

**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: Аксянова Алина Рустамовна, К3140

Санкт-Петербург 2024

## СОДЕРЖАНИЕ

<b>ВВЕДЕНИЕ .....</b>	<b>3</b>
<b>1 Суть проекта.....</b>	<b>5</b>
<b>2 Процесс работы над всем проектом .....</b>	<b>6</b>
<b>3 Моя роль в проекте .....</b>	<b>8</b>
3.1 Дизайн.....	8
3.2 Фронтенд-разработка .....	11
<b>ЗАКЛЮЧЕНИЕ .....</b>	<b>15</b>
<b>СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....</b>	<b>17</b>
<b>ПРИЛОЖЕНИЕ .....</b>	<b>18</b>

## ВВЕДЕНИЕ

Проект, над которым я с командой работали в рамках курсовой работы, является крайне актуальным на данный момент. После блокировки популярного сервиса “Discord”, которым пользовались миллионы геймеров для общения и поиска команды, возникла острая необходимость в альтернативных платформах. Геймерам важно иметь удобное место, где можно легко находить напарников для совместных игр, ведь командная игра включает в себя общение и новые знакомства. Кроме того, сегодня видеоигры стали огромной частью жизни многих людей. Многопользовательские игры требуют слаженной команды, но найти подходящих игроков бывает сложно, особенно если ты новичок или ищешь кого-то с определёнными навыками и стилем игры. Наш проект как раз направлен на создание прототипа сервиса, который поможет игрокам быстрее и проще находить подходящих людей. Система репутации в нашем веб-приложении будет помогать избегать недобросовестных пользователей, а удобные фильтры позволят выбирать напарников по конкретным критериям, таким как жанр игры, возраст и название игры. Наш проект будет полезен как опытным игрокам, которые ищут надёжных напарников для сложных заданий, так и новичкам, которым сложно влиться в игровое сообщество.

Объединяя вышесказанное, можно определить цель нашего проекта: Создание прототипа веб-приложения, которое позволит пользователям находить напарников для совместных игр, используя удобные фильтры и систему репутации.

Для успешной реализации проекта были сформулированы основные задачи, которые нужно было выполнить нашей команде:

- Исследовать аналоги существующих приложений для поиска напарников по играм, выделить их ключевые функции и интерфейсные решения

- Разработать макет пользовательского интерфейса, включающий профили пользователей, поиск и фильтрацию напарников
- Настроить базу данных для хранения профилей пользователей, их игровых предпочтений и истории взаимодействий
- Разработать интерфейс главной страницы
- Реализовать регистрацию и авторизацию пользователей с базовыми функциями безопасности
- Разработать интерфейс страницы профиля
- Реализовать функционал профиля с возможностью указывать игровые интересы, предпочтения и статусы
- Создать систему поиска напарников с фильтрацией по играм, уровню, времени доступности и другим параметрам
- Реализовать функционал уведомлений о новых запросах, сообщениях и приглашениях в игру.
- Разработать интерфейс страницы с заявками на игру пользователя

# РАБОТА НАД ПРОЕКТОМ

## 1 Суть проекта

Наш проект представляет собой прототип веб-приложения, которое будет помогать геймерам находить напарников для совместной игры. В современных условиях, когда многие популярные платформы недоступны или не предлагают удобного функционала для поиска команды, мы создали решение, ориентированное на потребности игроков.

Ключевые функции нашего приложения:

- Система поиска напарников. Пользователи могут находить игроков, используя фильтры по жанру игр, времени онлайн и возрасту.
- Система репутации. Это позволяет игрокам оценивать друг друга, что способствует созданию безопасной и комфортной среды.
- Профили пользователей. Участники сообщества могут создавать собственные страницы с указанием игровых интересов, предпочтений и статусов.
- Добавление в друзья. Пользователи могут оставлять заявки на игру с индивидуальными требованиями, которые другие пользователи могут откликаться, а в последствии добавлять друг друга в список друзей для быстрого доступа к профилям и лёгкого взаимодействия в будущем.

## **2 Процесс работы над всем проектом**

В соответствии со списком поставленных задач, работа делилась на определенные этапы. Для каждой задачи был определен свой ответственный, однако, так как мы работали в команде, мы всегда помогали друг другу в сложных или спорных моментах. Такой подход позволял эффективно распределять нагрузку и обеспечивать высокое качество выполнения каждой задачи.

В рамках проекта команда была разделена на три ключевые роли (стек):

- Бэкенд-разработчики (2 человека): создание серверной части и работа с базой данных.
- Фронтенд-разработчики (2 человека): разработка интерфейса пользователя.
- Дизайнер (1 человек): проработка дизайна и удобства использования.

Для поддержания эффективной коммуникации мы создали общий чат, где обсуждали ключевые вопросы, распределяли задачи и обменивались идеями. В чате также были выделены отдельные каналы для фронтенд и бэкенд разработки, что позволило каждому направлению делиться успехами в процессе разработки и совместно решать появившиеся вопросы. Руководитель проекта также озвучивал нам различные задачи в чате и предлагал удобные и полезные ресурсы для изучения неизвестных нам аспектов разработки.

Кроме того, мы регулярно проводили созвоны для обсуждения прогресса и решения возникающих вопросов. Такие встречи позволяли

координировать действия, делиться результатами работы и предлагать улучшения.

Для совместной работы с кодом был создан репозиторий на GitHub. Это обеспечило удобное хранение и управление проектом, а также позволило каждому участнику команды вносить изменения, тестировать их и отслеживать прогресс. Чтобы развернуть проект с GitHub локально на своем устройстве, каждому из участников нужно было установить Ruby (язык программирования) и соответствующий фреймворк Rails, так как эти ресурсы использовались непосредственно в разработке и в репозитории.

Подход к организации работы, описанный выше, помог нам эффективно справляться с задачами, минимизировать задержки и достичь согласованности во всех аспектах разработки — от дизайна до функционала.

### **3 Моя роль в проекте**

Моей основной ролью в проекте являлась фронтенд-разработка, но, в связи с тем, что дизайнер у нас был лишь один, сроки были ограниченны и работа была крупной, руководитель дал мне задачу также поработать над дизайн-проектом.

#### **3.1 Дизайн**

Для создания дизайн-шаблона мы использовали приложение Figma. Оно является наиболее известным среди веб-разработчиков, а также удобным и простым в использовании. Несмотря на понятность интерфейса Figma, мне пришлось изучить различную информацию в интернете, чтобы корректно спроектировать шаблон. Для удобной совместной разработки я и дизайнер создали единый рабочий проект в приложении. На протяжении всей совместной работы мы коммуницировали между собой, предлагали варианты для улучшения и помогали друг другу.

Цветовую гамму и общий стиль сайта подобрала дизайнер проекта, она оформила первые 2 страницы сайта – страницу профиля и поиска с помощью фильтрации (см. рисунок 1).



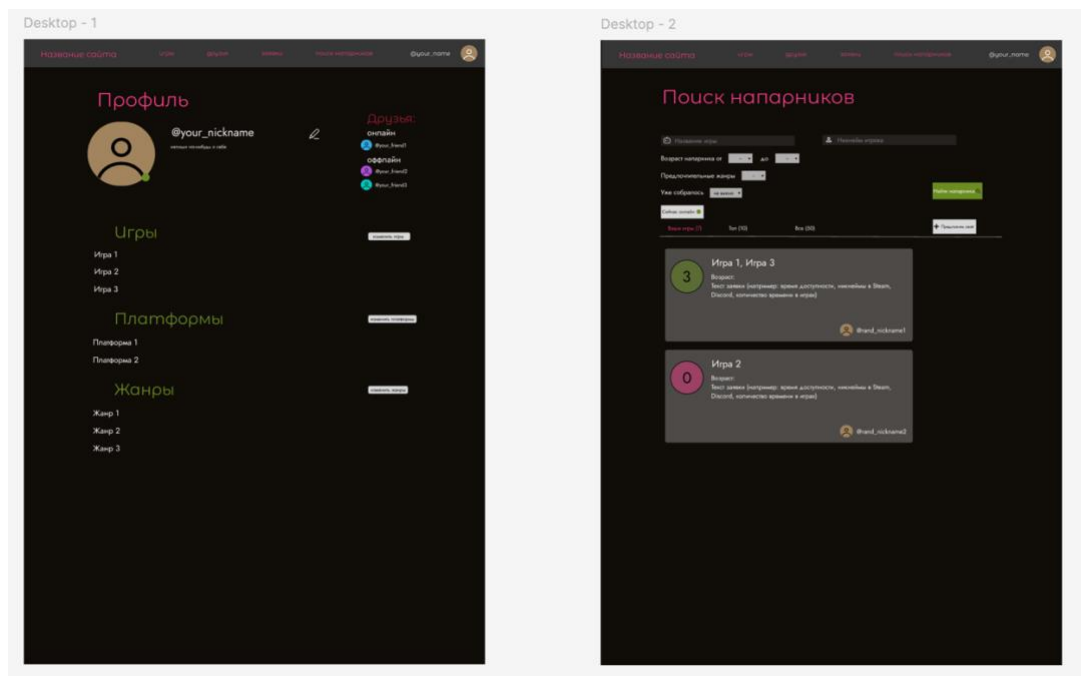


Рисунок 1 - Макеты страниц профиля и поиска

Далее я, используя подобранные дизайнером цвета и общую стилистику, спроектировала еще 3 макета страниц (главную страницу, страницу заявок и страницу друзей), а также шапку сайта (см. рисунок 2 и 3).

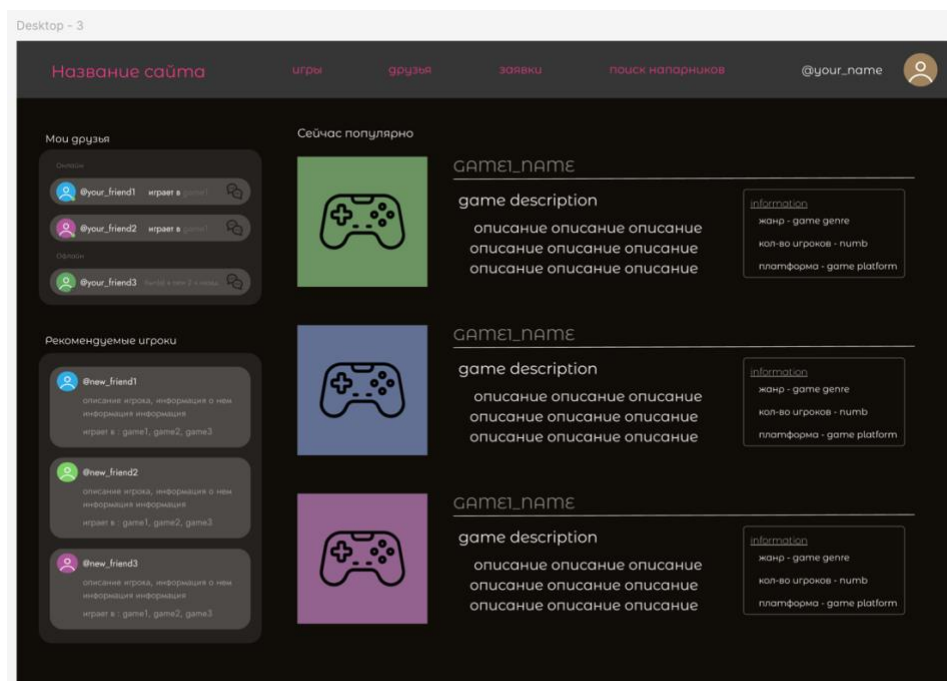


Рисунок 2 – Шаблон главной страницы

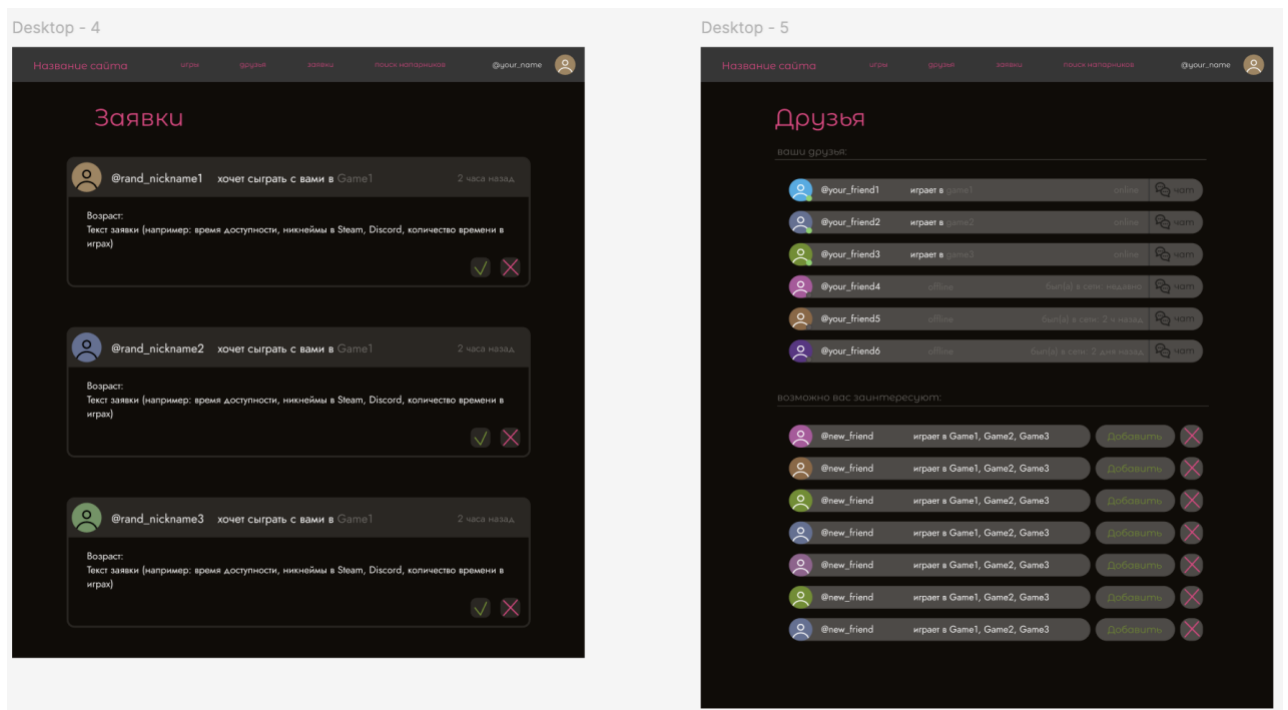


Рисунок 3 – Макет страницы заявок и друзей

На этом работа над дизайн-шаблоном была завершена, но в последствии, при верстке сайта были внесены незначительные изменения. Создание дизайна не вызвало у меня никаких особых трудностей, процесс был интересным и не особо сложным.

## 3.2 Фронтенд-разработка

Фронтенд-разработка включала в себя создание пользовательского интерфейса, а также реализацию функционала, который обеспечивал удобство и взаимодействие пользователей с приложением. По ходу работы я использовала дизайн-шаблоны созданные ранее в приложении Figma. Основными инструментами, которые я использовала для разработки, стали HTML, CSS, JavaScript и React. HTML и CSS использовались для разметки страниц и стилизации элементов, JavaScript отвечал за интерактивность, React нужен был для того, чтобы эффективно управлять состоянием и динамичным изменением интерфейса.

Одной из основных задач при разработке было создание адаптивного дизайна, который бы позволил пользователям видеть сайт одинаково и правильно как на мобильном устройстве, так и на ПК. Не все дизайн-шаблоны подходили для корректной адаптации, некоторые элементы требовали дополнительных изменений. Для корректного отображения и должного внешнего вида приложения на экранах разного размера мне пришлось работать с медиазапросами в CSS. Это был достаточно трудоёмкий процесс, занявший достаточно времени.

В процессе работы над проектом я также активно участвовала в обсуждениях с командой, делая предложения по улучшению интерфейса и функционала. Часто обсуждалось то, как улучшить взаимодействие между фронтендом и бэкендом, а также как лучше реализовать различные элементы дизайна, чтобы это было практично и удобно для пользователей. Мы регулярно проводили созвоны и обсуждения, где решали возникающие проблемы.

В результате моей работы, совместно со вторым фронтенд-разработчиком был создан полноценный сайт, который был захосчен и

выложен в сеть (т.е. он стал доступен не только локально, но и для других пользователей). На рисунке 4 можно увидеть, как он выглядит в финальной версии. А на рисунке 5 виден результат адаптивной верстки, интерфейс корректно отображается на маленьком экране смартфона.

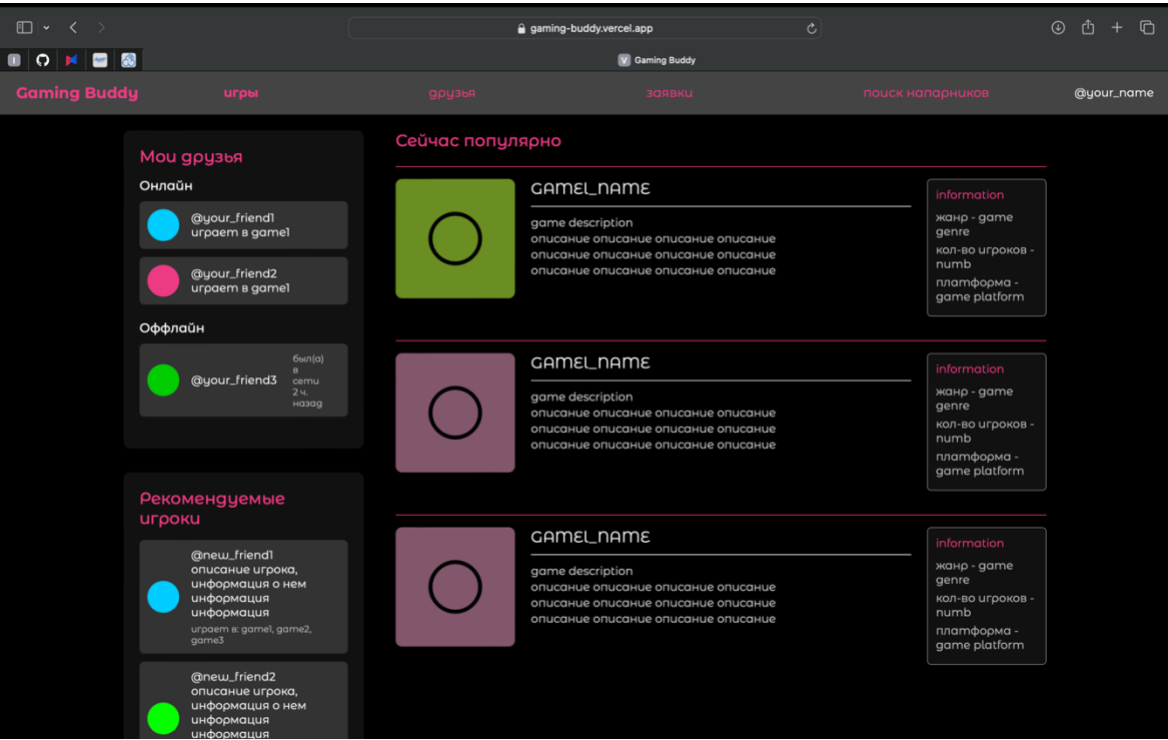


Рисунок 4 – Свёрстанный итоговый вид веб-приложения



Рисунок 5 – Адаптация интерфейса

### 3.3 Анализ моей работы

Работая над проектом, я получила ценный и полезный опыт, который позволил расширить мои знания и приобрести новые навыки в области веб-разработки, дизайна и командной работы. Проект позволил мне погрузиться в создание полноценного веб-приложения от идеи до реализации, благодаря чему я теперь имею полное представление о веб-разработке. В процессе работы я столкнулась с неизвестными мне инструментами разработки, например React. Изучая его, я узнала принципы построения динамических и интерактивных интерфейсов. Это включало как использование стандартных технологий, таких как HTML, CSS и JavaScript, так и работу с более сложными вещами, такими как

взаимодействие с API. Кроме того, проект дал мне возможность освоить основы фреймворка Rails, который использовался для серверной части. Это важный опыт, ведь в современной разработке многие компании используют именно эти ресурсы разработки и требуют от кандидатов осведомленности в них. Хочется отметить, что наш руководитель оказывал поддержку, объясняя непонятные вещи и предлагая решения для задач, с которыми у нас возникали проблемы. Он предлагал нам полезные ресурсы для изучения нужных нам тем.

Основной трудностью для меня стала работа с бэкендом. Поскольку у меня не было достаточного опыта в этой области, некоторые задачи оказались для меня сложными. Второй фронтенд взял на себя основную часть работы с серверной частью, так что я смогла сосредоточиться на фронтенд-разработке и в то же время получить базовое понимание принципов работы серверной части сайта и взаимодействия с базой данных.

К сожалению, времени на проект было ограничено, и мы не смогли реализовать все наши задумки. Некоторые элементы интерфейса и функционала остались не до конца доработанными, а часть идей мы не успели воплотить. Хотелось бы добавить больше функционала для нашего веб-приложения.

## ЗАКЛЮЧЕНИЕ

Подводя итог, хотелось бы оценить всю сделанную нами работу. Главная цель нашего проекта была достигнута, наша команда вместе с нашим руководителем разработали прототип веб-приложения для поиска напарников для совместных игр. Нами были выполнены практически все ранее поставленные цели: мы разработали дизайн-шаблоны веб-приложения, которые включали в себя страницы профиля, заявок, друзей, игр и страницу поиска по фильтру. Был реализован адаптивный дизайн, который позволит пользователям комфортно использовать сайт на различных устройствах. Также было налажено взаимодействие между клиентской и серверными частями через работу с базами данных.

Однако нам не удалось воплотить все задуманное из-за ограниченных сроков работы. Например, хотелось бы также добавить систему уведомлений, добавить больше возможностей для кастомизации профилей и расширить функционал приложения в целом. Тем не менее, эти недоработки не являются существенными для работы сайта, а идеи для улучшения можно реализовать в будущем.

Мой личный вклад в проект был связан с фронтенд-разработкой и участием в создании дизайна. Я разработала дизайн-проект 3 страниц нашего сайта – главной страницы и страниц друзей и заявок. Также я занималась версткой интерфейса, работала над адаптивностью сайта и взаимодействием между элементами дизайна и функционалом. Используя дизайн-шаблоны, я создала основные страницы приложения и реализовала их в реальном прототипе.

Работа в команде оказалась важным элементом успешной реализации проекта. Постоянное общение, совместное обсуждение идей и помощь друг другу помогли нам организовать процесс разработки

максимально эффективно. Созвоны, работа в чате и использование совместного репозитория на GitHub позволили нам координировать свои действия и оставаться в курсе прогресса.

Подводя итоги, я могу сказать, что проект стал для меня важным этапом в обучении. Я не только приобрела ценный практический опыт в области веб-разработки, но и научилась работать в команде, решать сложные задачи и использовать современные инструменты для создания веб-приложений. Полученные навыки и знания пригодятся мне в будущем и помогут мне сейчас развиваться в сфере IT.



## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Rails Guide – Action View Overview: [Электронный ресурс] - URL: [https://guides.rubyonrails.org/action\\_view\\_overview.html](https://guides.rubyonrails.org/action_view_overview.html).
2. Go Rails – Install Ruby on Rails on Windows: [Электронный ресурс] - URL: <https://gorails.com/setup/windows/10>
3. Tilda Education – Гид по Фигме: [Электронный ресурс] - URL: <https://tilda.education/articles-figma>
4. Homebrew : [Электронный ресурс] - URL: <https://brew.sh/ru/>
5. Документация по Ruby: [Электронный ресурс] - URL: <https://www.ruby-lang.org/ru/about/>

# 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.
- Лёгкость 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 бекенда: 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, развёрнутое 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.