**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика, системы управления и искусственный интеллект»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе №5

«Веб приложение Мессенджер с использованием клиент-серверной архитектуры.»

|  |  |  |
| --- | --- | --- |
| Выполнила: |  | Проверил: |
| Команда Elephango Team группы  ИУ5-32Б: |  | преподаватель каф. ИУ5 |
| Астахов И.А.  Копылов Н.Н.  Фролов И.О. |  | Гапанюк Ю.Е. |
| Подпись и дата: |  | Подпись и дата: |

Москва, 2024

### **Причина выбора темы проекта**

Перед командами была поставлена цель разработать веб приложение, использующее веб-сокеты для обмена сообщениями.

Одним из первых вариантов у нашей команды стало приложение Мессенджер *Elephango*, что являлось бы аналогом для Telegram.

***Цель работы –*** осуществить разработку программного продукта «***Elephango***» – клиент-серверного приложения, предназначенного для обмена сообщениями пользователями.

***Актуальность выполнения проекта.*** Социальные сети всё больше замещают реальное общение в нашем мире. По этой причине мы решили предложить свою реализацию быстрого Мессенджера, удобного в использовании для пользователей.

***Целевая аудитория разрабатываемого проекта:*** молодое поколение

***Анализ имеющихся программных продуктов*** показал, что Мессенджеров, обладающих базовых функционалом бесчисленное множество, но ни одно из них не использует тот стек разработки, что подразумевается нашим приложением и был описан в пункте про актуальность.

### **Методы и средства разработки**

Разработка проекта осуществлена в соответствии с выполнением следующих **этапов**:

1. Анализ потенциальных возможностей JavaScript-фреймворков и Rust-библиотек для реализации быстрого сервера;
2. Составление MVP проекта;
3. Первичное проектирование архитектуры приложения;
4. Разработка дизайна графического интерфейса;
5. Разработка приложения (серверной части и фронтенд части);
6. Тестирование и отладка с последующим исправлением возникающих ошибок;

Разработка ведётся с использованием системы контроля версий **git**, репозитории проекта расположены на **github:**

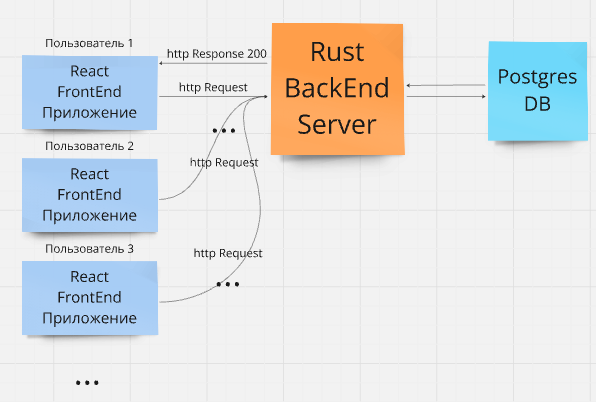
<https://github.com/IfroloffI/ElephangoMessenger>

**Используемые технологии**

##### ***Серверная часть***

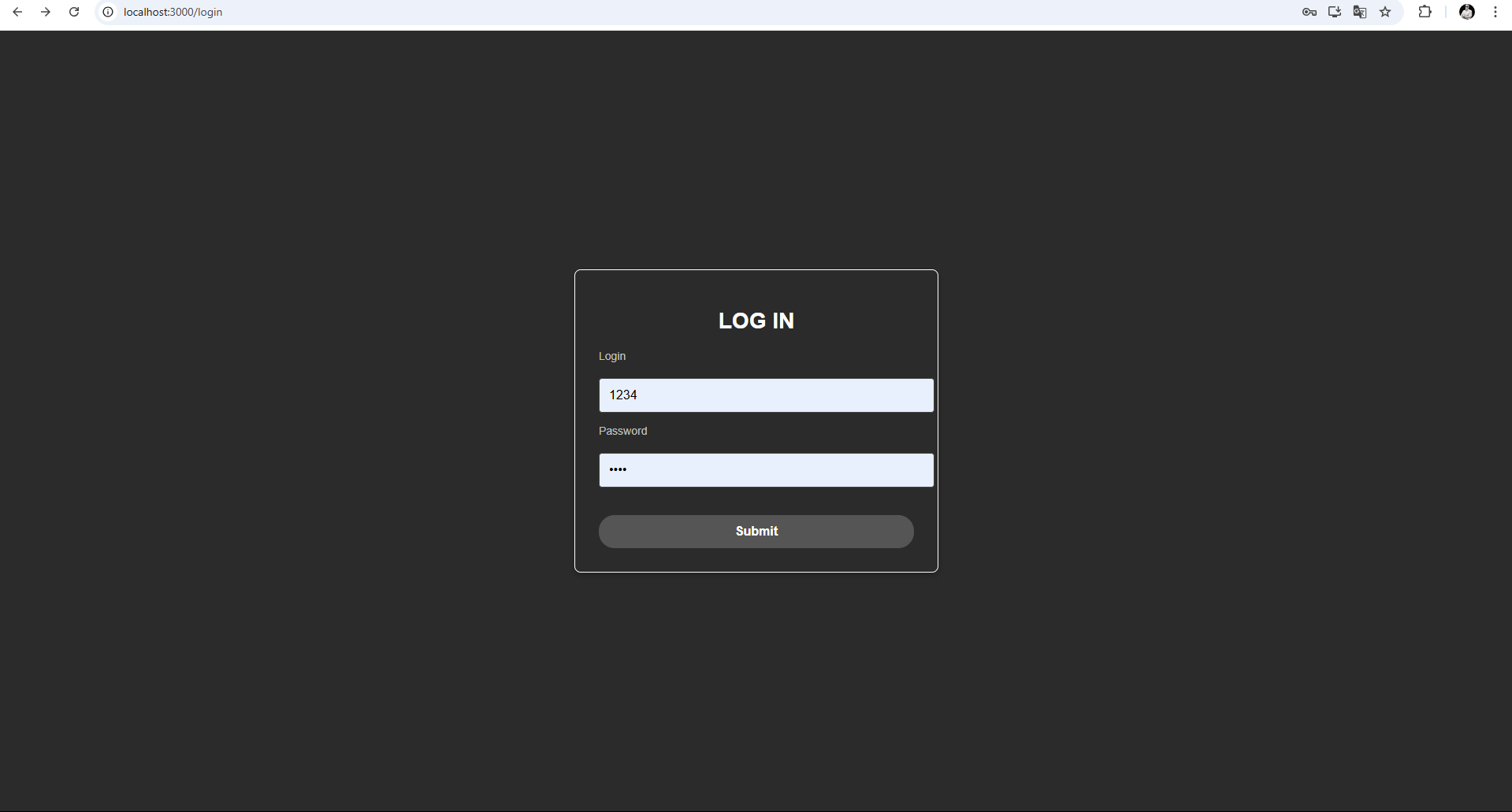
В качестве языков для написания серверной (backend) части проекта выбраны язык программирования **Rust**.

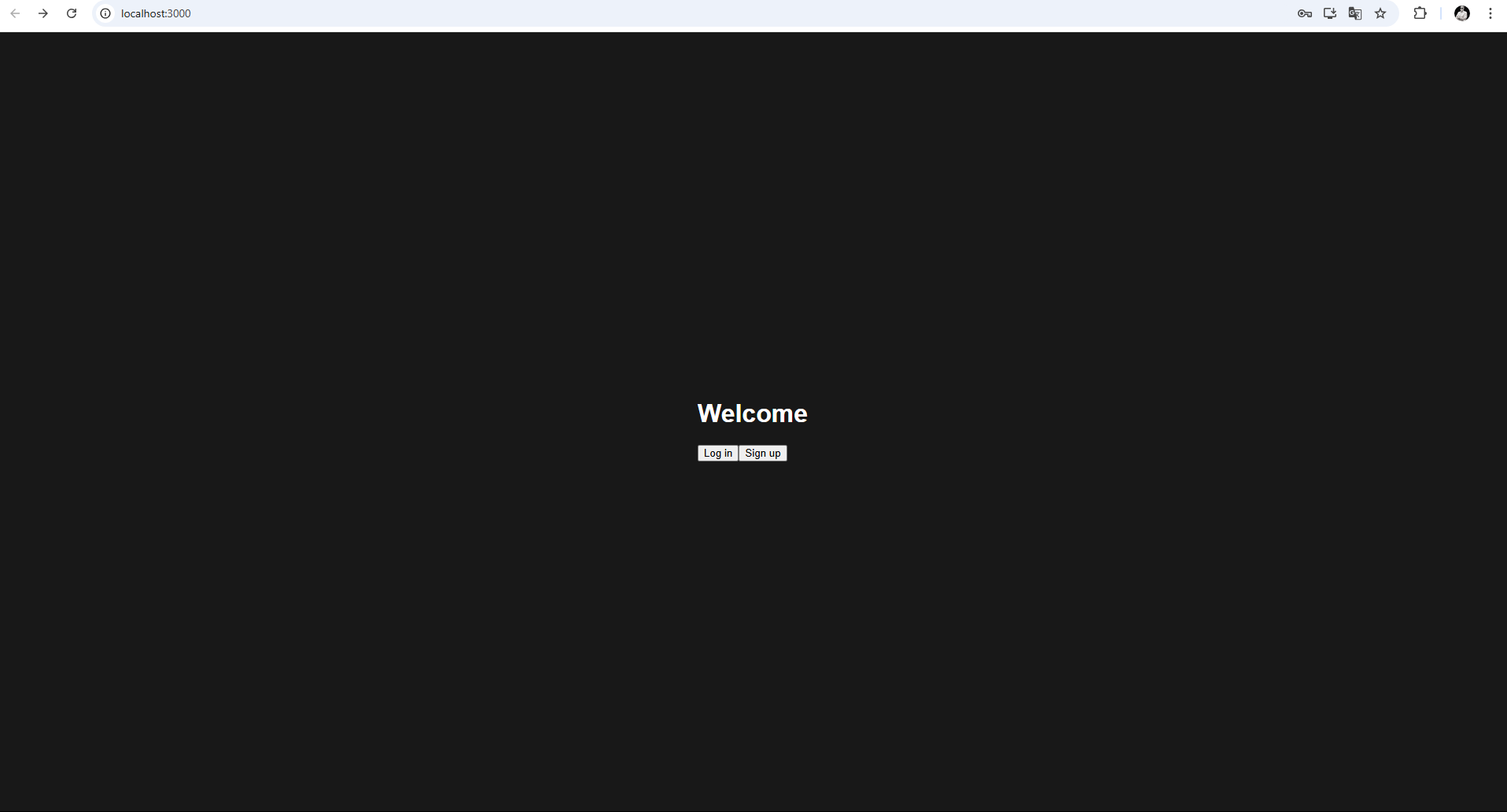
Было выбрано исполнение монолитного сервера, а не микросервисов в угоду скорости работы, производительности и в убыток масштабируемости для достижения поставленных целей проекта.

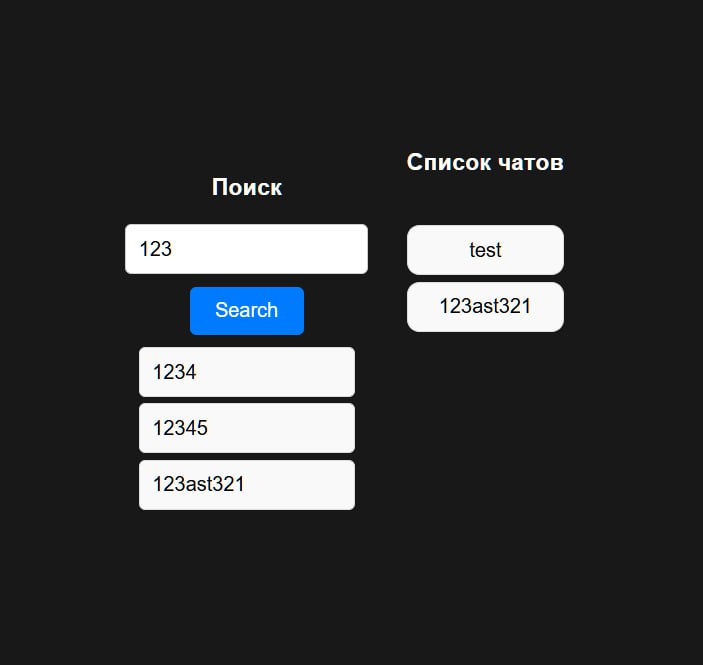
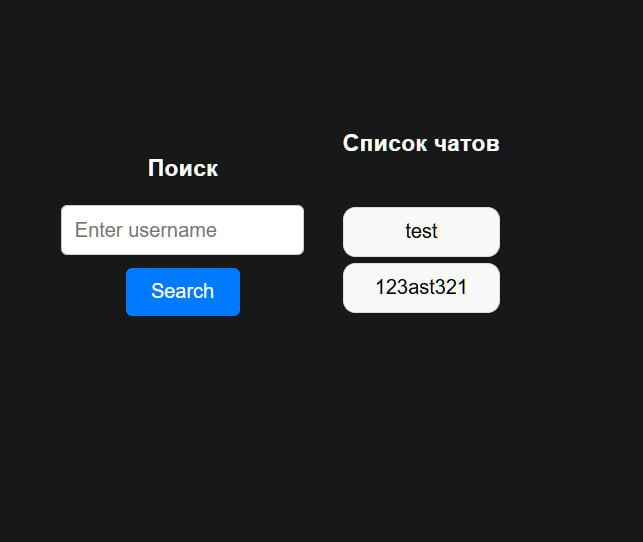
****

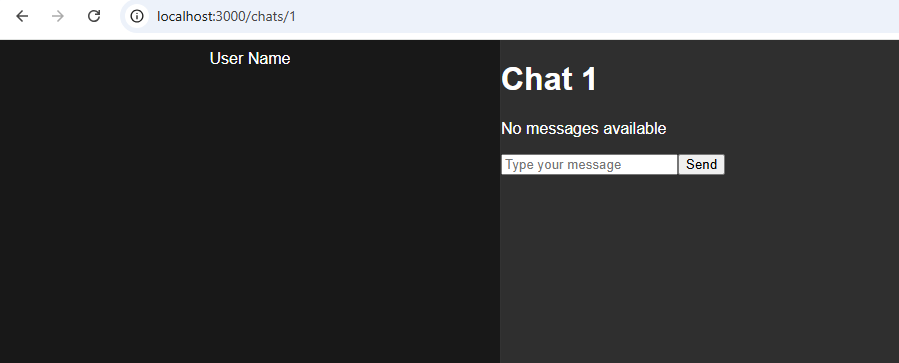
**Frontend**

Для создания веб интерфейса для всего функционала приложения использовался React в связи с его простотой, скоростью развертки и универсальности для разных платформ. Впоследствии стоит перевести проект на полноценное приложение, например на React Native или Electron для удобного развёртывания на Android и IOS.









### **Описание разработанного решения**

Разработанное решение состоит из двух программных продуктов: backend-сервера и браузерного клиента (WEB).

Разработанный сервис предоставляет следующую ***функциональность:***

* Отправка текстовых сообщений между пользователями
* Создание и Использование аккаунта для самоидентификации в Мессенджере
* Создание и удаление чатов между пользователями
* Умный поиск пользователей

Для запуска сервиса ***необходимо:***

1. Скачать исходный код из репозиториев, ссылки на которые приведены выше.
2. Запустить локально серверную часть в соответствии с инструкциями.
3. Запустить локальный frontend.

**Перспективы развития**

Предполагаемое ***расширение функциональности сервиса:***

* Добавление авторизации через Google или другие соц.сети ;
* Добавление возможности отправки стикеров и файлов
* Добавление возможности отслеживать онлайн пользователей.

***При дальнейшем развитии проекта планируется:***

* Улучшение безопасности сервиса;
* Переработка дизайна клиента;
* Развертка backend на сервере;
* Создание мобильного клиента (iOS, Android).