовки: Content-Type: application/json

— Тело запроса: {"auth\_token": <string>, "login": <string>, "entered\_at": <string>, "exit\_at": <string>}

— Ответы:

а) Если успешно: HTTP Status 200.

б) Если с ошибками авторизации: HTTP Status 403.  
JSON ответ: {"error": <string>}, если токен или логин не действителен.

в) Если с ошибками даты и времени: HTTP Status 400.  
JSON ответ: {"error": <string>}, если даты и время переданы не корректно.

## 2.2.6 Генерация записок

Генерирует текст записки по переданному промту.

— Эндпоинт: POST /api/v1/generate\_note

— Заголовки: Content-Type: application/json

— Тело запроса: {"auth\_token": <string>, "login": <string>, "prompt\_text": <string>}

— Ответы:

а) Если успешно: HTTP Status 200.

JSON ответ: {"note\_text": <string>}

б) Если с ошибками авторизации: HTTP Status 403.

JSON ответ: {"error": <string>}, если токен или логин не действителен

## 2.3 Игровой движок

В качестве игрового движка был выбран Unity.

Это популярный игровой движок для 3Д игр с поддержкой кроссплатформенности. Скрипты для него пишутся на языке программирования C#.

Несмотря на то, что он известен в массах как 3Д движок, на нем нередко делают и 2Д игры.

### 3. Реализация

#### 3.1 Игра «Fishventure»

В игре реализованы множество механик, включая генерацию записок через LLM Qwen 2.5, авторизацию и регистрацию, и отправку метрик.

##### 3.1.1 Главное меню

На рисунке 1 изображено главное меню. В него игрок попадает при запуске игры.

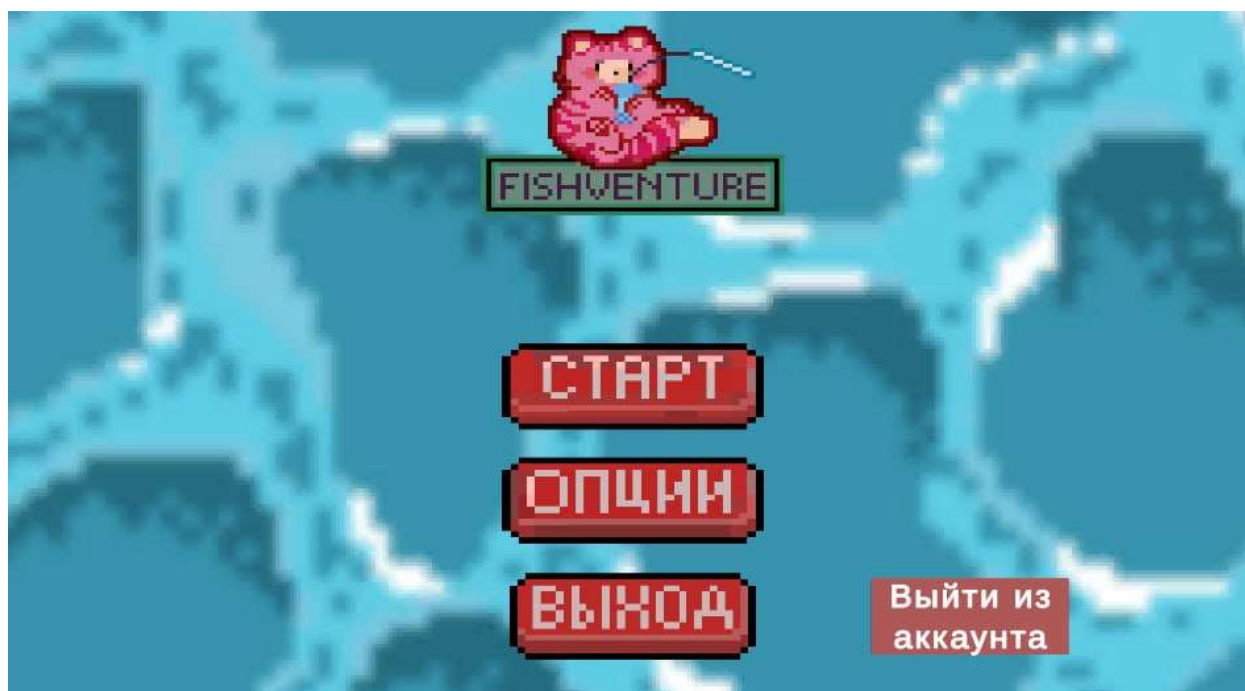
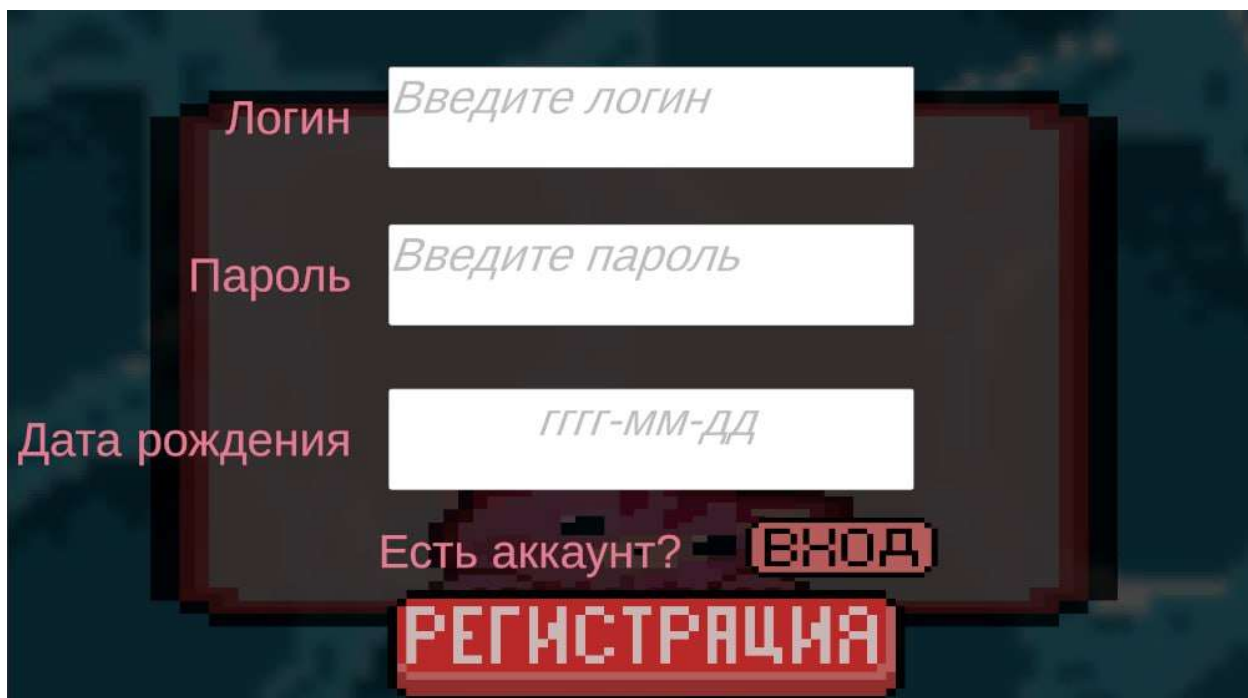


Рисунок 1 - Главное меню

Но если игрок заходит в игру впервые, то он попадает на форму регистрации/авторизации перед главным меню. Она показана на рисунке 2



Логин *Введите логин*

Пароль *Введите пароль*

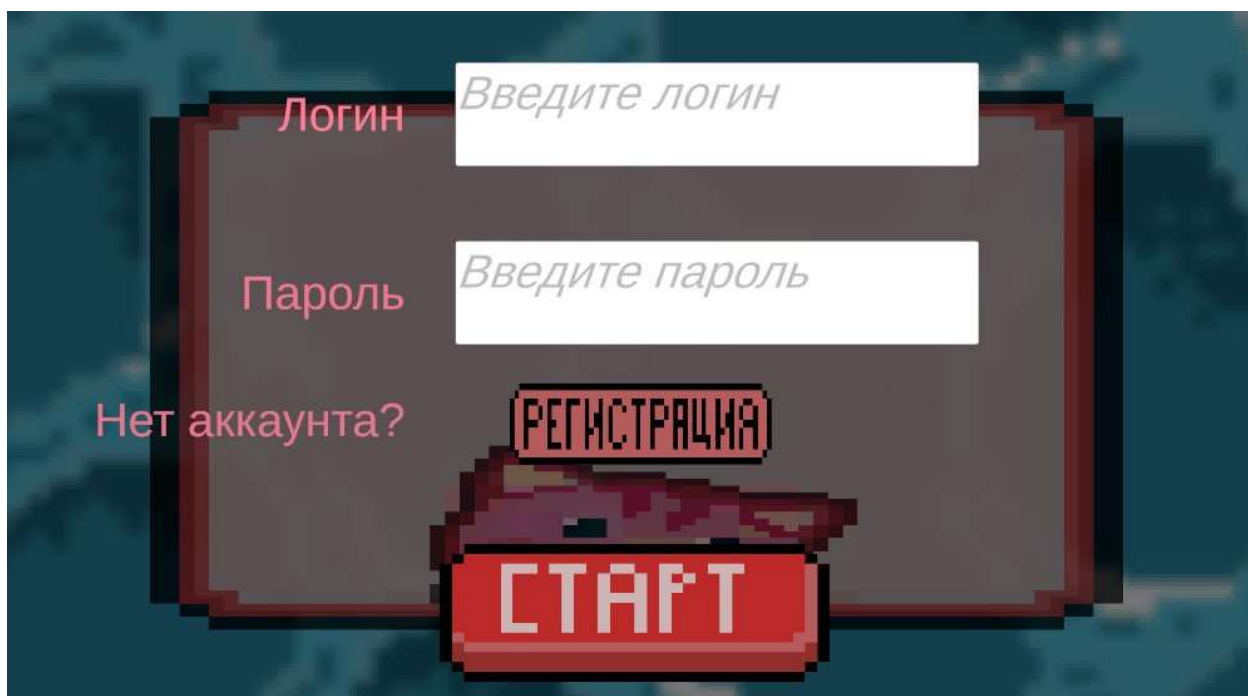
Дата рождения *гггг-мм-дд*

Есть аккаунт? **ВХОД**

**РЕГИСТРАЦИЯ**

The registration form is displayed on a dark, textured background. It features three input fields for 'Логин' (Login), 'Пароль' (Password), and 'Дата рождения' (Date of birth). The date field has a placeholder 'гггг-мм-дд'. Below the inputs, there is a link 'Есть аккаунт?' (Do you have an account?) and a button 'ВХОД' (Login). At the bottom, a large red button with white text says 'РЕГИСТРАЦИЯ' (Registration).

Рисунок 2 - Форма регистрации



Логин *Введите логин*

Пароль *Введите пароль*

Нет аккаунта? **РЕГИСТРАЦИЯ**

**СТАРТ**

The authorization form is displayed on a dark, textured background. It features two input fields for 'Логин' (Login) and 'Пароль' (Password). Below the inputs, there is a link 'Нет аккаунта?' (No account?) and a button 'РЕГИСТРАЦИЯ' (Registration). At the bottom, a large red button with white text says 'СТАРТ' (Start).

Рисунок 3 - Форма авторизации

### 3.1.2 Тайловая система карты

На рисунке 4 показано, как выглядит мир игры, и персонаж в нем. Мир представляет собой матричное поле из квадратиков, называемые тайлами.



Рисунок 4 - Мир игры

Такая система позволяет довольно просто проверять столкновения персонажа с объектами, наличие рядом тайлов воды.

Для такой структуре в Unity существует множество инструментов для создания и редактирования локаций.

### 3.1.3 Мини-игра рыбалки

Мини-игра рыбалки запускается при поклевке рыбы. На рисунке 5 продемонстрировано, как это выглядит в игре.



Рисунок 5 - Мини-игра рыбалки

Игра состоит из циклов, которые отсчитывает таймер в левом верхнем углу. За время этого таймера требуется провести по экрану в указанном стрелочками направлении в порядке с лева на право.

Если игрок успевает прожать всю последовательность стрелочек, то заполняется шкала прогресса — горизонтальная шкала вверху экрана. Соответственно, в случае, когда время таймера вышло, но не все стрелочки прожаты — шкала частично опустошается.

Если шкала становится полностью пустой, то рыба срывается и тратит наживку. Обратная ситуация, когда шкала заполнена — рыба поймана и добавляется в инвентарь.

### 3.1.4 Инвентарь

Для того, чтобы хранить пойманную рыбу, в игре предусмотрен инвентарь. Он показан на рисунке 6.



Рисунок 6 - Инвентарь

В верхних ячейках хранятся пойманные рыбы, во второй строке – наживки для удочки, а в нижней — найденные записки с историями о мире игры.

При нажатии на наживку она насаживается на удочку — появляется в отдельной ячейке справа.

Если нажать на крестик, то инвентарь закроется.

### 3.1.5 Система наживок

При использовании наживки более высокого уровня, повышается шанс поймать более ценную и редкую рыбу.



Рисунок 7 - Кузнечик в мире игры

На рисунке 7 показан кузнечик, которого можно встретить в мире и подобрать. Он появится в инвентаре, где его можно выбрать в качестве наживки.

### 3.1.6 Магазин

Пойманную рыбу можно продавать в прилавке с весами. Он показана на рисунке 8.





Рисунок 8 - Ларек продажи рыбы

Рыба продается автоматически, если подойти к прилавку.

Так же есть ларек с котом-продавцом, он показан на рисунке 9. Это магазин наживок, в который можно купить наживки разных уровней.



Рисунок 9 - Магазин наживки

Меню магазина показано на рисунке 10.



Рисунок 10 - Меню магазина

## 3.2 Бэкенд

### 3.2.1 API

API реализовано на FastAPI, оно предоставляет эндпоинты для регистрации/авторизации, генерации записок с лором игры через LLM Qwen 2.5, отправки логов и метрик об игроке.

На рисунке 11 показан Swagger для существующих эндпоинтов.

Auth ^		
POST	/auth/register	Register
POST	/auth/login	Login
POST	/auth/refresh	Refresh
Logs ^		
GET	/logs/	List Logs
POST	/logs/	Create Log
GET	/logs/{log_id}	Get Log
PlayerLogs ^		
GET	/player_logs/	List Player Logs
POST	/player_logs/	Create Player Log
GET	/player_logs/{log_id}	Get Player Log
Qwen ^		
POST	/qwen/generate	Generate Text

Рисунок 11 - Swagger

### 3.2.2 Админ панель

Админ панель написана на основе фреймворка Django. Она позволяет смотреть, редактировать и удалять записи из таблиц в базе данных. На рисунке 12 показана одна страница админ панели.

Через эту панель можно управлять таблицами игроков, их метрик и общими логами.

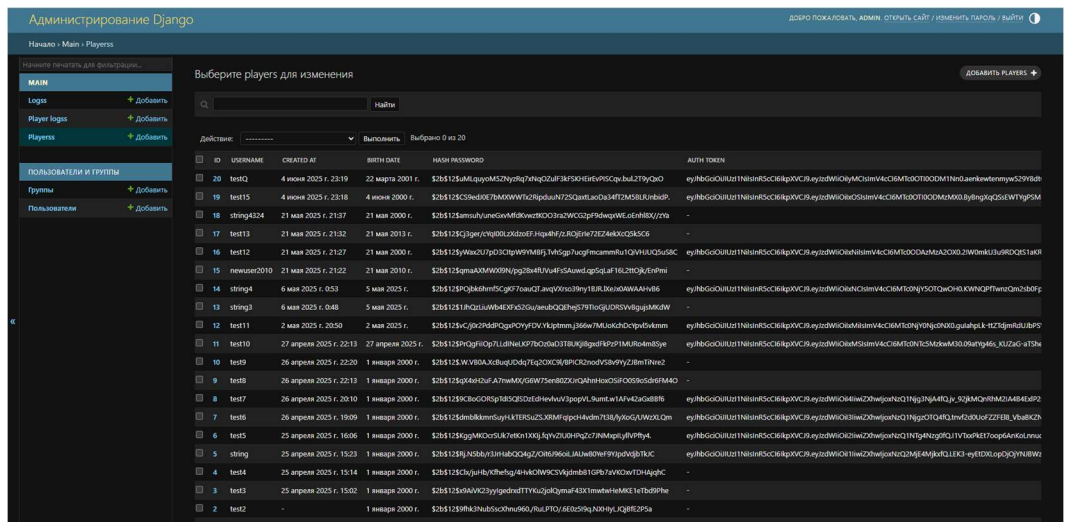


Рисунок 12 - Админ панель

### **3.3 Дополнительно**

#### **3.3.1 Тестирование**

Для API прописаны юнит тесты, которое автоматически запускаются в процессе CI/CD при разворачивании на сервере.

На каждый эндпоинт прописаны юнит тесты, которые проверяют множество сценариев поступающих на них запросов.

#### **3.3.2 Метрики**

При заходе в игру сохраняется время начала сеанса игры локально на устройстве. В момент выхода из игры на API сервера отправляется запрос на сохранение метрик. Передаются логин, токен авторизации, время входа и время выхода.

На основе этих данных можно формировать аналитику о среднем времени сеансов для каждого игрока, время пиковой активности игроков.

## Заключение

В рамках данной курсовой работы была спроектирована и реализована мобильная приключенческая игра с элементами коллекционирования под названием Fishventure. Игра сочетает в себе механику рыбалки, исследование мира, улучшение оборудования и сбор уникальных видов рыб. Особое внимание было уделено использованию современных технологий, включая интеграцию LLM-модели Qwen 2.5 для генерации уникального лора игрового мира, что позволило повысить реиграбельность и вовлечённость игроков.

Был проведён анализ существующих аналогов, таких как Stardew Valley Fishing Planet и Fishdom, на основе которого были определены ключевые особенности и отличия разрабатываемой игры. Также были изучены целевая аудитория и рынок, что дало возможность создать продукт, ориентированный на интересы потенциальных пользователей. В качестве платформы для разработки была выбрана мобильная экосистема Android, а в качестве игрового движка — Unity, что обеспечило кроссплатформенность и удобство разработки.

Техническая часть проекта включала проектирование API, реализацию авторизации и регистрации, работу с базой данных через СУБД MySQL, а также разработку клиентской части игры с такими компонентами, как мини-игра по ловле рыбы, система улучшений, магазин, инвентарь и тайловая система мира. Для хранения пользовательских данных, метрик и логов был реализован серверный стек на основе FastAPI и Django.

Особый акцент был сделан на масштабируемости, производительности и безопасности системы. Архитектура приложения позволяет легко добавлять новые виды рыб, локации и игровые механики без значительных изменений в кодовой базе. Также были реализованы механизмы сохранения прогресса

игрока, отправка метрик и логов, а также генерация записок с историей мира с помощью языковой модели.

В результате выполнения проекта была создана полноценная прототипная версия мобильной игры, которая может быть доработана и выпущена в магазине RuStore. Проведённое тестирование показало работоспособность основных функциональных блоков, а также соответствие требованиям к производительности и юзабилити.

Выполнение данной курсовой работы позволило не только применить теоретические знания на практике, но и получить опыт в разработке полного цикла игровых приложений, начиная от анализа и проектирования, заканчивая реализацией и тестированием. Проект продемонстрировал возможности использования искусственного интеллекта в игровой индустрии, а также важность правильного выбора технологического стека и подходов к архитектуре программного обеспечения.

Таким образом, разработка игры Fishventure стала успешным примером практической реализации современных подходов к созданию интерактивных приложений и может служить основой для дальнейшего развития в направлении полноценного коммерческого продукта.