

**Министерство науки и высшего образования Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**  
**(Университет ИТМО)**

Факультет      **Прикладной информатики**

Направление подготовки **09.03.03 Прикладная информатика**

Образовательная программа **Мобильные и сетевые технологии**

## **КУРСОВОЙ ПРОЕКТ**

Тема: «Разработка прототипа веб-приложения для поиска напарников для видеоигр»

Обучающийся: Янченко Денис Сергеевич, К3141

Санкт-Петербург 2024

## **СОДЕРЖАНИЕ**

<b>ВВЕДЕНИЕ</b>	<b>3</b>
1 Суть проекта	5
2 Процесс работы над всем проектом	7
3 Моя роль в проекте	10
3.1 Поставленные задачи	10
3.2 Решение поставленных задач	12
4 Анализ выполненной работы	15
5 Взаимодействие в проекте	17
<b>ЗАКЛЮЧЕНИЕ</b>	<b>19</b>
<b>СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ</b>	<b>20</b>
<b>ПРИЛОЖЕНИЕ</b>	<b>21</b>

## **ВВЕДЕНИЕ**

В наше время цифровые технологии играют важную роль в повседневной жизни, а видеоигры становятся неотъемлемой частью досуга миллионов людей. Совместная игра с напарниками позволяет не только улучшить игровой опыт, но и укрепить социальные связи. Однако найти подходящих партнеров для игр, особенно с учетом навыков, времени активности и других предпочтений, является сложной задачей. Существующие платформы, такие как форумы или случайные лобби, не всегда предоставляют удобные механизмы поиска, что делает актуальным создание специализированного решения.

Проект, разработанный нашей командой, направлен на создание веб-приложения, которое позволит пользователям находить напарников для совместной игры, используя современные подходы к фильтрации и системе репутации. Это приложение будет полезно как опытным игрокам, ищущим надёжных партнёров для сложных игровых заданий, так и новичкам, которые хотят легче влиться в игровое сообщество.

Цель проекта: разработка функционального прототипа веб-приложения, обеспечивающего удобный поиск напарников для видеоигр с использованием различных фильтров и системы рейтингов.

Задачи проекта:

1. Исследовать существующие платформы для поиска игровых напарников, выявить их ключевые функции и предложить улучшения.
2. Разработать макеты пользовательского интерфейса, включая страницы профиля, поиска и фильтрации.
3. Спроектировать и настроить базу данных для хранения профилей пользователей, их предпочтений и истории взаимодействий.
4. Реализовать регистрацию и авторизацию пользователей с соблюдением требований безопасности.

5. Создать функционал профиля с возможностью указывать игровые интересы и предпочтения.
6. Реализовать поиск напарников с учётом фильтров по играм, уровню навыков, времени доступности и другим параметрам.
7. Внедрить систему уведомлений о заявках, приглашениях и новых сообщениях.
8. Тестировать и дорабатывать функционал для улучшения пользовательского опыта.

## 1 Суть проекта

Проект "Разработка прототипа веб-приложения для поиска напарников для видеоигр" направлен на создание удобного и интуитивно понятного инструмента для игроков, желающих найти партнёров для совместных игр. Веб-приложение предоставляет пользователям возможность не только создавать и управлять своими профилями, но и искать подходящих напарников на основе различных фильтров.

Основные функции приложения включают:

1. Создание и управление профилем: Пользователи могут создавать персонализированные профили, указывая свои игровые предпочтения, опыт, а также информацию о любимых играх и жанрах.
2. Система поиска и фильтрации: Встроенная система поиска позволяет пользователям находить потенциальных партнёров по заданным критериям, таким как выбранные игры, уровень опыта или время, в которое они доступны для игры.
3. Механизм взаимодействия между пользователями: После нахождения подходящего напарника пользователь может отправить запрос на совместную игру. Приложение уведомляет о новых запросах и приглашениях, чтобы обеспечить своевременную коммуникацию между участниками.
4. Система репутации: Для повышения доверия между пользователями реализована система репутации, позволяющая оставлять отзывы и оценки после совместных игровых сессий. Это помогает выделять надёжных и доброжелательных игроков.
5. Регистрация и авторизация пользователей: Для обеспечения безопасности и персонализированного опыта, в приложении реализованы функции регистрации и авторизации пользователей, включая возможность входа через социальные сети.

Данный проект представляет собой прототип, который можно развить и дополнить новыми функциями в зависимости от обратной связи и требований

пользователей. В будущем, приложение может стать важным инструментом для игроков, улучшая их опыт и создавая новые возможности для совместных развлечений.

## 2 Процесс работы над всем проектом

Работа над проектом "Разработка прототипа веб-приложения для поиска напарников для видеоигр" включала несколько ключевых этапов, каждый из которых был направлен на решение определённых задач и достижение целей проекта.

### Этапы работы

- Сбор требований и анализ:
  - Выполнен анализ существующих решений на рынке, что позволило выделить уникальные функции для будущего продукта. В частности, акцент сделан на удобстве поиска и возможности взаимодействия между игроками.
  - Итогом этапа стал документ с функциональными и нефункциональными требованиями, а также сформированная концепция приложения.
- Проектирование интерфейса и дизайн:
  - На основе собранных требований были разработаны макеты интерфейса в Figma. Для каждой ключевой функции (поиск, профиль, взаимодействие) созданы визуальные схемы.
  - Макеты тестировались на пользователях для получения обратной связи, после чего дизайн дорабатывался. Например, были упрощены фильтры поиска и добавлена возможность быстрого просмотра профилей.
  - Обеспечена адаптивность интерфейса для корректного отображения на различных устройствах.

- Проектирование архитектуры и настройка среды разработки:
  - Архитектура приложения была спроектирована с учетом разделения на серверную часть (Backend) и клиентскую часть (Frontend).
  - Серверная часть реализована с использованием Ruby on Rails благодаря его удобству для создания RESTful API [1].
  - Клиентская часть разработана на React, что позволило обеспечить динамичность интерфейса.
  - Настроена база данных PostgreSQL для хранения профилей пользователей, запросов и взаимодействий между игроками [3].
  - Создан репозиторий на GitHub для управления кодом, настройка CI/CD для автоматической проверки и развертывания кода.
- Разработка основных модулей приложения:
  - Регистрация и авторизация пользователей: реализована система безопасного входа с использованием JWT-токенов.
  - Создание профиля: пользователи могут добавлять информацию о себе, выбирать интересующие их игры и указывать предпочтительные критерии для поиска напарников.
  - Поиск и фильтрация: реализован алгоритм быстрого поиска с возможностью задавать фильтры по уровню опыта, типу игры и времени активности.
  - Взаимодействие пользователей: добавлены функции отправки запросов на совместную игру, система уведомлений о новых запросах и принятых приглашениях.
  - Репутация: внедрена система рейтинга, позволяющая оставлять отзывы и ставить оценки другим пользователям после завершения совместной игры.
- Тестирование и отладка:
  - Выполнено функциональное тестирование, включающее проверку всех основных функций приложения.



- Выявленные баги были зафиксированы в баг-трекере, а затем устранены разработчиками.

### **3 Моя роль в проекте**

#### **3.1 Поставленные задачи**

В рамках работы над проектом "Разработка прототипа веб-приложения для поиска напарников для видеоигр" передо мной была поставлена задача разработки ключевых элементов серверной части приложения. Основным вызовом заключался в необходимости создания функциональной и надежной системы, способной обеспечить регистрацию, авторизацию пользователей, а также удобный поиск напарников с применением различных фильтров.

Для реализации указанных целей была определена последовательность задач:

1. Проектирование схемы базы данных (БД):  
Разработать структуру БД, которая поддерживала бы хранение данных о пользователях, играх, уровнях опыта, времени доступности и системе репутации, а также обеспечивала бы целостность и оптимизацию данных.
2. Создание модели и контроллера User с авторизацией через Devise:  
Настроить механизм управления пользователями с использованием готового решения Devise, обеспечивающего аутентификацию и управление профилями.
3. Регистрация и авторизация пользователей:  
Реализовать функциональность для безопасной регистрации, входа и выхода пользователей, а также восстановления паролей.
4. Создание системы поиска напарников:  
Разработать интерфейс и серверную логику для подбора напарников на основе заданных пользователями критериев, таких как предпочитаемые игры, уровень, время активности и другие параметры.
5. Обеспечение безопасности приложения:  
Внедрить базовые функции защиты, включая шифрование паролей,

защиту от CSRF-атак, подтверждение адреса электронной почты и ограничение количества попыток входа.

### 3.2 Решение поставленных задач

Работа над серверной частью началась с проектирования схемы базы данных. Этот этап был выполнен совместно с коллегой по backend-разработке, чтобы обеспечить согласованность структуры данных и их использования в приложении.

- Структура базы данных:
  - Основной таблицей является таблица `user`, которая содержит информацию о пользователях, включая их никнейм, возраст, пол, email, пароль, а также связанные параметры, такие как предпочтения в играх, репутация и уведомления.
  - Таблица `games` хранит список доступных игр, что позволяет пользователям выбирать и указывать свои предпочтения.
  - Для связи пользователей с играми используется таблица `user_games`, которая дополнительно хранит игровую статистику.
  - Таблица `reputation` отвечает за хранение рейтинга пользователей и отзывов, предоставленных другими участниками системы.
  - Таблица `gaming_preferences` хранит жанры игр и платформы, предпочтительные для пользователей.
  - Для уведомлений и системы взаимодействия предусмотрены таблицы `application` и `post`.
- Оптимизация базы данных:
  - Для ускорения запросов были добавлены индексы на ключевые поля, такие как `user_id`, `game_id` и `email`.
  - Реализованы связи между таблицами с использованием внешних ключей для обеспечения целостности данных.

Итоговая схема представлена на рисунке 1, что отражает детализированную структуру данных.



Рисунок 1 - Схема БД

Для управления регистрацией и авторизацией пользователей был использован гем Devise [2]. Этот инструмент позволил быстро интегрировать следующие функции:

- Регистрация новых пользователей с подтверждением email.
- Авторизация с использованием зашифрованных паролей.
- Восстановление пароля через email.
- Ограничение количества попыток входа для предотвращения атак перебора.

Система поиска напарников была реализована с учётом необходимости фильтрации по нескольким параметрам:

1. Серверная логика:
  - Фильтры включали параметры, такие как предпочитаемые игры, уровень навыков, время доступности и репутация.
  - Для повышения эффективности поиска использовалась динамическая генерация SQL-запросов, которые обрабатывались на стороне сервера.
2. Репутационная система:
  - Была внедрена возможность оценивать напарников после завершения совместной игры.

- Рейтинги сохранялись в базе данных и влияли на видимость пользователей в результатах поиска.

Во время работы над поставленной задачей возникло несколько трудностей:

1. Сложности с проектированием базы данных:

- В процессе обсуждения с коллегой возникли разногласия по поводу структуры базы данных и необходимости определенных таблиц. Потребовалось несколько итераций и согласований, чтобы прийти к оптимальному решению.
- Из-за отсутствия должного опыта в проектировании БД было затрачено большое количество времени для изучения этой темы, что сильно повлияло на оставшееся время для дальнейшей разработки.

2. Интеграция с Devise:

- Интеграция гема Devise заняла больше времени, чем ожидалось, из-за необходимости кастомизации стандартного функционала, такого как подтверждение email и ограничения попыток входа.
- Возникли сложности с локализацией сообщений об ошибках, требовалось детальное изучение документации.

#### **4 Анализ выполненной работы**

Работа над проектом представляла собой сложный и многогранный процесс, который потребовал глубокого анализа задач, структурированного подхода и взаимодействия с коллегами. Одной из ключевых задач стало проектирование базы данных, которая должна была не только охватывать все необходимые аспекты, но и обеспечивать оптимальную производительность при работе с фильтрами и поисковыми запросами.

Мне удалось организовать планомерную работу благодаря заранее составленному плану, который включал четкие этапы разработки и тестирования. Регулярные встречи с коллегами помогли синхронизировать задачи и оперативно вносить изменения в случае необходимости. Однако мне не всегда удавалось строго следовать изначальному графику, так как на некоторых этапах задачи требовали больше времени из-за необходимости изучения большого количества информации.

Проект предоставил множество возможностей для профессионального роста, и в процессе работы я значительно расширил свои навыки:

1. Проектирование баз данных: Я углубил свои знания в области проектирования БД, включая определение связей между таблицами, выбор оптимальных типов данных и добавление индексов для ускорения запросов. Этот опыт помог мне лучше понять, как проектировать масштабируемые и производительные структуры данных.

2. Интеграция готовых решений: Работа с гемом Devise для реализации аутентификации пользователей научила меня эффективно использовать сторонние библиотеки. Я освоил настройку Devise для кастомизации функционала и добавления функций безопасности, таких как шифрование паролей и защита от CSRF-атак.

3. Разработка серверной логики: Создание серверной части системы поиска напарников с фильтрацией по нескольким параметрам стало ценным

опытом работы с SQL-запросами, их оптимизацией и построением гибких алгоритмов фильтрации.

4. Командная работа: Совместное проектирование базы данных и обсуждение технических решений помогли мне развить навыки коммуникации и аргументации своих идей. Также я научился адаптироваться к различным стилям работы и находить компромиссы.

5. Управление временем и задачами: Проект научил меня более эффективно расставлять приоритеты, а также оперативно реагировать на изменения в графике работы. Это позволило мне лучше организовать свою работу и справляться с неожиданными задачами.

6. Понимание полного цикла разработки: Выполнение проекта дало мне представление о том, как все этапы разработки (проектирование, разработка, тестирование) взаимосвязаны и как важно учитывать долгосрочные последствия решений, принимаемых на начальных этапах.

Этот проект стал важным этапом в моем профессиональном развитии. Он помог мне не только углубить технические знания, но и приобрести уверенность в своих силах, научиться справляться с трудностями и адаптироваться к изменяющимся условиям. Полученные навыки и опыт я планирую применять в будущем, как в командных, так и в индивидуальных проектах.



## 5 Взаимодействие в проекте

Взаимодействие с командой в рамках проекта было организовано на высоком уровне. Регулярные обсуждения задач и активная коммуникация через общий чат способствовали оперативному решению возникающих вопросов. В процессе работы над проектированием базы данных я тесно взаимодействовал с коллегой, который также отвечал за backend-разработку. Совместные обсуждения позволили разработать оптимальную структуру базы данных и заранее учесть возможные сложности.

Работа с фронтенд-разработчиками была менее активной, так как мои задачи в основном касались серверной стороны приложения. Тем не менее, важные моменты интеграции были согласованы заранее, что помогло избежать лишних проблем на финальном этапе разработки.

Взаимодействие с руководителем проекта происходило на регулярной основе. Руководитель был доступен для обсуждения текущих вопросов и всегда готов помочь в случае возникновения сложностей. На ранних этапах он предоставил четкие требования к функционалу и активно участвовал в согласовании архитектурных решений, что помогло сэкономить время в дальнейшем. Его профессиональный подход и готовность к диалогу создали комфортную рабочую атмосферу, что положительно сказалось на общей продуктивности команды.

Оценка работы руководителя проекта исключительно положительная. Он грамотно распределял задачи между членами команды, помогал определить приоритеты и предоставлял обратную связь, которая позволяла корректировать курс работы в нужное время. Руководитель не только координировал процессы, но и мотивировал команду заинтересованностью в конечном результате. Также на созвонах руководитель часто давал советы по разработке на Ruby on Rails и делился своим опытом в решении каких-то задач, что позволяло легче освоиться в этой среде.

Подводя итог, могу сказать, что взаимодействие с командой и руководителем проекта было продуктивным и способствовало успешной реализации задач.

## ЗАКЛЮЧЕНИЕ

Результаты работы над проектом в целом можно оценить положительно. Основная цель разработки прототипа веб-приложения для поиска напарников для видеоигр была достигнута. Приложение реализовало ключевые функции, такие как регистрация и авторизация пользователей, поиск напарников с использованием фильтров, управление профилем, а также система уведомлений. Все поставленные задачи, обозначенные в рамках проекта, были успешно выполнены.

Мой личный вклад в достижение целей проекта включал следующие ключевые аспекты:

- Проектирование базы данных: Я участвовал в создании схемы базы данных, определял структуры таблиц и связи между ними. Это позволило обеспечить удобство хранения и доступа к данным, а также поддерживать целостность и производительность приложения.
- Реализация системы авторизации: Я разработал функционал регистрации, входа в систему и восстановления пароля с использованием Devise, что обеспечило удобство и безопасность для пользователей.
- Серверная логика поиска напарников: Я создал логику работы фильтров, позволяющих пользователям искать напарников по разным параметрам. Это включало оптимизацию SQL-запросов для повышения скорости поиска.
- Внедрение элементов безопасности: Были реализованы шифрование паролей, защита от CSRF-атак и другие функции, повышающие защищенность пользовательских данных.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Rusrails: Ruby on Rails по-русски // rusrails.ru URL: <https://rusrails.ru/> (дата обращения: 05.01.2025).
2. GitHub - heartcombo/devise: Flexible authentication solution for Rails with Warden. // github.com URL: <https://github.com/heartcombo/devise> (дата обращения: 05.01.2025).
3. PostgreSQL: The world's most advanced open source database // postgresql.org URL: <https://www.postgresql.org> (дата обращения: 05.01.2025).

## ПРИЛОЖЕНИЕ

### Техническое задание

#### 1 Общие положения

1.1 Название проекта: «Разработка прототипа веб-приложения для поиска напарников для видеоигр».

1.2 Цель (назначение): создать работающий прототип веб-приложения, который позволит пользователям находить напарников для совместных игр, используя удобные фильтры и систему репутации.

1.3 Сроки выполнения:

– Начало: 01.11.2024

– Окончание: 20.12.2024

1.4 Команда проекта:

– Исполнитель проекта (руководитель проекта): Еникеев Ринат Алиевич

– Дизайн и UX: Пластинина Мария Вячеславовна

– Backend-разработчики: Эль Хеннави Александр, Янченко Денис Сергеевич

– Frontend-разработчики: Марченко Вадим Александрович, Аксянова Алина Рустамовна

1.5 Основные этапы задач:

– Сбор требований и анализ

– Проектирование интерфейса и дизайн

– Проектирование архитектуры и настройка среды разработки

– Разработка основных модулей приложения

– Тестирование и отладка

#### 2 Технические требования

##### 2.1 Функциональные требования

– Создание и управление профилем: возможность регистрации нового пользователя, редактирования личных данных, изменения

списка интересов и просмотра собственного статуса в системе.

- Система поиска и фильтрации: поиск напарников по играм, жанрам, навыкам, времени доступности, языку общения и другим параметрам; выдача результатов с учётом выбранных фильтров.

- Механизм отправки запросов на совместную игру: пользователи должны иметь возможность отправлять приглашения, принимать или отклонять запросы, а также видеть их статус (ожидание, принято, отклонено).

- Уведомления о запросах и приглашениях: система должна уведомлять пользователей в режиме реального времени (или при следующем входе) о поступивших заявках, сообщениях и изменениях в их профилях.

- Регистрация и авторизация пользователей: доступ к основному функционалу только после авторизации; поддержка базовых мер безопасности (хеширование паролей, защита от SQL-инъекций).

## 2.2 Нефункциональные требования

- Удобство использования (UI/UX): интерфейс должен быть простым, адаптивным и не перегруженным элементами; основные функции должны быть доступны без дополнительных сложных переходов.

- Легкость внесения обновлений: кодовая база должна поддерживать совместную разработку и интеграцию новых модулей без риска сломать существующую функциональность.

- Производительность: система должна корректно обрабатывать запросы пользователей и предоставлять результаты поиска в разумные сроки (при проектной нагрузке не менее 1000 пользователей одновременно).

- Используемый стек технологий:

Язык разработки бэкенда: Ruby (часть на Rails),

высоконагруженные сервисы — Rust по необходимости.

Язык разработки фронтенда: JavaScript (React).

СУБД: PostgreSQL (хранение профилей, заявок, статистики).

– Совместимость: корректная работа в современных браузерах (Chrome, Firefox, Safari, Edge).

### 3 Ожидаемый результат

3.1 MVP системы, удовлетворяющее описанным выше функциональным и нефункциональным требованиям, развернутое на сервере и доступное для тестирования конечными пользователями и членами команды.

3.2 Гибкая архитектура, позволяющая в дальнейшем безболезненно масштабировать проект, добавлять новые фильтры, модули геймификации (уровни, награды) и дополнительную аналитику (навыки игроков, эффективность команд).

3.3 Продуманный интерфейс, ориентированный на интуитивную навигацию и высокую скорость выполнения основных операций (создание профиля, фильтрация, поиск, отправка заявок).

4 Конечная цель: предоставить игрокам удобное средство для нахождения и приглашения напарников в различных сетевых видеоиграх, улучшая их совместный опыт и ускоряя организацию командных сессий.