

МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГУ»)

Факультет Компьютерных наук

Кафедра программирования и информационных технологий

Курсовой проект  
Мессенджер с подбором собеседников по интересам «TextMe»

Руководитель \_\_\_\_\_ В. С. Тарасов, ст. преподаватель \_\_.\_\_.2021

Обучающийся \_\_\_\_\_ Е. Д. Проскуряков, 3 курс, д/о

Обучающийся \_\_\_\_\_ Д. С. Каширин, 3 курс, д/о

Обучающийся \_\_\_\_\_ А. Д. Кузнецов, 3 курс, д/о

Воронеж 2021

## Содержание

Содержание .....	2
Введение .....	4
1 Постановка задачи .....	5
2 Анализ предметной области .....	5
2.1 Обзор аналогов .....	5
2.2 Термины и сокращения .....	6
2.3 Анализ задачи .....	7
2.3.1 Варианты использования приложения .....	7
2.3.2 Взаимодействие компонентов системы .....	7
2.3.3 Варианты состояния системы .....	10
2.3.4 Варианты действий в системе .....	10
2.3.5 Развертывание приложения .....	11
2.3.6 Диаграмма классов .....	12
2.3.7 Диаграмма объектов .....	14
2.3.8 IDEF0 .....	15
2.4 Воронка конверсии .....	16
2.5 Яндекс.Метрика .....	17
2.5.1 Цель «Вход в приложение» .....	17
2.5.2 Цель «Начать беседу» .....	18
2.5.3 Цель «Список чатов» .....	18
2.5.4 Анализ пользователей .....	18
3 Реализация приложения .....	20
3.1 Средства реализации .....	20

3.2 Реализация Backend .....	20
3.2.1 Пользователь и регистрация .....	20
3.2.2 Заполнение карточки .....	21
3.2.3 Беседы.....	23
3.2.4 Подбор собеседников.....	24
3.2.5 Мессенджер .....	26
3.2.6 Администрирование .....	28
3.2.7 Цикл блокировки пользователя .....	29
3.3 Интерфейс пользователя .....	30
3.3.1 Эмблема приложения.....	30
3.3.2 Стартовые страницы .....	31
3.3.3 Главная страница.....	35
3.3.4 Чат.....	38
3.3.5 Мессенджер. Список чатов .....	41
3.3.6 Профиль пользователя.....	42
3.3.7 Администрация.....	48
3.3.8 Отображение на мобильном устройстве.....	53
4 Тестирование .....	54
4.1 Модульное тестирование .....	54
4.2 Ручное функциональное тестирование .....	54
4.3 Юзабилити-тестирование .....	56
Заключение .....	57
Список литературы .....	58

## **Введение**

У любого человека существует базовая потребность в общении, и каждый волен удовлетворять ее самым подходящим для себя образом. Безусловно, самый распространенный способ общения в 21 веке - интернет. В мире программного обеспечения существует немало решений подобного рода задачи, и люди активно ими пользуются, зачастую - сразу несколькими приложениями. Каждый программный продукт с течением времени увеличивает свой функционал, но иногда бывает так, что приложением обладает таким количеством возможностей, что они будут только отвлекать и нагружать пользователя, являясь для него лишними и ненужными. Задачей данного курсового проекта является разработка не нагруженного веб-приложения, предоставляющего пользователю функциональность, которая поможет найти интересного собеседника в сети интернет, основываясь на собственных увлечениях и предпочтениях.

## **1 Постановка задачи**

Главная задача разрабатываемого приложения - обеспечение поиска интересного собеседника и последующее общение с ним. Поиск должен осуществляться с помощью заполнения собственного профиля. Помимо общих полей должен быть указан специальный список тегов, отражающий интересы пользователя. Таким образом, пользователь составляет перечень интересов, по которым хочет завести диалог, а приложение выдает ему список самых подходящих кандидатов.

Выбрав понравившегося, пользователь может перейти в личный чат и начать диалог с базовой для мессенджера функциональностью - создание, отправка и редактирование собственных сообщений.

Второстепенной задачей является возможность модерирования нежелательного контента во всем приложении: каждый пользователь должен иметь возможность подать жалобу на любой профиль с указанием краткого описания причины. Модератор приложения, в свою очередь, должен иметь возможность просматривать жалобы, отклонять или принимать их, тем самым блокируя аккаунт пользователя, на которого подали жалобу.

## **2 Анализ предметной области**

### **2.1 Обзор аналогов**

На текущий момент на рынке представлено огромное количество программных продуктов, которые предоставляют схожие возможности. На российском рынке можно выделить следующие из них:

- «ВКонтакте»
- «Telegram»
- «Badoo»
- «Limer»

Сайт «ВКонтакте» предоставляет огромную функциональность для различных потребностей. Но для знакомств и поиска единомышленников данная социальная сеть не представила удобного сервиса. Пользователям

сайта приходится вступать в отдельные сообщества, подавать анкеты в них и ждать одобрения, что не для каждого человека может быть удобно. У «ВКонтакте» с недавних пор имеется микро-сервис «Lovina», который, в свою очередь, является аналогом «Badoo». Как первое, так и второе приложение ставят своей основной целью поиск не столько собеседника, сколько настоящего знакомого для личной встречи. Приложение, которое будет разработано в ходе данного курсового проекта, решит эти проблемы – оно будет удобным и быстрым для поиска тех людей, с которыми пользователю «здесь и сейчас» хочется обсудить интересующую его тему.

Мессенджер «Telegram», также включенный в список, предоставляет пользователю возможность мгновенной пересылки текстовых сообщений, как и разрабатываемый курсовой проект. Но, в отличие от последнего, целью «Telegram» является общение между уже знакомыми людьми, никакого поиска в этом приложении нет.

Сайт «Limega» предоставляет схожую функциональность, но обязует пользователя указывать реальное имя, пол, город проживания и дату рождения. Разрабатываемый в ходе данного курсового проекта мессенджер рассчитан на то, что пользователю не должно быть важно какого возраста, пола или какой расы собеседник, главное то, чем он интересуется и как об этом мыслит.

## **2.2 Термины и сокращения**

- Мессенджер, проект – разрабатываемое веб-приложение.
- Пользователь – авторизованный в системе человек, пользующийся функциональностью веб-приложения;
- Администратор – человек, имеющий доступ к расширенному функциональности веб-сервиса;
- Модератор – пользователь, имеющий более узкие права, чем администратор, но более широкие права, чем обычные пользователи.

- Карточка – структурный элемент страницы, на котором отображается краткая информация о Пользователе и набор выбранных им тегов;
- Тег – тематика, которой интересуется Пользователь;

## 2.3 Анализ задачи

### 2.3.1 Варианты использования приложения

На рисунке 1 продемонстрирована диаграмма прецедентов, которая показывает какие основные сценарии использования приложения доступны пользователю. У пользователя есть четыре основных сценария, каждый из которых расширяется дополнительными возможностями или вариантами развития событий.

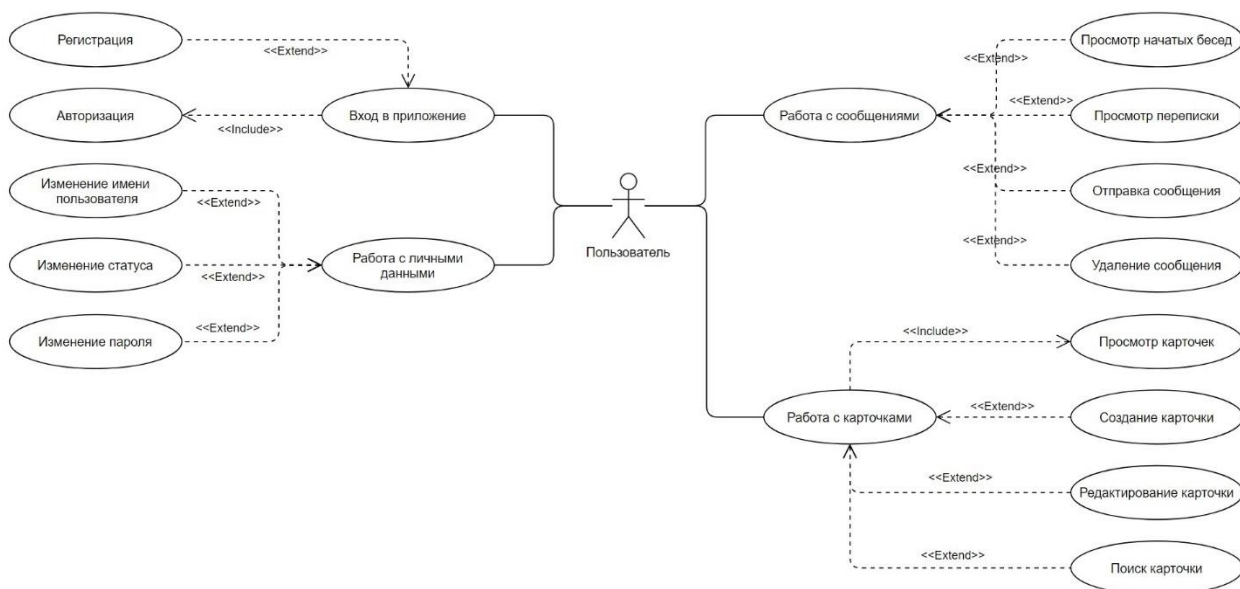


Рисунок 1 - Диаграмма прецедентов

### 2.3.2 Взаимодействие компонентов системы

На рисунке 2 показана диаграмма последовательности для взаимодействия пользователя с системой для следующих прецедентов:

- Авторизация;
- Просмотр списка диалогов;
- Редактирование личных данных;

Диаграммы взаимодействия для этих действий представлены на рисунках 3-5.

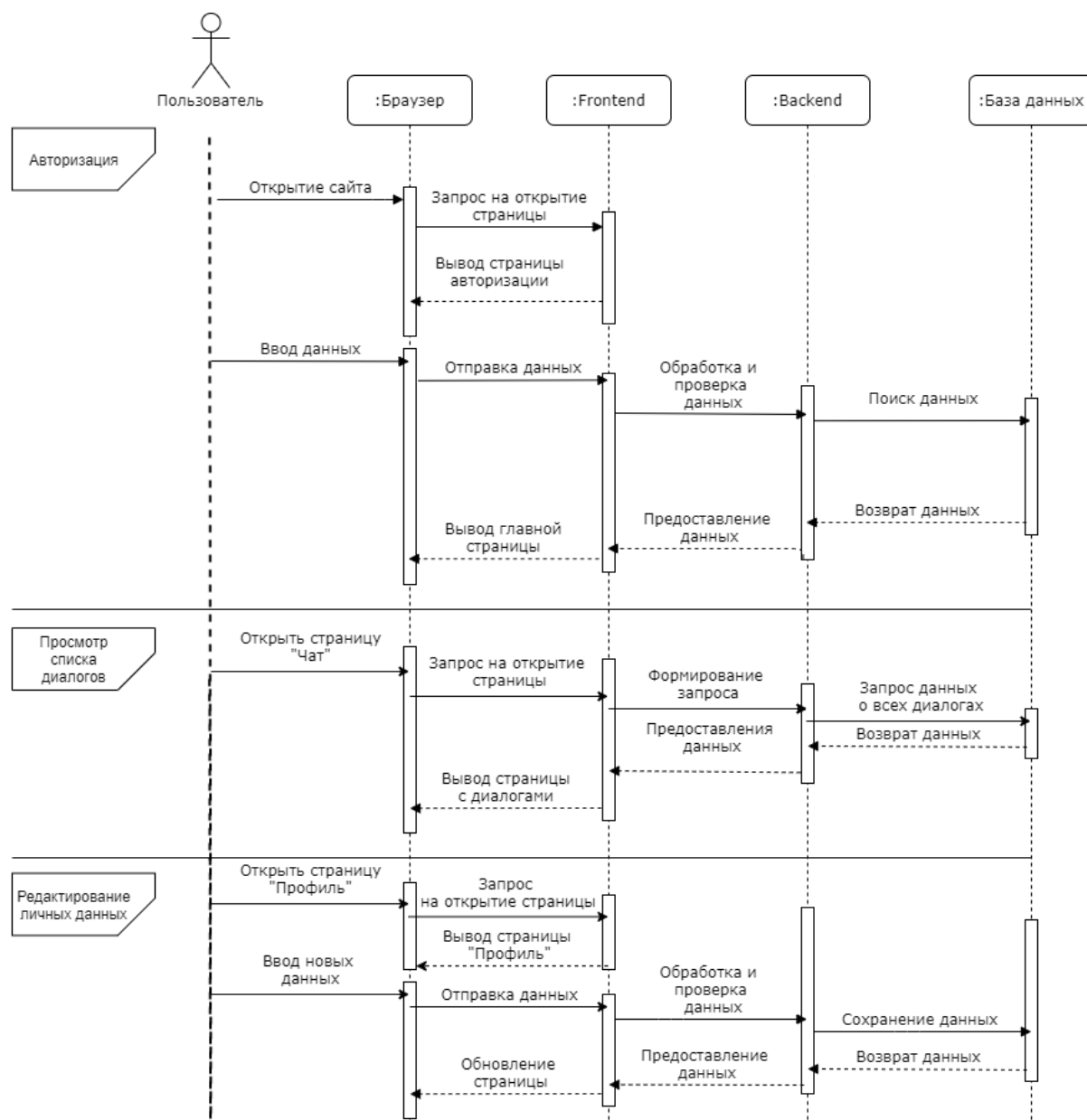


Рисунок 2 - Диаграмма последовательностей для пользователя



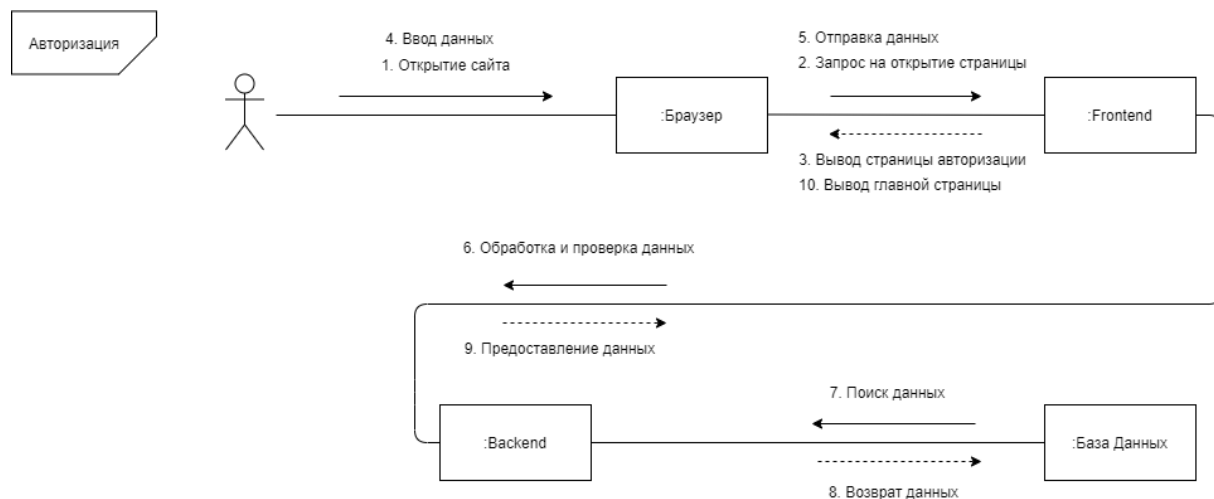


Рисунок 3 - Диаграмма взаимодействия для прецедента «Авторизация»



Рисунок 4 - Диаграмма взаимодействия для прецедента «Просмотр списка диалогов»

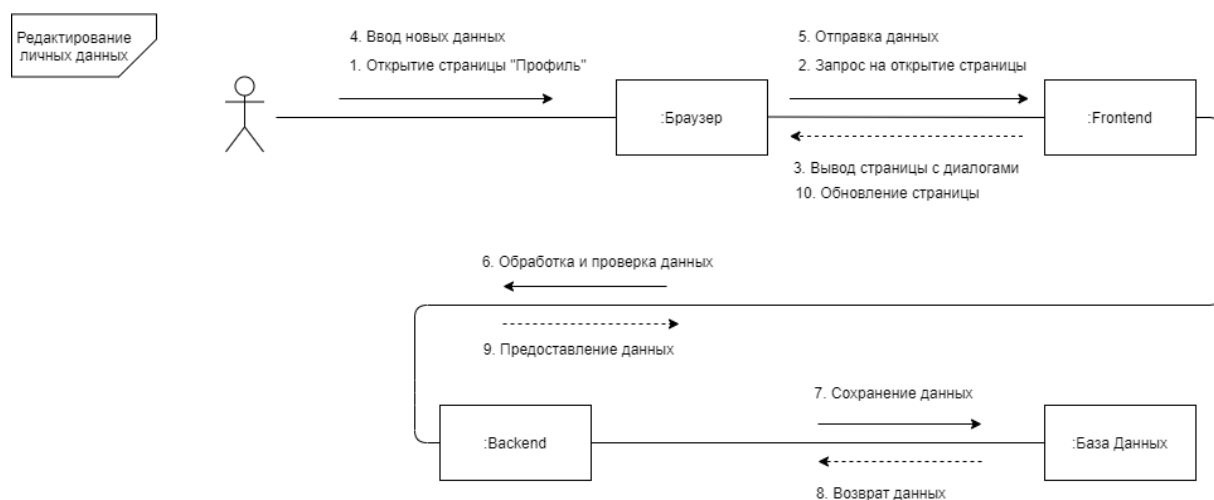


Рисунок 5 - Диаграмма взаимодействия для прецедента «Редактирование личных данных»

### 2.3.3 Варианты состояния системы

Диаграмма состояний, изображенная на рисунке 6, отображает возможные состояния системы. Для использования приложения, при запуске требуется авторизоваться или зарегистрироваться, после чего пользователь попадает на главную страницу.

После авторизации в системе пользователь может перейти на страницу профиля, списка начатых диалогов и поиска собеседников. На странице профиля пользователь может редактировать личные данные. На странице списка начатых диалогов, просмотреть список всех пользователей, с которыми был начат диалог. Если пользователь хочет отметить другого пользователя, чтобы позже начать с ним диалог, он может добавить его в избранное, нажав соответствующий пункт в контекстном меню.

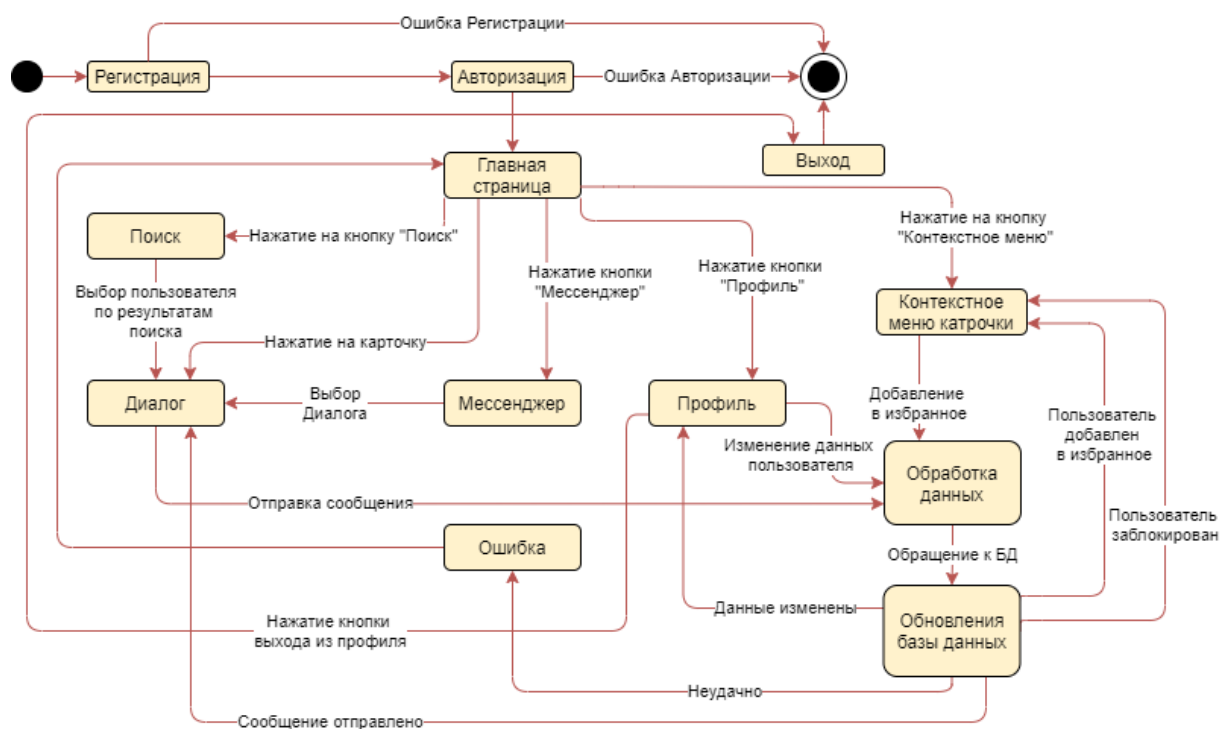


Рисунок 6 - Диаграмма состояний

### 2.3.4 Варианты действий в системе

Диаграмма активности, изображенная на рисунке 7, отражает возможные действия пользователя в системе.

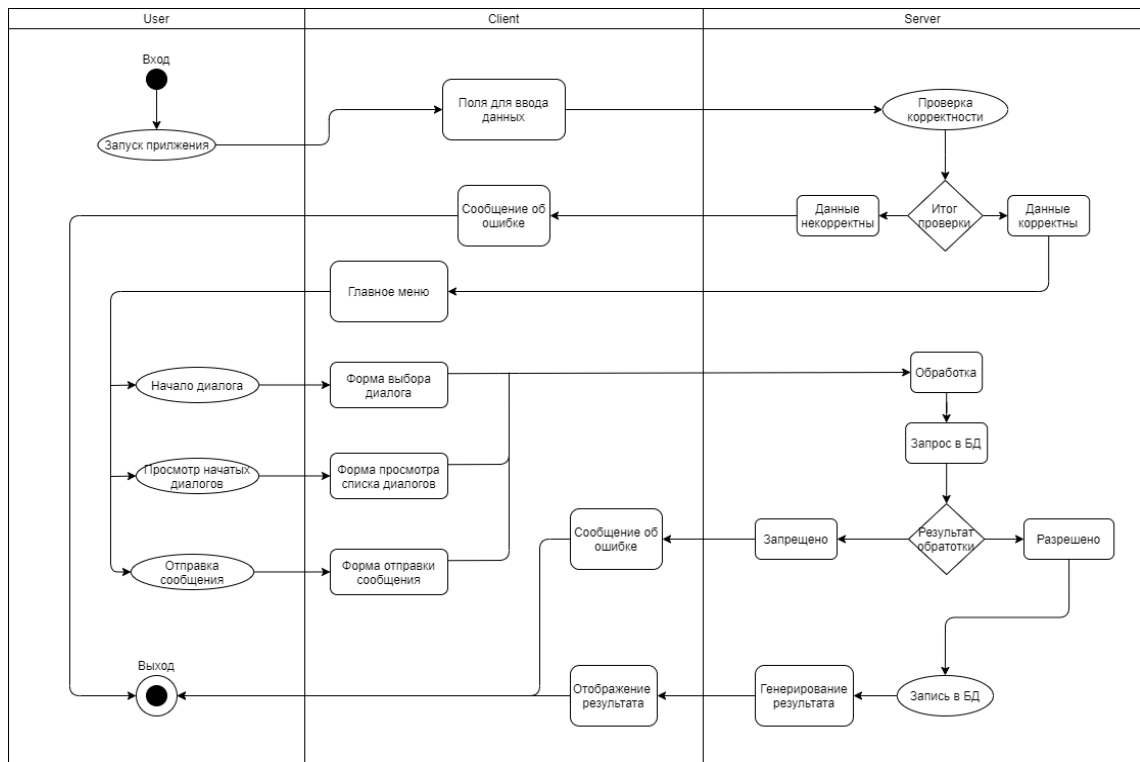


Рисунок 7 - Диаграмма активности

### 2.3.5 Развертывание приложения

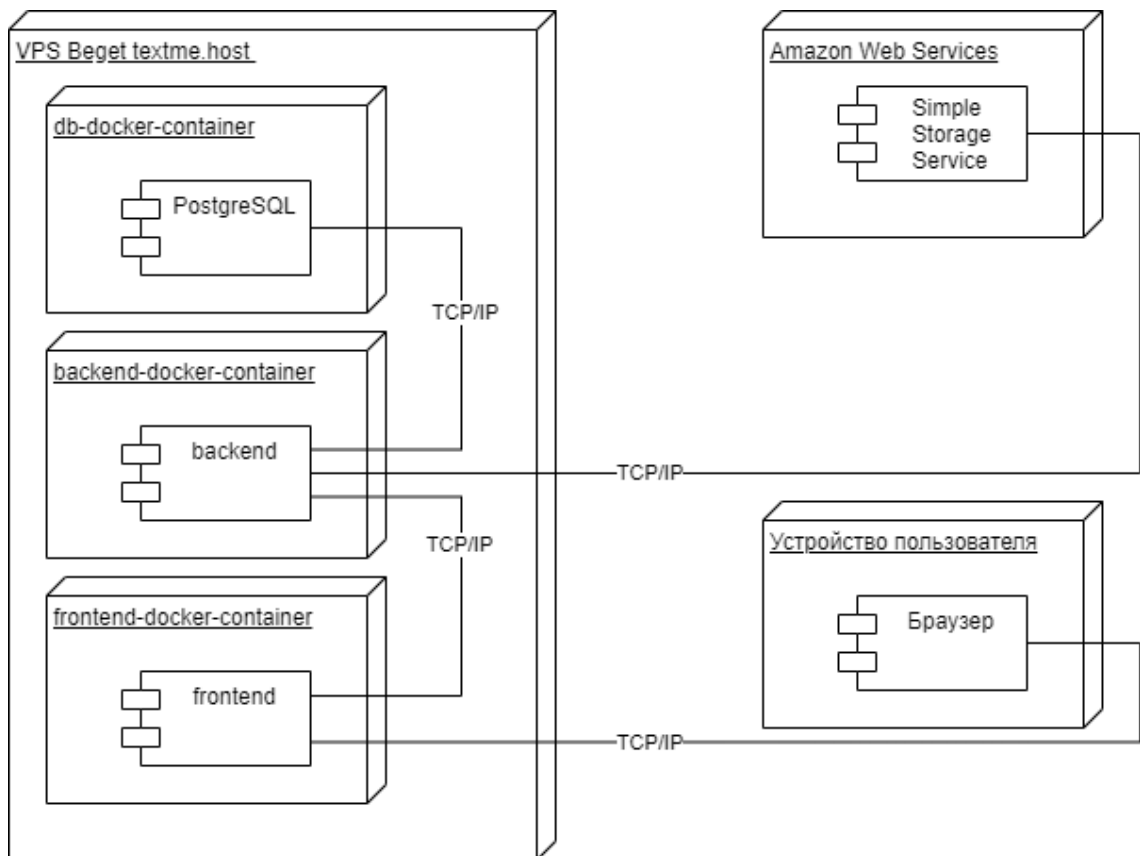


Рисунок 8 - Диаграмма развертывания

На рисунке 8 представлена диаграмма развертывания, которая олицетворяет действительное размещение компонентов и блоков всего приложения. База данных PostgreSQL, SpringBoot и Angular приложения размещены в Docker контейнеры, которые запущена на виртуальном персональном сервере от Beget. Backend имеет связь с серверами Amazon Web Services, а именно с хранилищем AmazonS3, в котором хранятся все файлы, загружаемые пользователями. Пользователь на любом своем устройстве запускает браузер, и через него подключается к frontend модулю приложения.

### 2.3.6 Диаграмма классов

Декомпозированная диаграмма классов представлена на рисунках 9-11 и показывает основные сущности системы, связи между ними и их атрибуты.

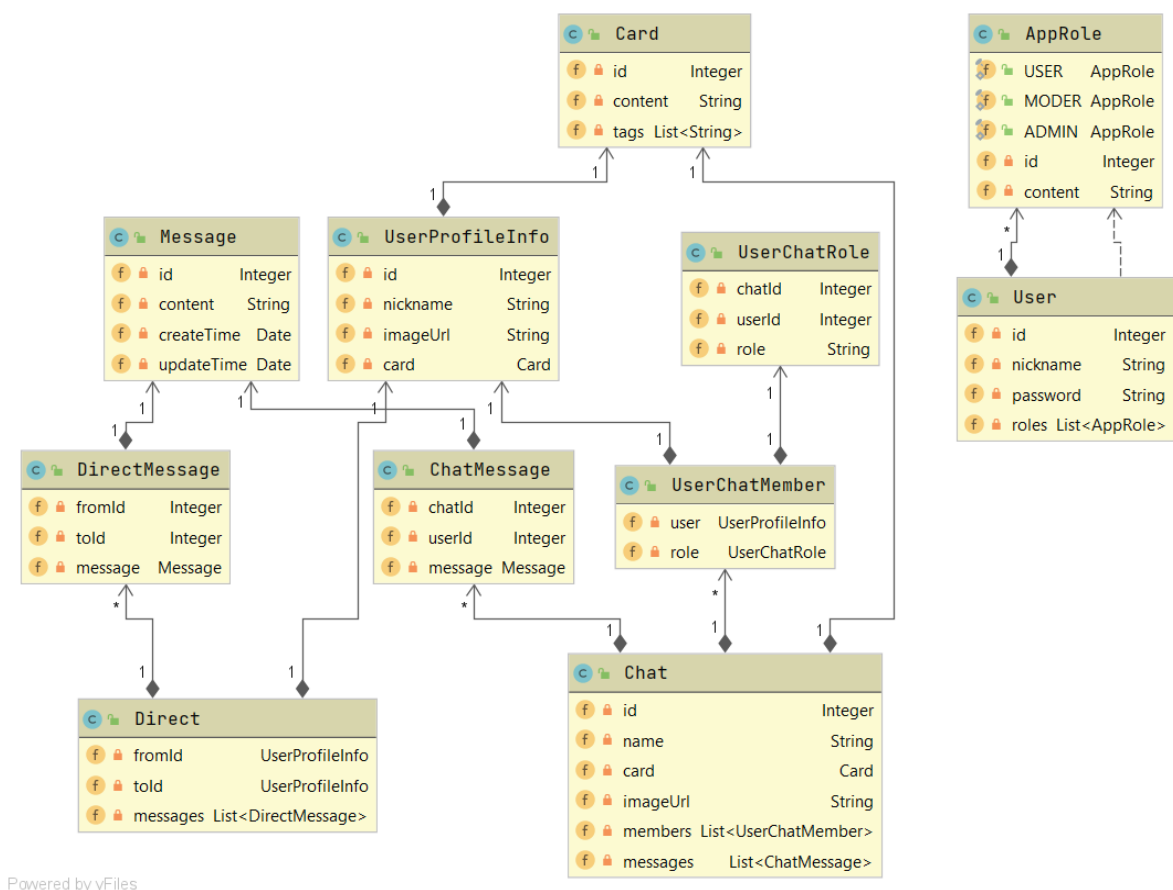


Рисунок 9 - Диаграмма классов. Сущности

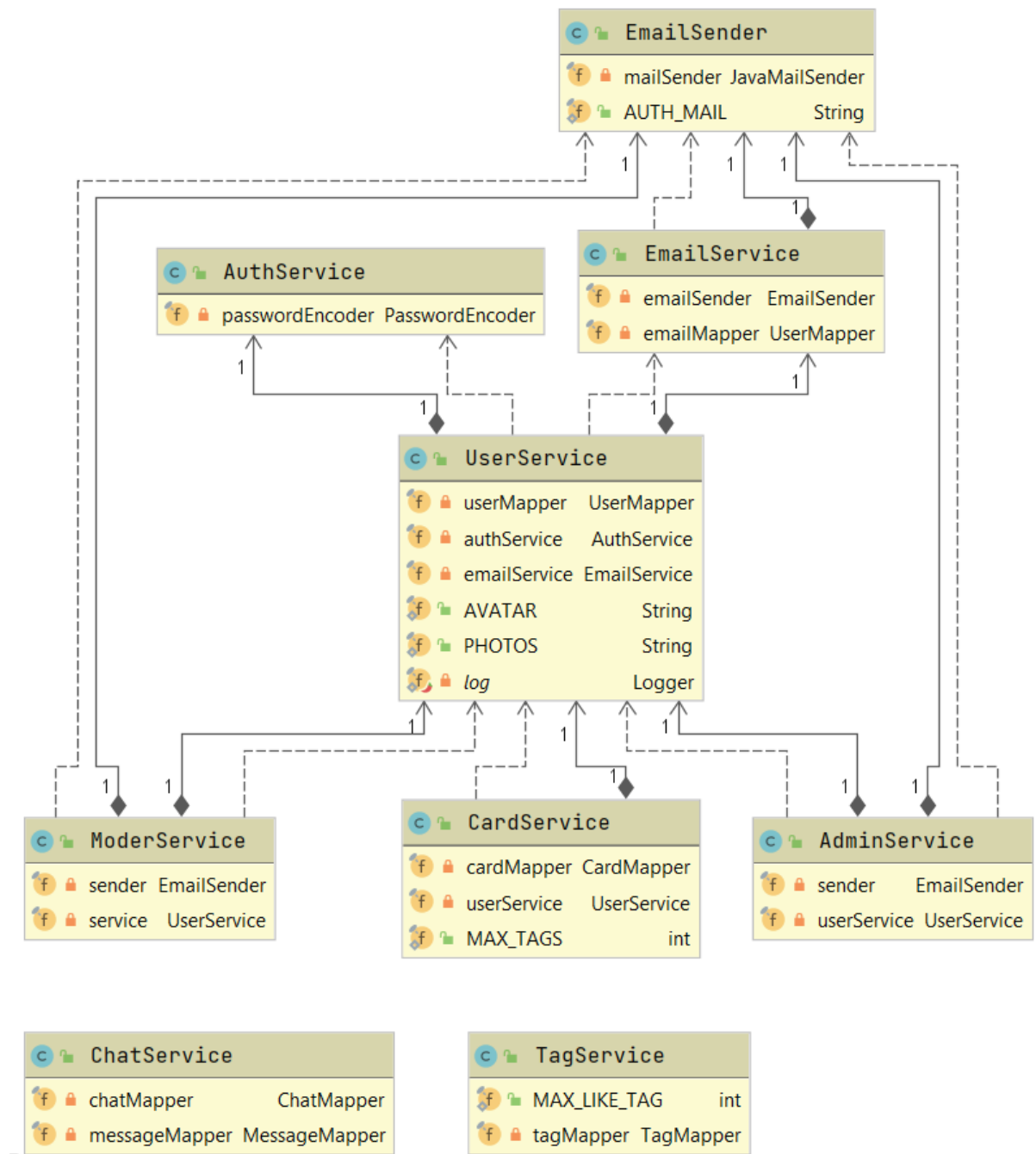


Рисунок 10 - Диаграмма классов. Сервисы

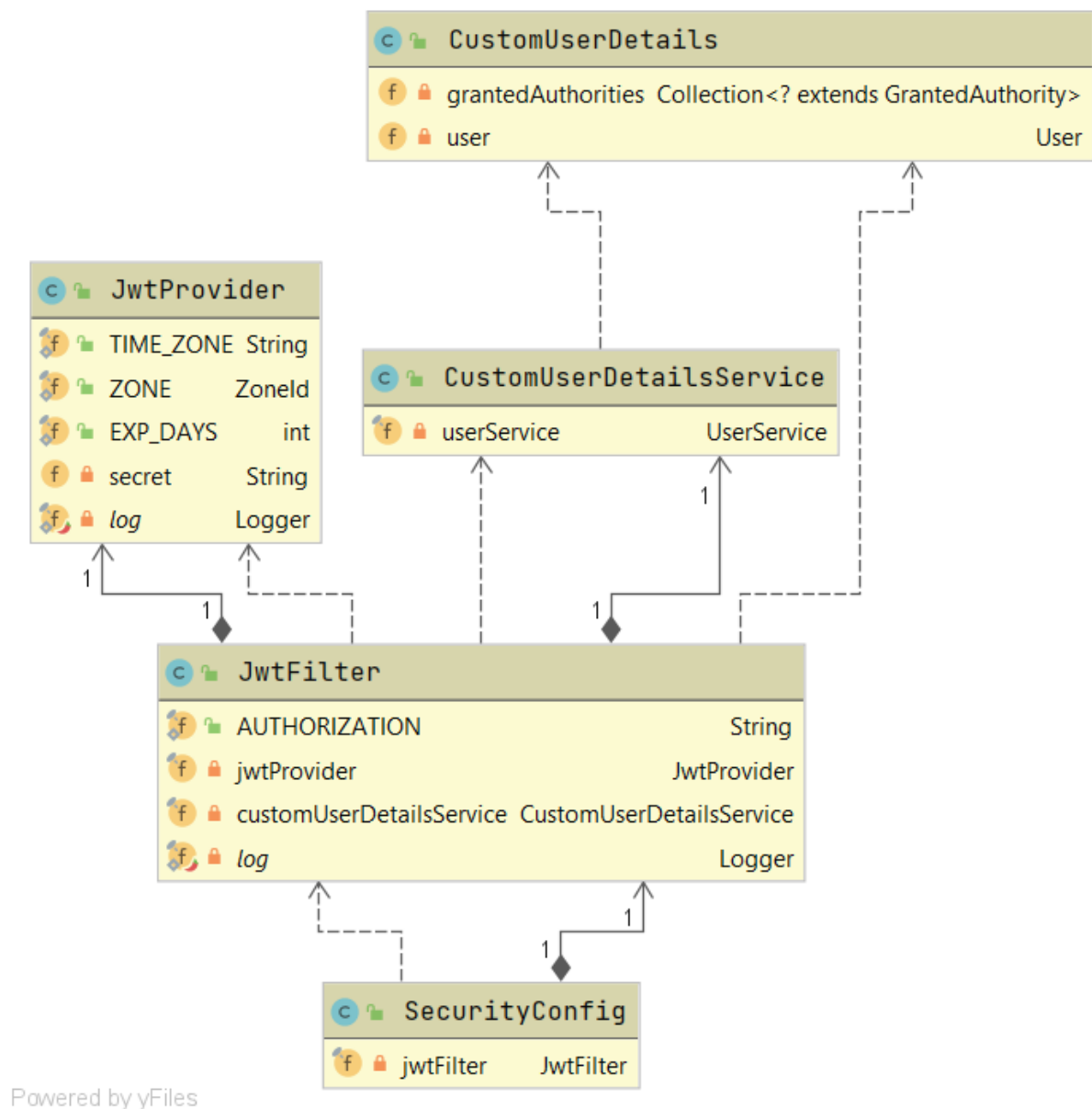


Рисунок 11 - Диаграмма классов. Авторизация

### 2.3.7 Диаграмма объектов

Для лучшего понимания диаграммы классов на рисунке 12 представлена диаграмма объектов, на которой отображены объекты классов сущностей в некоторый момент времени.

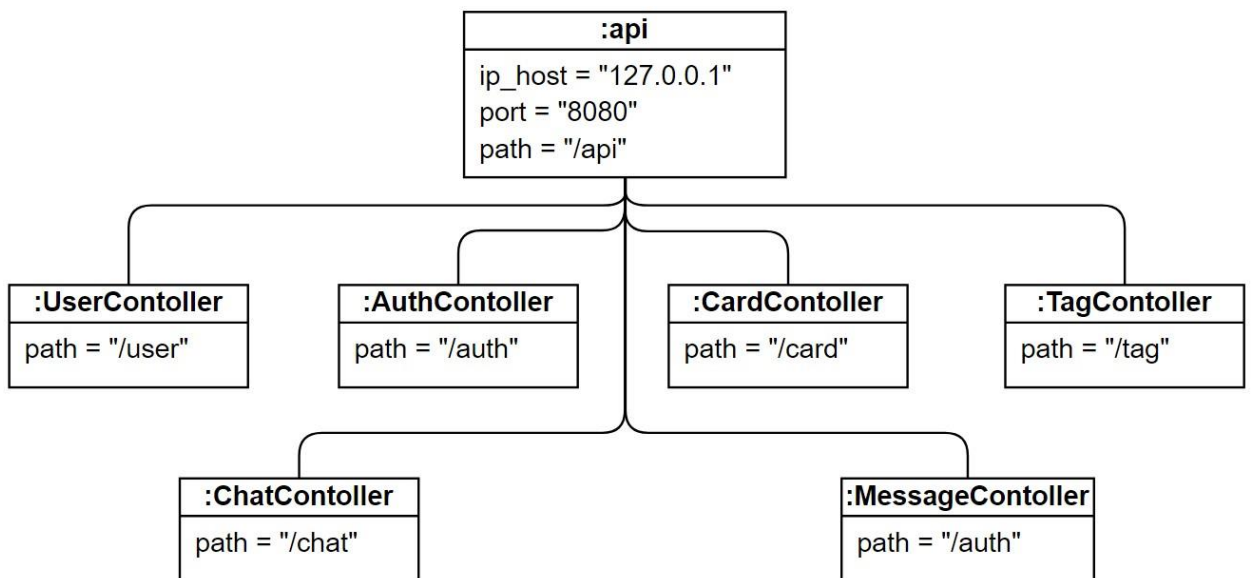


Рисунок 12 - Диаграмма объектов

### 2.3.8 IDEF0

На рисунке 13 изображена диаграмма IDEF0, иллюстрирующая работу системы. Работу сервиса регулирует законодательство РФ. Для функционирования системы необходима администрация. На вход в систему поступает человек, который хочет найти собеседника, для обсуждения интересных тем.

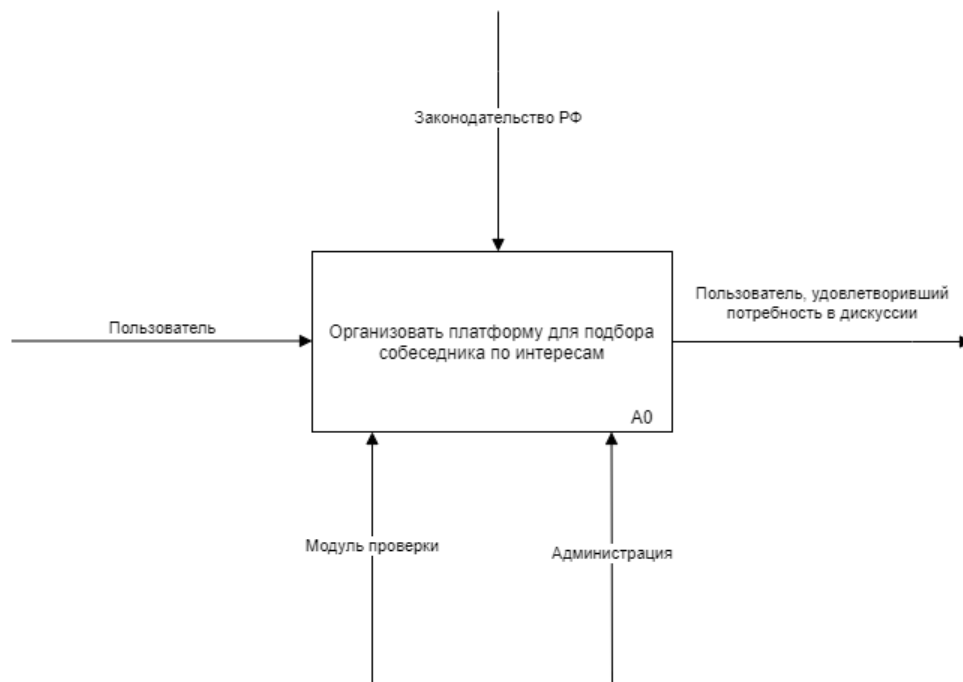


Рисунок 13 - Диаграмма IDEF0

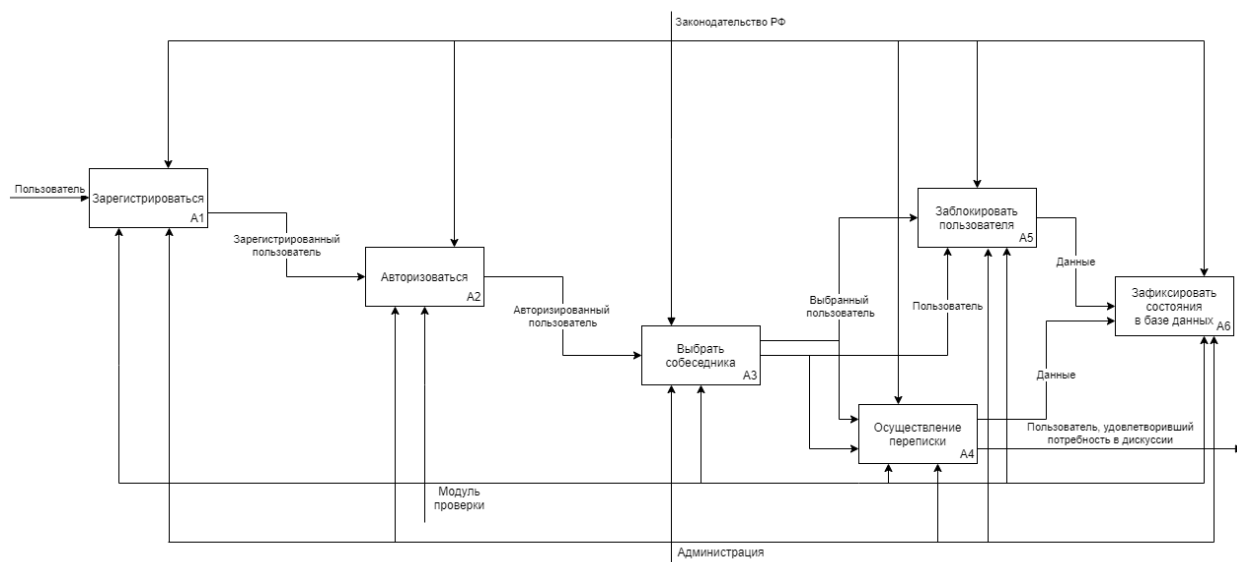


Рисунок 14 - Диаграмма декомпозиции первого уровня

## 2.4 Воронка конверсии

Аналитическая воронка, продемонстрированная на рисунке 15, демонстрирует примерное количество людей, в процентах, дошедших до совершения целевого действия, в данном случае, рекомендация посетить сайт, после регистрации и опробования предоставляемых сайтом возможностей.

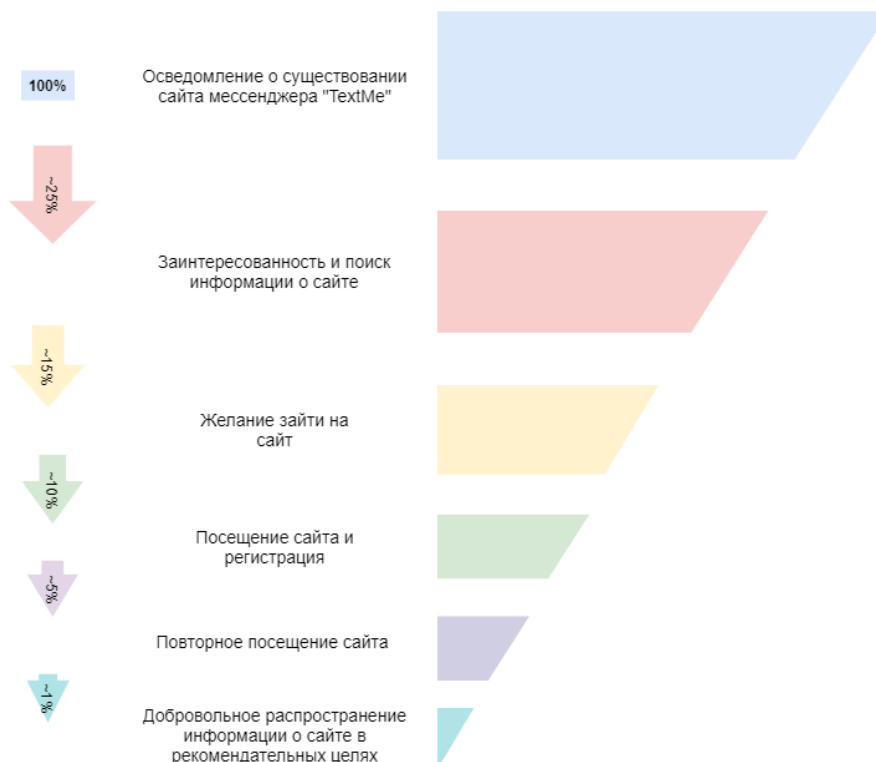


Рисунок 15 - Воронка конверсии



## 2.5 Яндекс.Метрика

Яндекс.Метрика – интернет-сервис, предназначенный для оценки посещаемость веб-сайтов и анализа поведения пользователей. С помощью специально настроенных целей можно отслеживать активность пользователя в тех ситуациях, которые максимально интересны владельцу сайта.

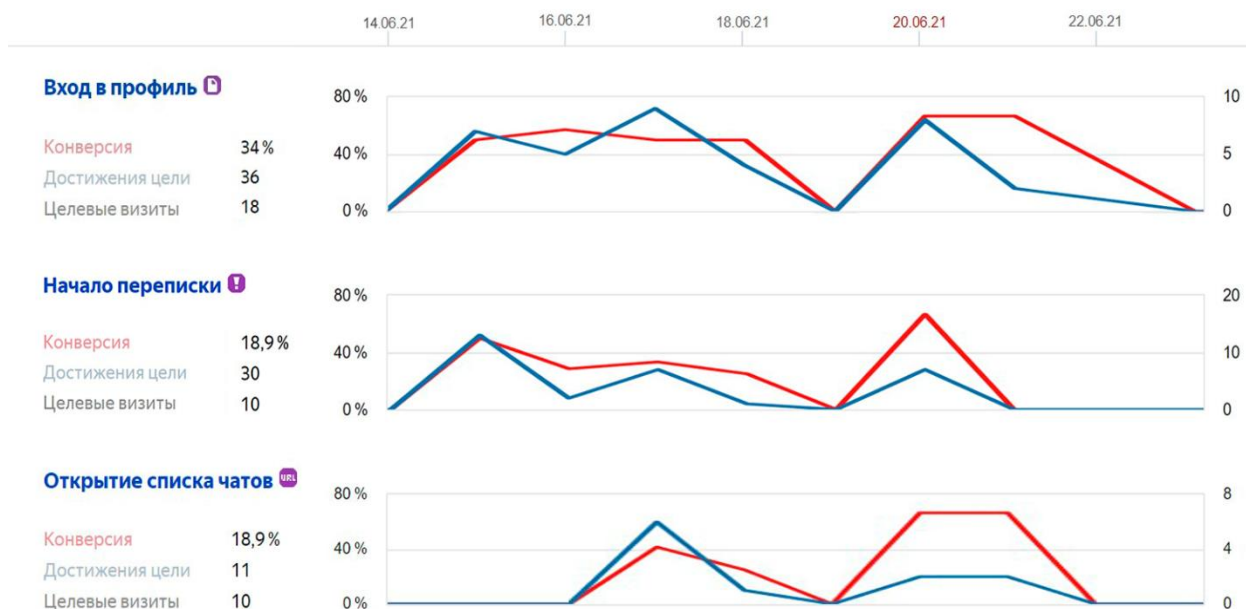


Рисунок 16 - Метрика, полученная по основным целям

### 2.5.1 Цель «Вход в приложение»

Использование этой цели поможет определить то, как часто пользователи возвращаются в мессенджер или входят в свой аккаунт с различных устройств. Такие данные необходимы для того, чтобы понять какой популярностью пользуется мессенджер.

Для успешного достижения этой цели пользователю необходимо выполнить следующие действия:

- Открыть сайт
- Перейти с стартовой страницы на страницу авторизации
- Ввести логин и пароль
- Войти в приложение

Достижением цели является отправка формы с логином и паролем.

### **2.5.2 Цель «Начать беседу»**

Так как основной задачей мессенджера является осуществление коммуникации между пользователями, то не менее важной является цель поиска нового собеседника и вступление с ним в диалог.

Для успешного выполнения этой цели авторизированному пользователю необходимо выполнить следующие действия:

- Зайти на главную страницу приложения
- Из набора карточек выбрать понравившуюся
- Нажать на кнопку «начать беседу»
- Перейти к переписке с пользователем

Цель является достигнутой после того, как выполнилось JavaScript событие нажатия на кнопку «начать беседу».

### **2.5.3 Цель «Список чатов»**

Не менее важной статистикой является статистика того, сколько пользователей вернулось к ранее начатым чатам, то есть сколько пользователей захотело и дальше пользоваться мессенджером и продолжать общение с ранее найденными собеседниками.

Для выполнения этой целевой задачи авторизированному пользователю необходимо перейти на страницу «мессенджер» и выбрать любой из начатых чатов.

### **2.5.4 Анализ пользователей**

С помощью Яндекс.Метрики была полученная статистика возраста пользователей, посещавших данный мессенджер. На диаграмме, продемонстрированной на рисунке 17, видно, что мессенджер посещали люди различного возраста, но можно явно выделить тот факт, что мессенджер является особо популярным среди людей возрастом от 18 до 24 лет.

Статистика используемых пользователями устройств продемонстрировано на рисунке 18. На ней явно видно, что чаще всего пользователи посещают данный сайт с персональных компьютеров. Далее по

популярности идут смартфоны. Скорее всего это связано с тем, что мессенджер реализован в формате веб-приложения и он корректнее работает на компьютерах, чем мобильных устройствах. В свою очередь, посещений с мобильного телефона больше 25%, это говорит о том, что четверть всех пользователей устраивает такой вариант использования приложения.

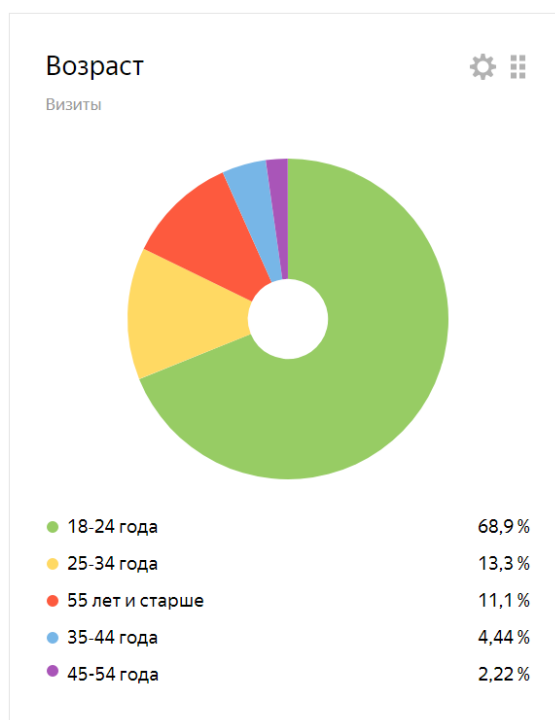


Рисунок 17 - Возраст пользователей

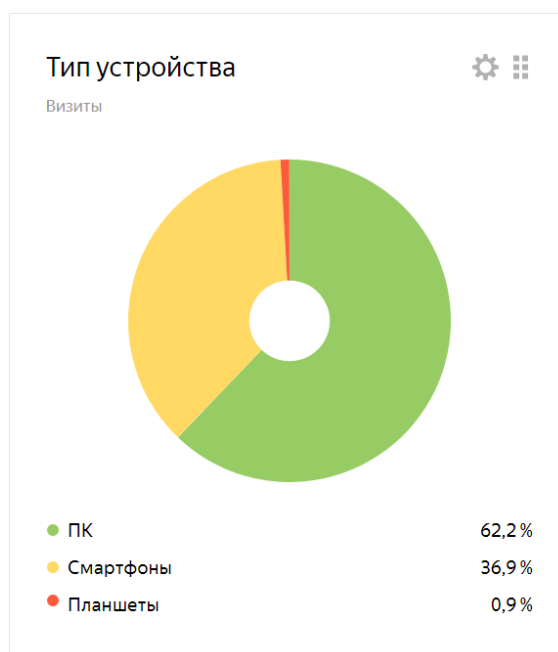


Рисунок 18 - Устройства пользователей

## **3 Реализация приложения**

### **3.1 Средства реализации**

Для реализации программно-аппаратной части были выбраны следующие технологии:

- Язык программирования Java;
- Фреймворк Spring Boot;
- СУБД PostgreSQL;
- Фреймворк MyBatis;
- Инструмент миграции баз данных Liquibase;
- Система автоматической сборки Gradle.

Для реализации клиентской части были выбраны следующие технологии:

- Фреймворк Angular;
- CSS фреймворк UIKit;
- Язык разметки HTML;
- Язык программирования TypeScript;

### **3.2 Реализация Backend**

#### **3.2.1 Пользователь и регистрация**

Точкой входа в большинство веб-приложений является регистрация и авторизация. Чтобы это сделать, пользователю необходимо заполнить соответствующие формы на клиентской части приложения. Есть три варианта взаимодействия пользователя с серверной частью на этапе входа в приложение:

- Регистрация
- Подтверждение электронной почты
- Авторизация

При первом входе в приложение пользователь должен зарегистрировать свой профиль, введя уникальные никнейм и электронную почту, пароль. После проверки на уникальность введенных полей в базу данных вносится новая

запись о пользователе, а также о новой зарегистрированной почте, на которую будет прислано письмо для подтверждения регистрации (рисунок 19).

После перехода по ссылке, которая будет указана в сообщении аккаунту будет выдана роль USER. Важно отметить, что до тех пор, пока почта не подтверждена, без роли USER нельзя использовать основное API приложения.

Для рассылки почты пользователям была создана специальная почта для приложения, а в качестве технологии отправки сообщения выбрана библиотека spring-boot-starter-mail с использованием протокола smtps.

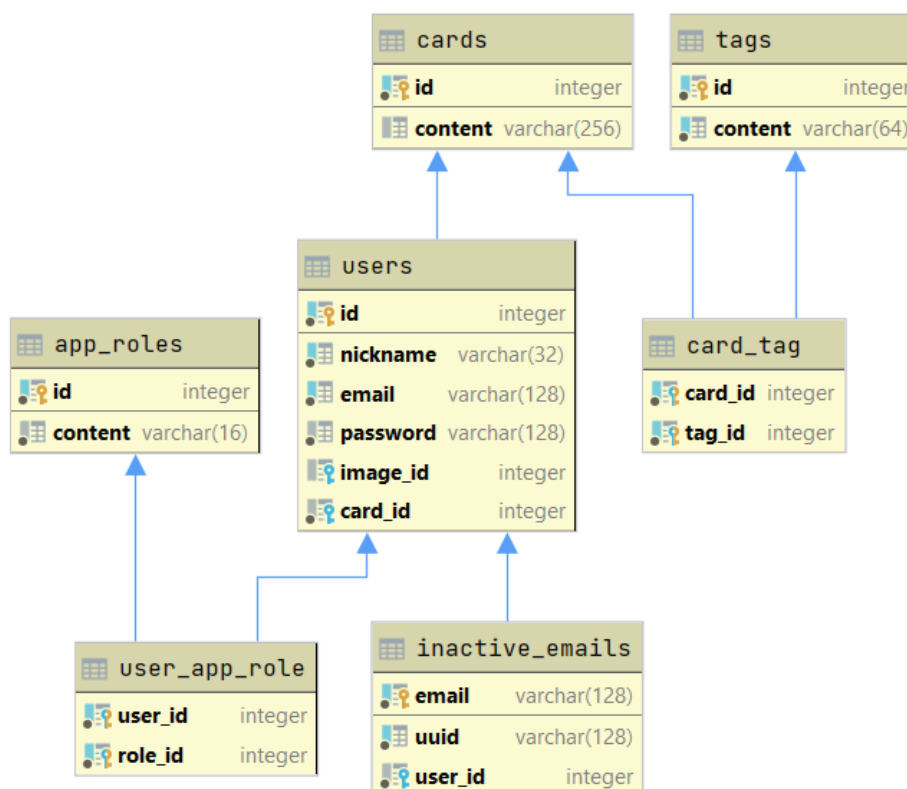


Рисунок 19 - Схема таблицы users в базе данных

### 3.2.2 Заполнение карточки

После успешного входа в приложение пользователю доступен интерфейс для заполнения информации о своих интересах. Эта информация представлена в форме набора тегов (до 16-ти), указывающих на то, какие интересы имеет пользователь и чем он увлекается. С помощью простого SQL-запроса, на основе начала введённого тега можно предложить пользователю самые популярные на данный момент теги.

Ещё одним параметром в профиле является статус, используется в качестве краткого собственного описания пользователя. В базе данных имеется таблица, описывающие все возможные теги, которые когда-либо были привязаны к карточкам. Чтобы избежать совпадений одинаковых слов с символами в разном регистре, при вставке в базу каждый новый тег приводятся к нижнему регистру.

Для загрузки изображения профиля к приложению было подключен облачный сервис от Amazon. Со стороны сервера можно выделить следующие этапы загрузки нового изображения:

- Валидация фреймворком Spring размера Multipart файла (по умолчанию 1 МБ, можно настроить, определив переменную окружения `spring.http.multipart.max-file-size`).
- Валидация типа файла на расширение JPEG / PNG
- Конвертация в JPEG, если полученное изображения формата PNG для сжатия файла и, соответственно, уменьшения его размера в конечном хранилище.
- Генерация URL файла для облачного сервиса и его сохранение в нём.
- Сохранение URL файла в базе данных (рисунок 20) и установка пользователю нового ID его аватара, если до этого поле ID было равно NULL.
- Отправка пользователю ответа с URL его нового изображения профиля.

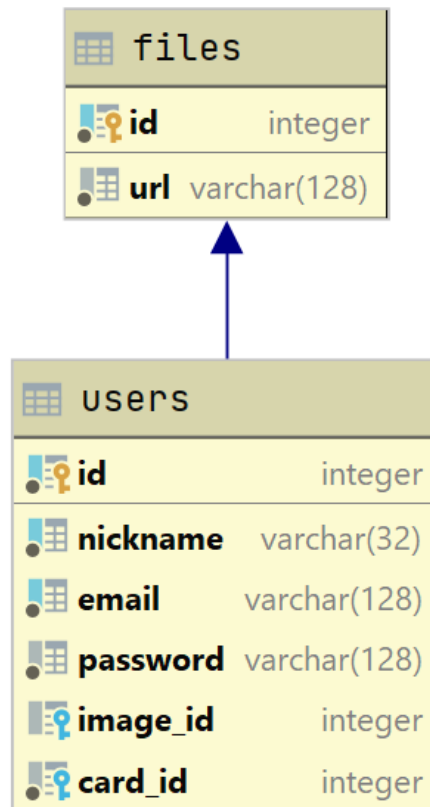


Рисунок 20 - Изображения в базе данных

### 3.2.3 Беседы

Беседы служат для собрания группы пользователей, имеющих общие интересы в одном месте. У беседы, как и у пользователя, существует собственная карточка, которую можно точно так же заполнить. Чтобы определить логику пользования беседой, были разработаны вспомогательные таблицы (рисунок 21), которые определяют текущий статус и роль участника беседы. Приведём название и описание функций каждого из них ниже:

Статусы:

- **MEMBER** (участник). Является флагом того, что человек вообще является участником беседы.
- **LEAVE** (вышедший). Обозначает вышедшего по своей воле участника. Он не будет получать новые сообщения из беседы, но в любой момент может вернуться в неё.

— **BLOCKED** (выгнанный). Обозначает выгнанного участника. Если у пользователя такой статус в беседе, он не может вернуться сам.

Роли:

— **MEMBER** (обычный участник). Даёт основные права для написания сообщений в чат.

— **MODER** (модератор). Позволяет выгонять участников из чата и удалять из беседы чужие сообщения.

— **ADMIN** (администратор). Имеет права модератора, а также назначает и снимает их из числа обычных участников, а также изменять содержимое карточки чата.

— **OWNER** (владелец). Даёт все права на изменение ролей и статусов каждого из участников беседы.

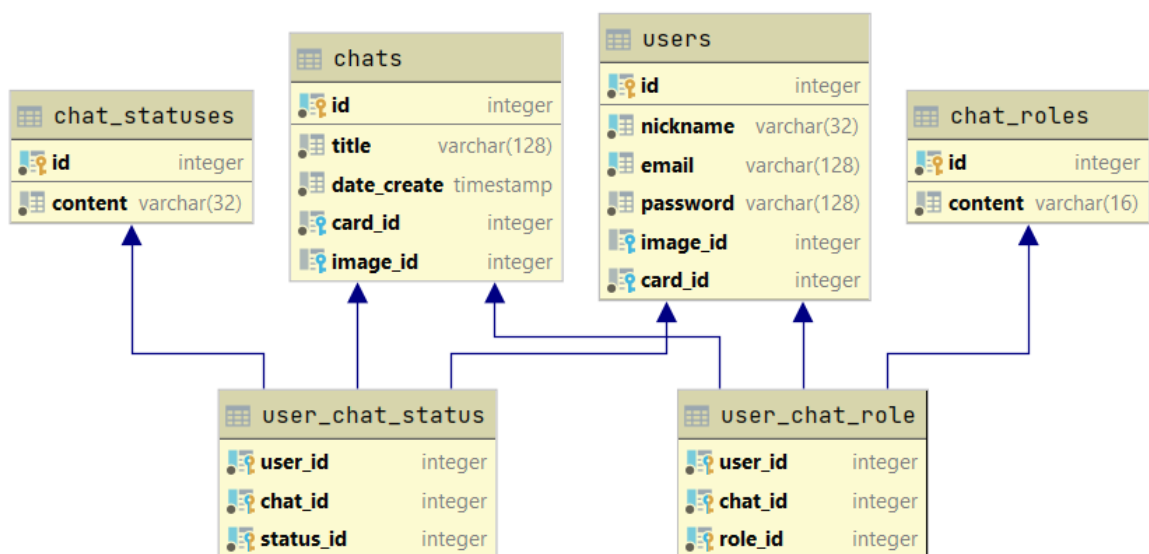


Рисунок 21 - Беседы в базе данных

### 3.2.4 Подбор собеседников

Главной особенностью разрабатываемого приложения является поиск собеседника по собственным интересам. Для использования данной возможности было разработано API, позволяющее искать профили отдельных пользователей или целых чатов по следующим выборкам:



- Отдельно выбранный тег.
- Максимальное совпадение с собственной карточкой.
- Случайность.

В первом случае производится простой поиск из числа всех пользователей приложения по введённому тегу и возвращается случайная выборка пользователей страница с этим тегом.

Во втором случае производится сортировка всех карточек по количеству совпадений каждой с карточкой пользователя, запросившего подбор. Таким образом, первыми в списке всегда будут отображаться те люди, с которым самое большое пересечение по интересам.

Третий тег существует для людей, которые просто хотят найти интересные для себя новые профили.

Для сохранения карточки понравившегося пользователя служит сущность favorites (рисунок 22). В ней можно добавить карточки тех людей, которые понравились пользователю, но по какой-то причине он не может написать им прямо сейчас.

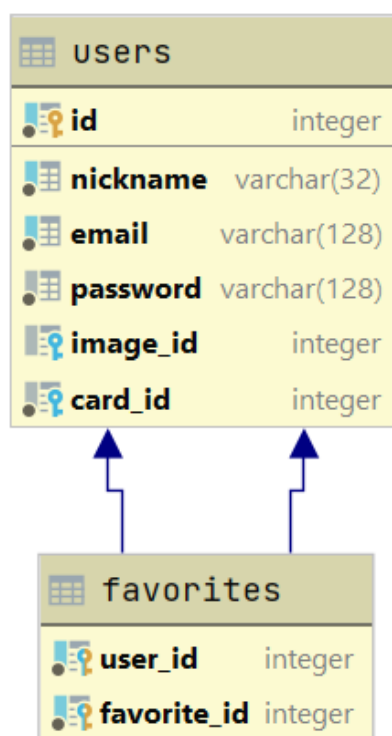


Рисунок 22 - Сущность избранных пользователей

### 3.2.5 Мессенджер

Одна из основных целей разрабатываемого приложения – установка соединения между двумя пользователями посредством их общения в чате.

Главной сущностью, определяющей логику переписок, является `messages` (рисунок 23). Она описывает сообщение, отправленное пользователем другому пользователю или в чат и хранит в себе непосредственно текст самого сообщения, время его создания и последнего обновления.

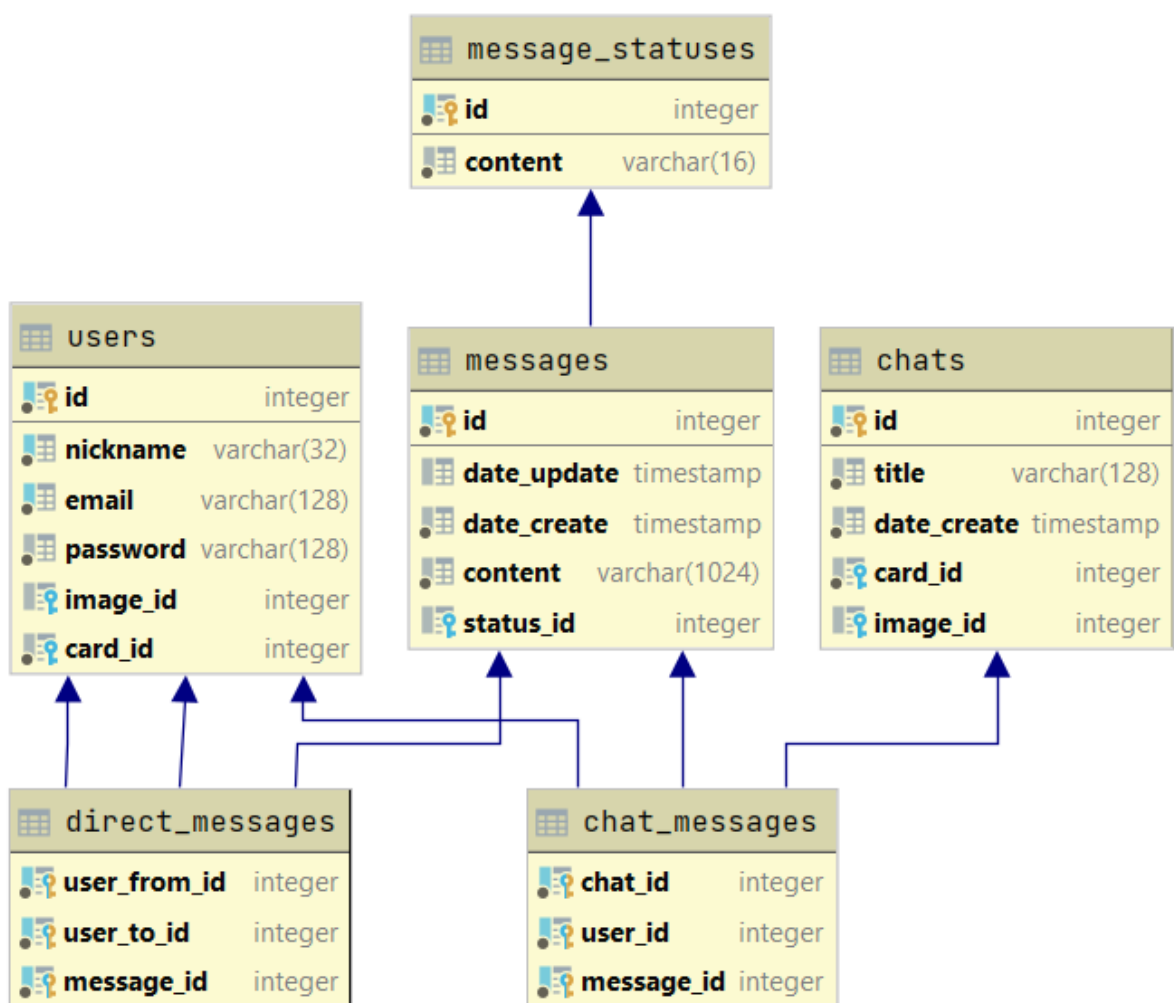


Рисунок 23 - Представление сообщений в базе данных

Существует вспомогательная таблица, в которой хранится статус сообщения. Он отвечает за отображение сообщения на клиентской части приложения:

- Отправлено. Говорит клиенту о том, что сообщение было отправлено адресату, но ещё не прочитано.
- Получено. Говорит клиенту о том, что сообщение было успешно доставлено и прочитано.
- Удалено. Сообщения не показываются в чатах и отсеиваются в каждом запросе.

Сам механизм коммуникации был реализован с помощью STOMP — протокола обмена сообщениями, работающего поверх web-socket, созданный предельно простым. Он основан на фреймах по образцу HTTP. Фрейм состоит из команды, необязательных заголовков и необязательного тела. Другими словами, при помощи данного протокола, можно осуществлять связь между сервером и клиентом почти также, как через обычные HTTP запросы.

Для выбранного протокола были определены четыре команды (пути API) по которым происходит непосредственная работа с сообщениями в личных сообщениях и чатах:

- /send-message
- /update-message
- /read-message
- /delete-message

Сначала клиент соединяется с сервером при помощи метода CONNECT протокола STOMP. Если он авторизован в приложении, соединение происходит успешно. После чего, с помощью метода SUBSCRIBE клиент подписывается на все пути по адресу /queue/messenger который отвечает за рассылку пользователям уведомлений о каких-либо изменениях в чате, будь то новое сообщение, его успешная доставка или удаление.

Чтобы применить изменения к сообщению после совершения каких-либо манипуляций с ним, клиент отправляет по web-socket соединению сообщение на соответствующий путь. Есть два случая отправки сообщения —

в чат или в директ. В каждом из них сервер, в первую очередь, авторизовывает пользователя.

Ограничения, накладываемые на сообщения:

- Изменить статус сообщения на «прочитано» можно только чужому сообщению.
- Можно изменить или удалить только своё сообщение.

Естественно, находясь в чёрном списке другого пользователя или целой беседы, нельзя никак взаимодействовать с ним. При каждом запросе заблокированные пользователи отфильтровываются и им приходит соответствующее уведомление на канал ошибок `/queue/chat/errors`.

### **3.2.6 Администрирование**

Для поддержания порядка в приложении была создана система управления правами (рисунок 19) внутри приложения и сущности жалоб/репортов (рисунок 24) с помощью которых пользователи смогут сообщить администрации сайта о каких-либо нарушениях в карточках других пользователей или чатов. Существуют две роли, принадлежащие к группе администрации сайта:

- Модератор. Осуществляет просмотр жалоб пользователей и принимает решение по ним.
- Администратор. Может выполнять функции модератора, но помимо этого способен назначать и снимать с должности последних.

Если права модератора могут быть выданы администратором при помощи графического интерфейса на стороне клиента, то запрос на выдачу прав самого администратора вносится в файл миграции базы данных или выполняется вручную из консоли.



В случае блокировки модератор выбирает время, на которое будет заблокирован пользователь, после чего клиентом отправляется запрос на сервер на соответствующий API. Далее, производится автоматическое внесение ID и даты рассмотрения для всех ещё не рассмотренных жалоб на заблокированного пользователя, так как следует считать, что модератор рассмотрел все жалобы на данного пользователя, помимо той, что появилась первой в списке, и принял решение на основании их всех.

Блокировка. В случае, если пользователя заблокировали, каждый его запрос блокируется до тех пор, пока не истечёт срок, установленный модератором.

### **3.3 Интерфейс пользователя**

#### **3.3.1 Эмблема приложения**

Для данного проекта был придуман и разработан логотип, продемонстрированный на рисунке 25. Он объединяет в себе два признака формирования текста из разных времен. Первый признак – это наконечник перьевой ручки, который олицетворял написание текста в доцифровую эпоху. Второй признак – это троеточие, которые в нынешнюю эпоху практически во всех мессенджерах показывают, что собеседник набирает сообщение.



Рисунок 25 - Эмблема мессенджера «TextMe»

### 3.3.2 Стартовые страницы

При первой загрузке сайта, пользователь автоматически перенаправляется на стартовую страницу (рисунок 26).

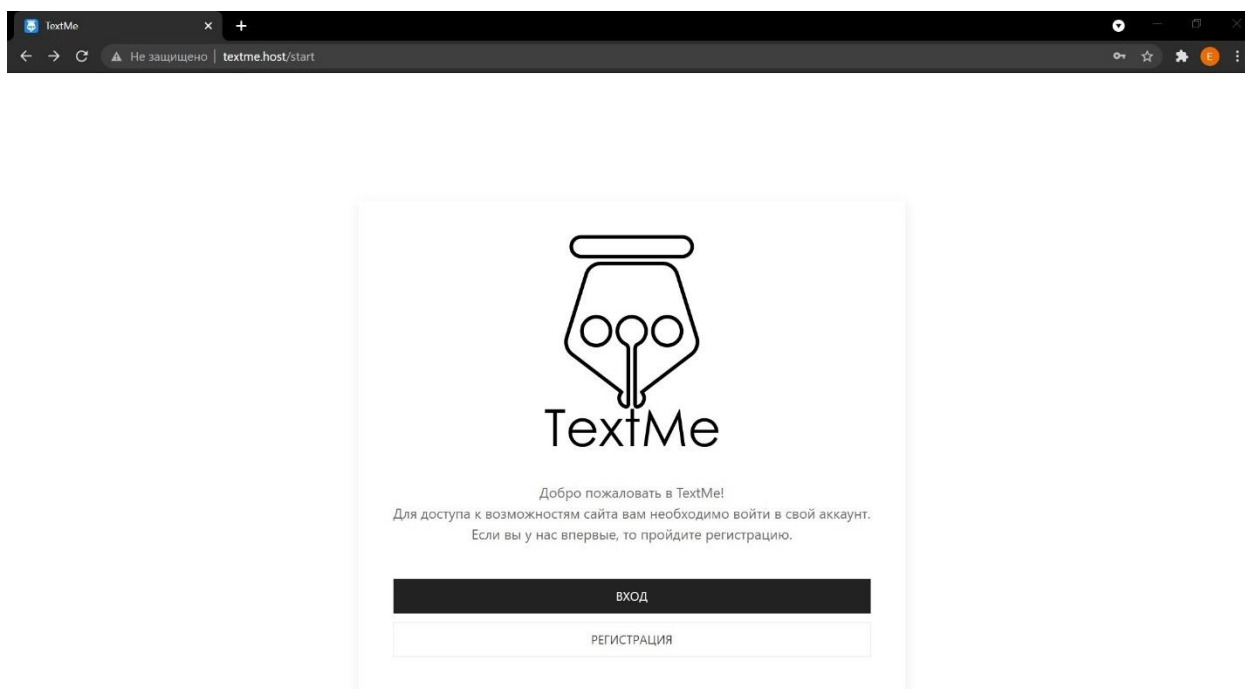


Рисунок 26 - Стартовая страница

Здесь он может увидеть приветствие и краткую памятку о том, что необходимо иметь аккаунт на данном сайте. После нажатия на кнопку «регистрация» пользователь попадает на страницу с формой регистрации (рисунок 27).

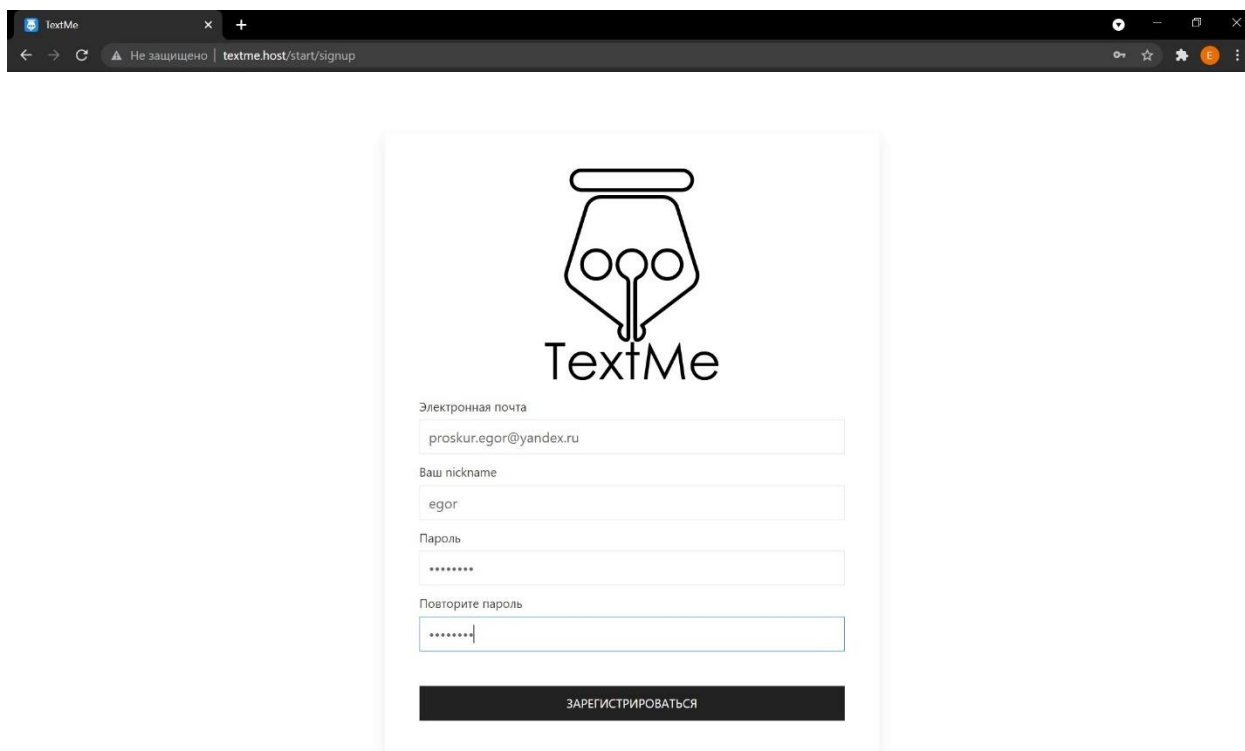


Рисунок 27 - Страница регистрации

После ввода всех необходимых данных происходит проверка их на правильность. Если есть ошибки, то пользователь увидит сообщение об ошибке, написанное красным текстом, после чего он сможет исправить ошибку и попробовать еще раз.

Если при заполнении формы ошибок не обнаружено, то запрос уходит на программно-аппаратную часть, после чего, получив ответ с нее, пользователю выводится диалоговое окно, продемонстрированное на рисунке 28, с просьбой перейти в свой почтовый ящик и подтвердить регистрацию в приложении.



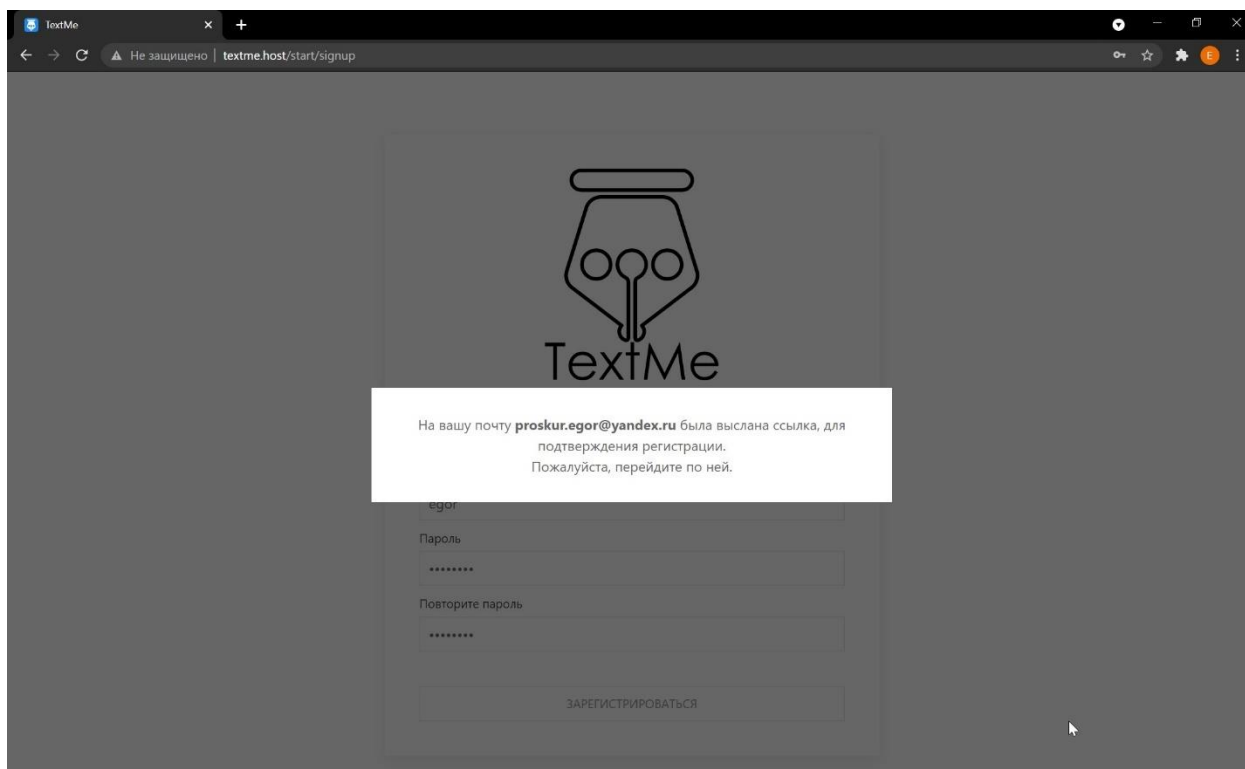


Рисунок 28 - Диалоговое окно с просьбой проверить почтовый ящик

На рисунке 29 продемонстрировано письмо, которое получает пользователь при регистрации. Перейдя по ссылке, почта автоматически подтверждается, и пользователь оказывается на странице, продемонстрированной на рисунке 30.

Activation link



auth.textme@ya.ru auth.textme@ya.ru  
Вам: proskur.egor@yandex.ru

сегодня в 12:34

<http://textme.host/start/activate/MTZjYzY0ZTUtZWU5NC00ZDg2LTg3NmYtNGJlY2E0ZGEwZmNm>

Рисунок 29 - Письмо с ссылкой на подтверждение регистрации

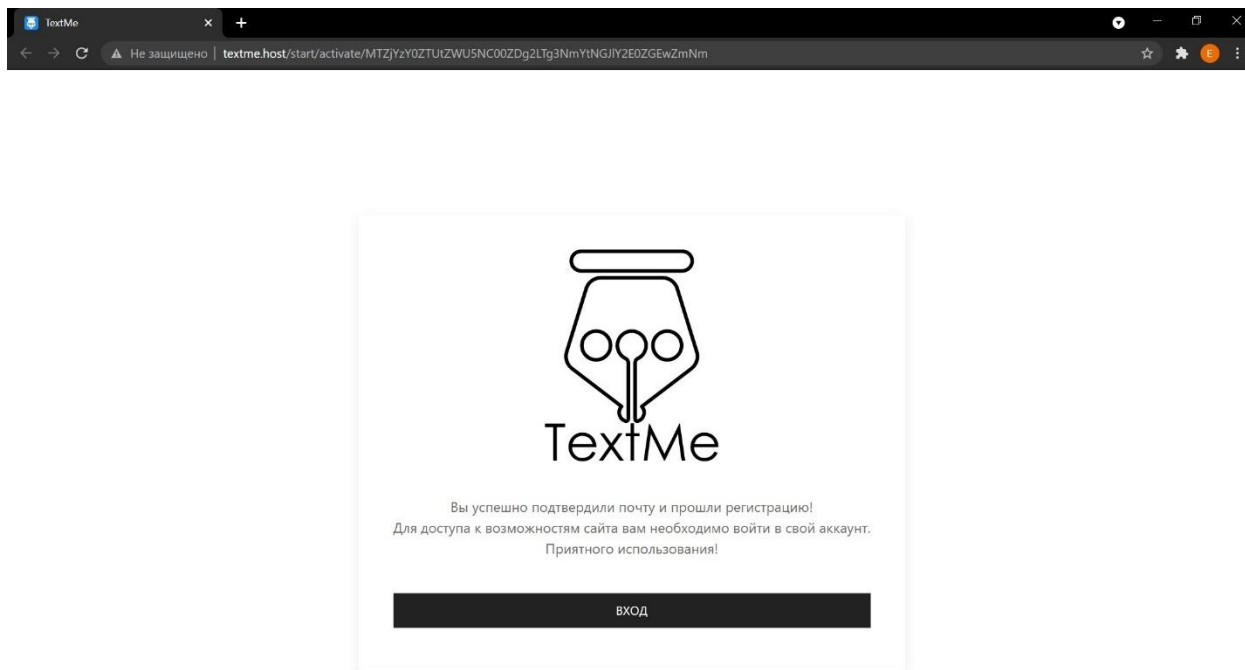


Рисунок 30 - Страница, подтверждающая успешную регистрацию

На данной странице пользователь видит сообщение об успешной регистрации и ему предлагают пройти авторизацию и войти в приложение. Нажав на кнопку «вход», пользователь перенаправляется на страницу авторизации, которая показана на рисунке 31.

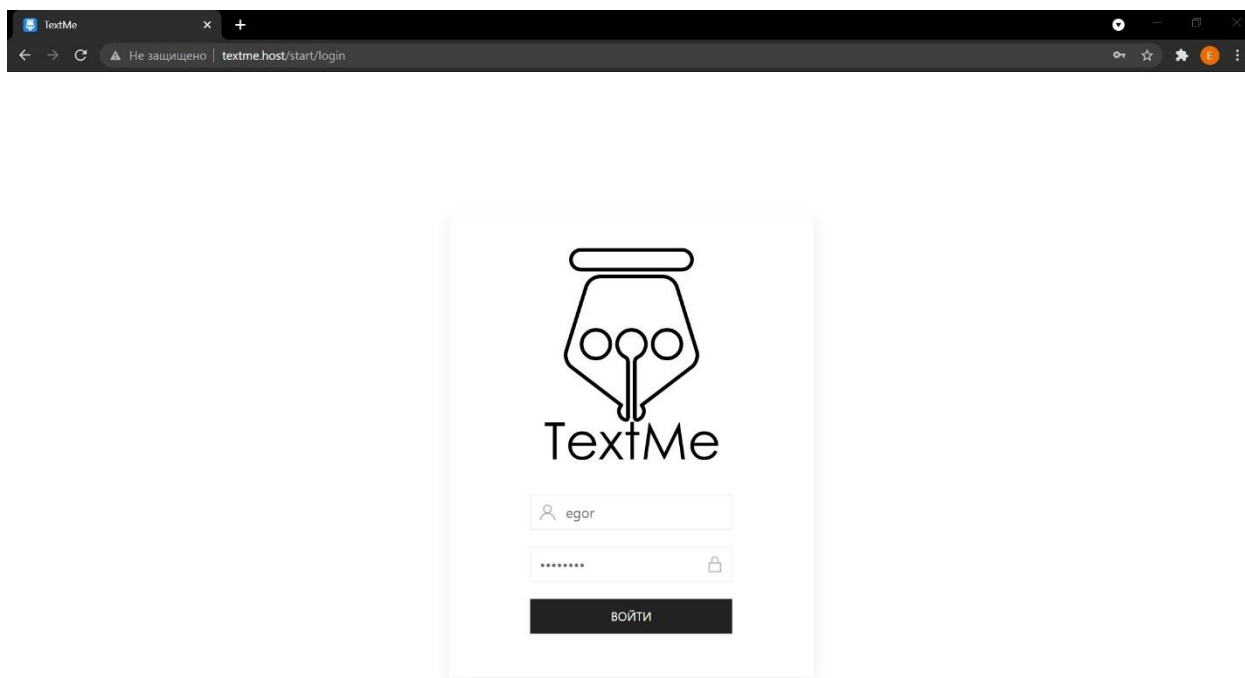


Рисунок 31 - Страница авторизации

Введя в форму в качестве логина свое имя или электронную почту, а в качестве пароля пароль, который был указан при регистрации, пользователь может нажать на кнопку «Войти», после чего будет отправлен запрос на backend. Если пользователь ошибся, то на форме отобразится ошибка, продемонстрированная на рисунке 32.

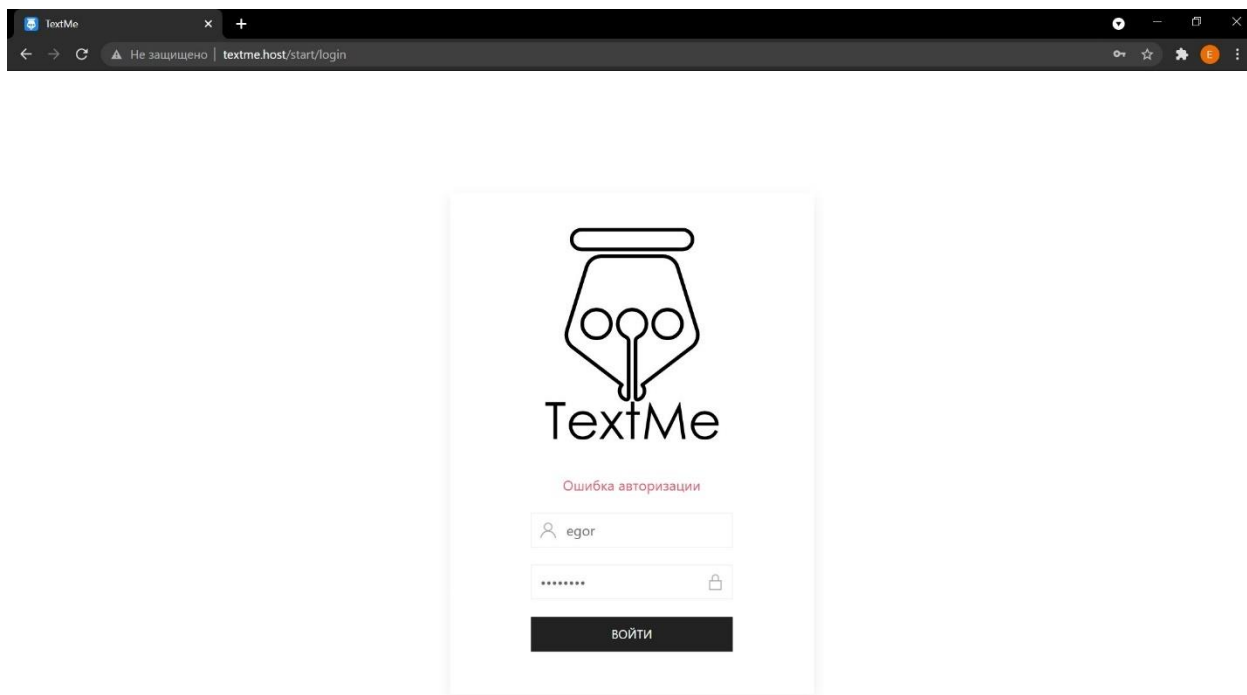


Рисунок 32 - Ошибка авторизации

Проверив логин и пароль, пользователь может повторить попытку, и в случае успеха, сайт перенаправит его на главную страницу приложения, продемонстрированную на рисунке 33.

### 3.3.3 Главная страница

Вверху главной страницы находится навигационная панель. В левой части расположена эмблема и название приложения. Далее следуют навигационные кнопки «главная», «мессенджер», «профиль», которые ведут на соответствующие страницы. В правой части навигационной панель находится кнопка поиска, которая позволяет производить поиск карточек по конкретным тегам и кнопка фильтрации, после нажатия на которую

появляется диалоговое окно фильтрации, продемонстрированная на рисунке 34.

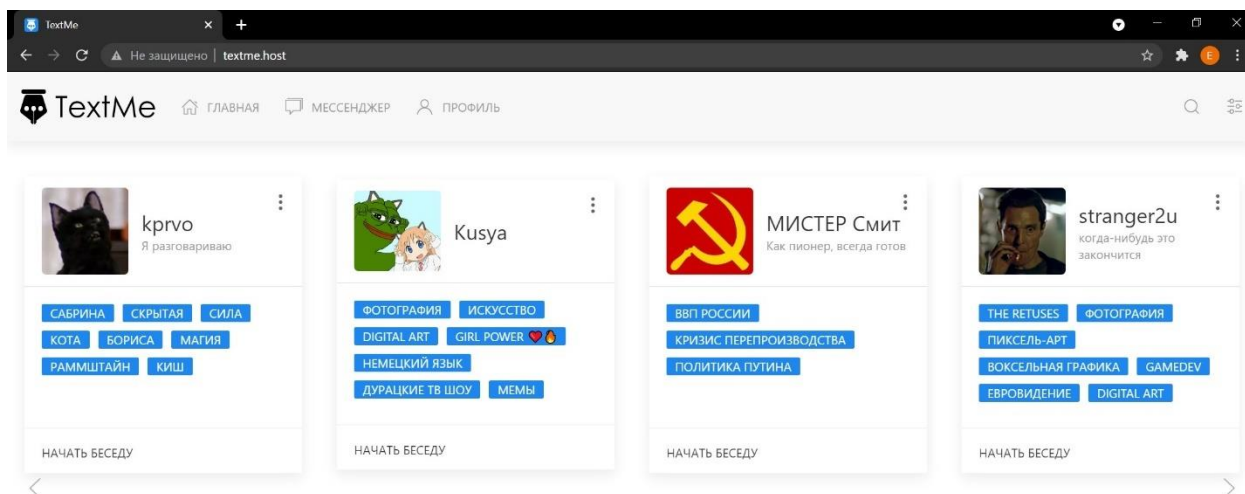


Рисунок 33 - Главная страница

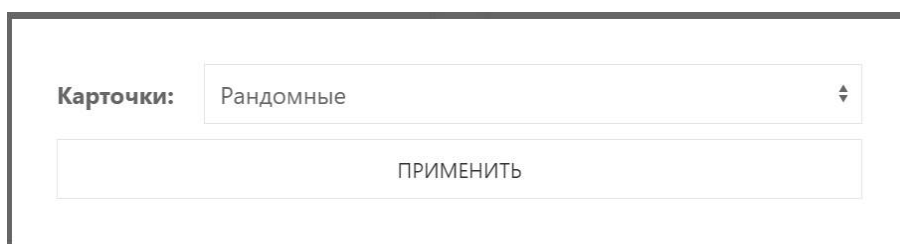


Рисунок 34 - Диалоговое окно фильтрации карточек

В выпадающем списке, пользователь может выбрать один из трех вариантов подбора карточек: «рандомные», «подобранные», «избранные».

С самого начала автоматически стоит режим «рандомные». Он означает, что карточки, которые будут отображаться на главной странице, подобраны случайным образом, а следовательно, при каждой перезагрузке они будут отличаться.

Выбрав режим «подобранные», на главной странице будут отображаться карточке, выбранные по максимальному совпадению тегов с карточкой пользователя.

Выбрав режим «избранное», пользователь сможет увидеть карточки, которые он заранее добавил в этот список.

После нажатия кнопки «применить», карточки на главной странице автоматически и без перезагрузки страницы меняются на те, которые выбрал пользователь.

На карточке в правом верхнем углу находится вертикальное троеточие, нажав на которую, пользователь увидит контекстное меню карточки (рисунок 35).

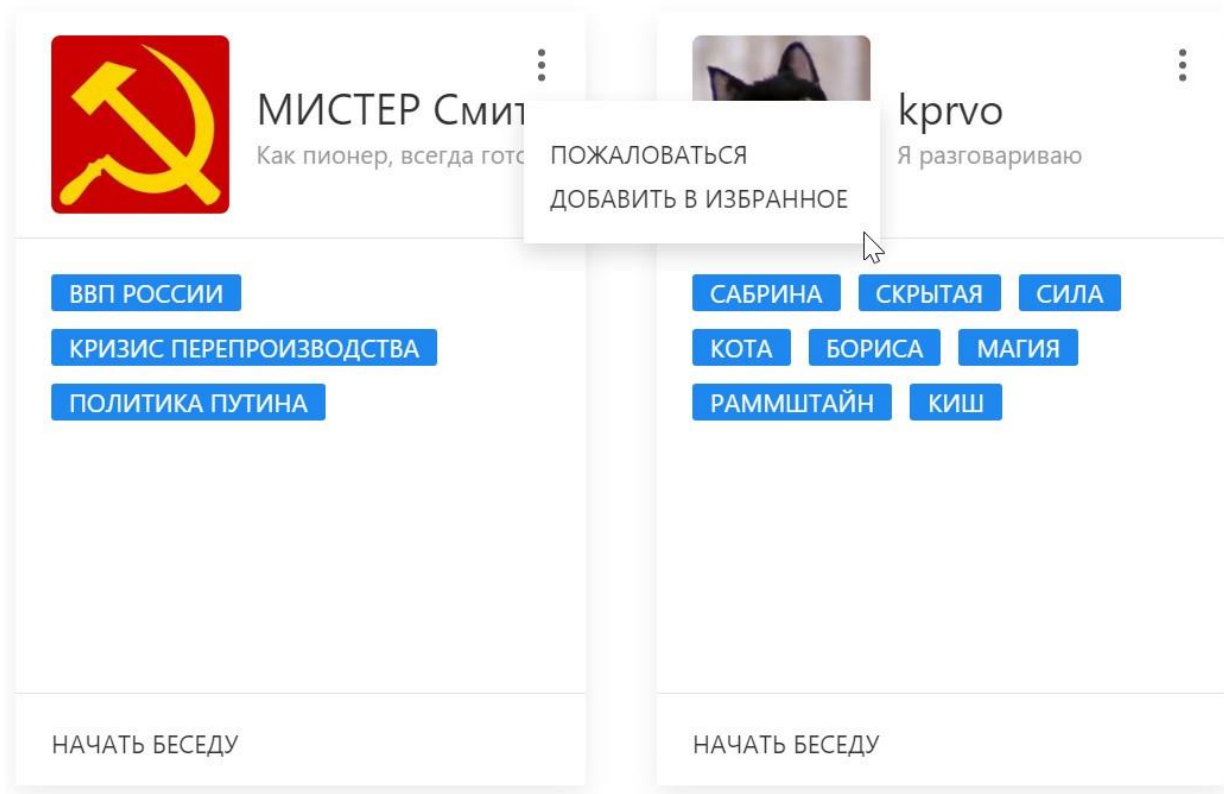
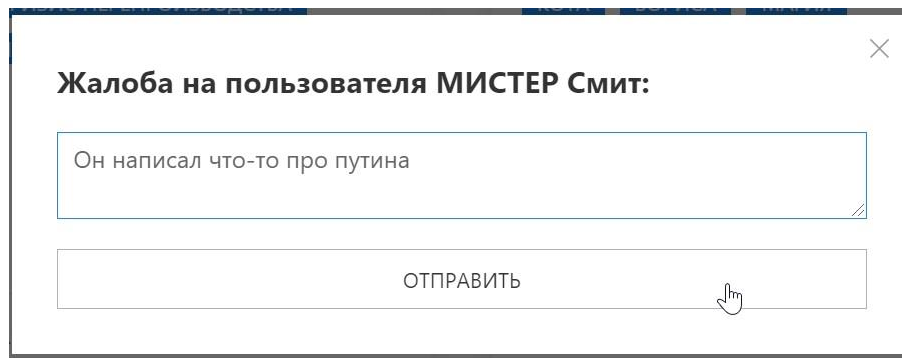


Рисунок 35 - Контекстное меню карточки

В данном меню находится две кнопки. Пользователь может нажать на кнопку «добавить в избранное», после чего выбранная им карточка окажется в списке избранных. А также, пользователь может нажать на кнопку «пожаловаться» и увидеть диалоговое окно написание жалобы (рисунок 36). После формирования жалобы, пользователь, нажав на кнопку «отправить», видит сообщение об успешной отправки жалобы (рисунок 37).

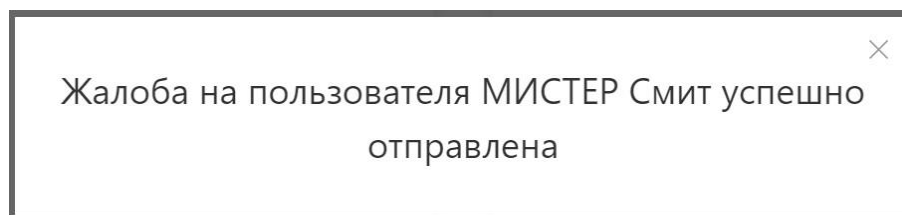


Жалоба на пользователя МИСТЕР Смит:

Он написал что-то про путина

ОТПРАВИТЬ

Рисунок 36 - Формирование жалобы



Жалоба на пользователя МИСТЕР Смит успешно отправлена

Рисунок 37 - Сообщение об успешной отправке жалобы

### 3.3.4 Чат

Внизу карточки находится кнопка «начать беседу». Нажав на нее, пользователь сразу оказывается в чате с выбранным пользователем (рисунок 38). Здесь он может отправить сообщение, прочитать сообщение, изменить или удалить свои ранее отправленные сообщения. Сообщения, которые были отправлены, но не прочитаны, отображаются с синим задним фоном. После того, как собеседник их прочитает, они отображаются в стандартном черно-белом варианте.

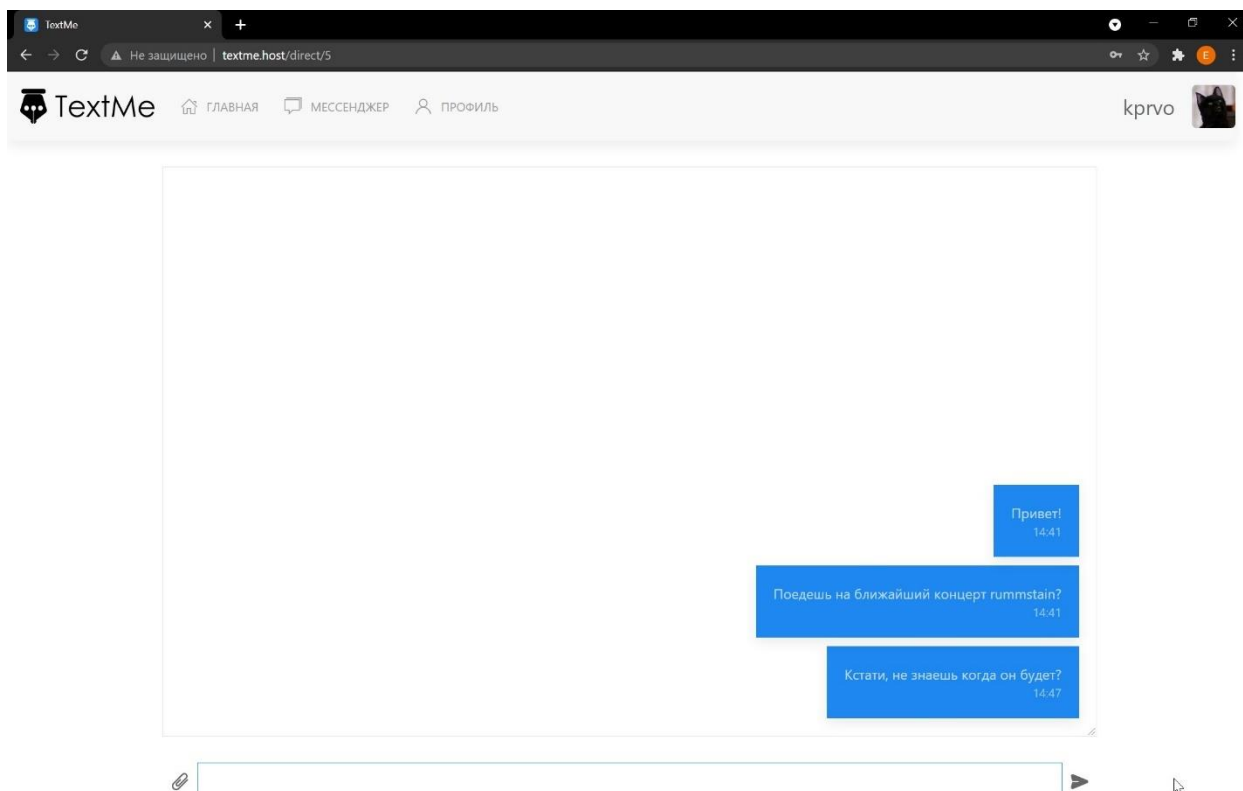


Рисунок 38 - Чат с выбранным пользователем

Если нажать на любое из своих сообщений, то у пользователя появляется диалоговое окно управления сообщения (рисунок 39), в котором он может нажать на кнопку «удалить», и тем самым удалить сообщение, или нажать на кнопку «изменить», после чего появится дополнительное поле для редактирования сообщения (рисунок 40). После изменения текста сообщения и нажатие на кнопку «применить изменения», сообщение в чате изменяется, а дата отправки заменяется датой редактирования, к которой приписывается слово «ред.». Измененное сообщение продемонстрировано на рисунке 41.

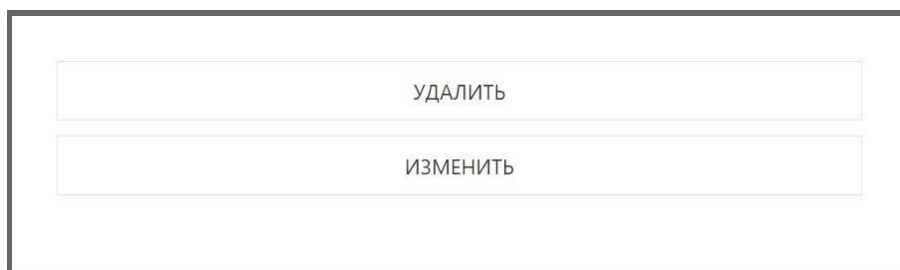
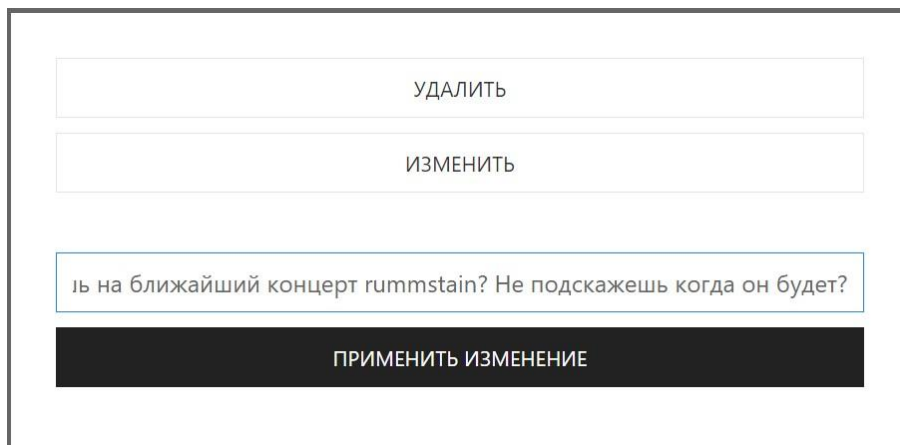


Рисунок 39 - Диалоговое окно удаления и изменения сообщения



УДАЛИТЬ

ИЗМЕНИТЬ

Поедешь на ближайший концерт rummstain? Не подскажешь когда он будет?

ПРИМЕНИТЬ ИЗМЕНЕНИЕ

Рисунок 40 - Форма изменения выбранного сообщения

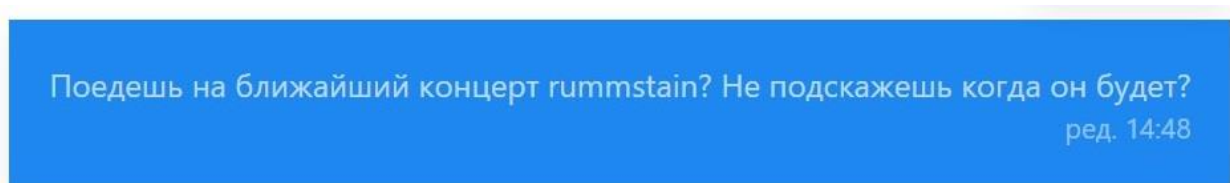


Рисунок 41 - Вид сообщения, после его редактирования

На рисунке 42 продемонстрирован вид чата, в котором есть прочитанные сообщения от пользователя и его собеседника.

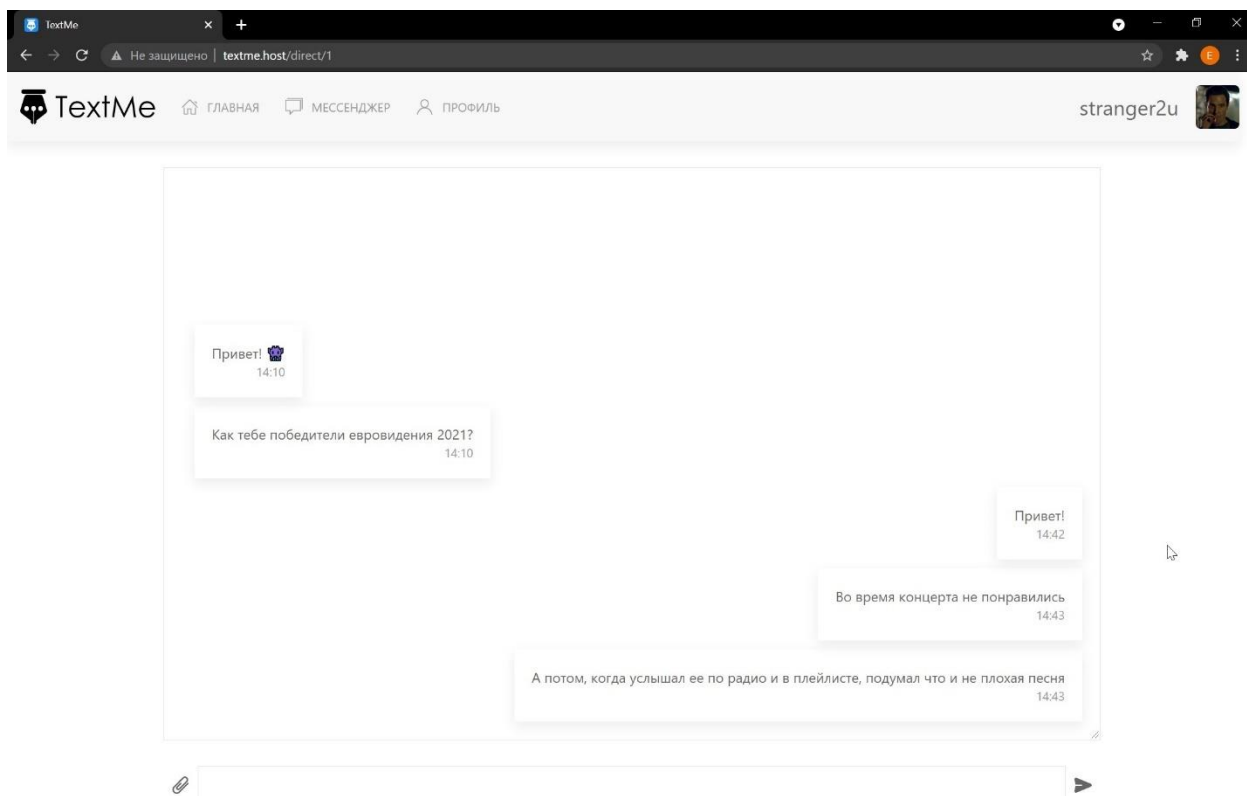


Рисунок 42 - Чат с прочитанными сообщениями



### 3.3.5 Мессенджер. Список чатов

Нажав на навигационной панели кнопку «мессенджер», пользователь попадает на страницу, на которой он может увидеть таблицу с активными переписками (рисунок 43). В данной таблице отображается имя собеседника, фото его профиля, последнее сообщения и дата отправки этого сообщения. Если сообщения отправлено от рассматриваемого пользователя, а не от собеседника, то в начале сообщения приписывается слово «Вы:».

Таблица фильтруется по времени отправки сообщения, то есть в самом начале показываются самые свежие. Если сообщение не прочитано, то данная строка в таблице подкрашивается в серый цвет.

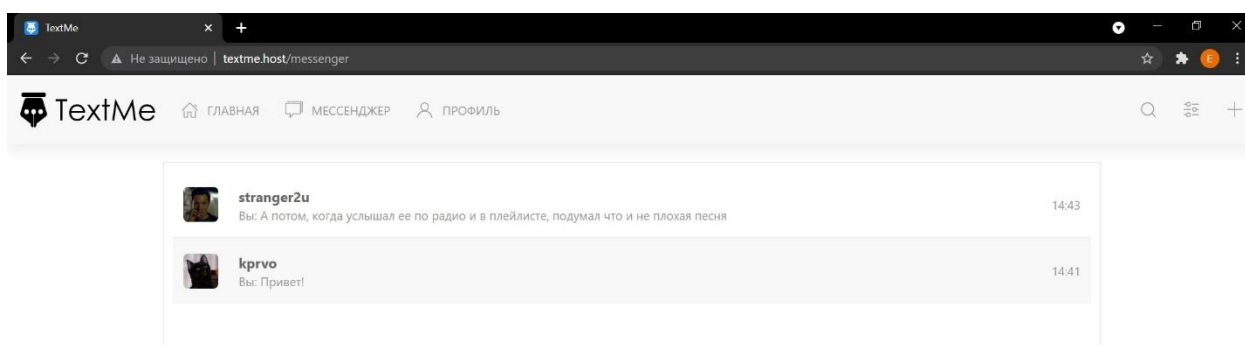


Рисунок 43 - Список чатов

В правом углу находится иконка настроек, нажав на которую появляется диалоговое окно фильтрации чатов (рисунок 44). Здесь, в выпадающих списках, пользователь может выбрать тип отображаемых чатов: все, личные переписки, групповые беседы. А также, здесь он может выбрать тип сообщений, которые будут отображаться в таблице, а именно: прочитанные, не прочитанные, все.

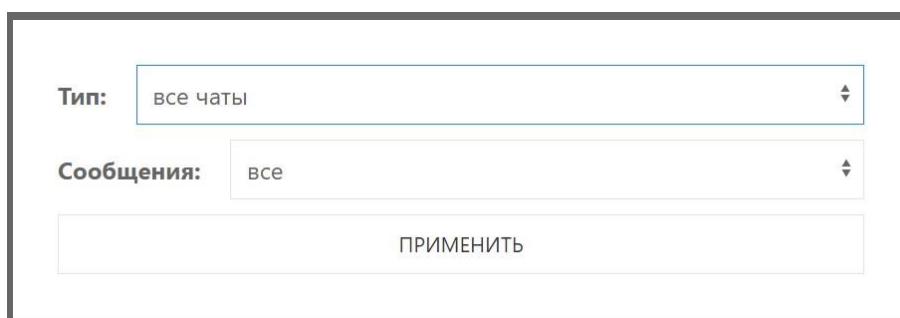


Рисунок 44 - Фильтр чатов

### 3.3.6 Профиль пользователя

Нажав на кнопку «профиль», которая находится на навигационной панели, пользователь попадает на страницу с своим профилем, продемонстрированную на рисунке 45.

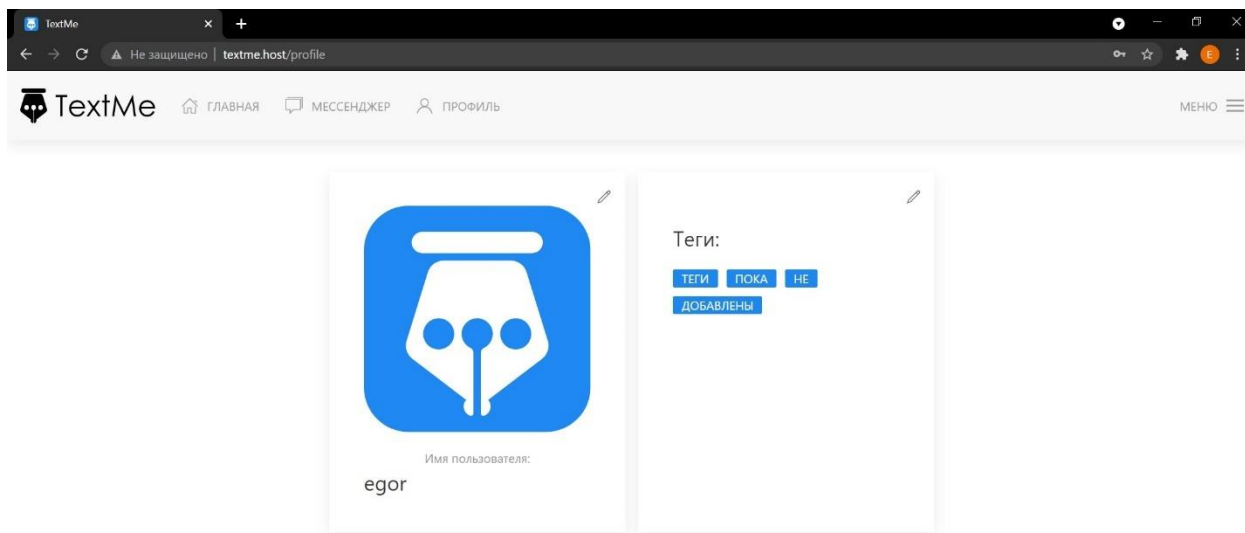


Рисунок 45 - Профиль пользователя, сразу после регистрации

После регистрации профиль пользователя является незаполненным. Вместо фото профиля отображается стандартный логотип приложения, а в место тегов отображается стандартная надпись, разделенная на 4 тега «теги пока не добавлены».

Пользователю необходимо заполнить свой профиль. Он может это сделать, нажав на иконку карандаша, которая находится в правом верхнем углу карточки, с основной информацией пользователя. После нажатия на данную иконку, появляется диалоговое окно редактирования профиля, продемонстрированное на рисунке 46.

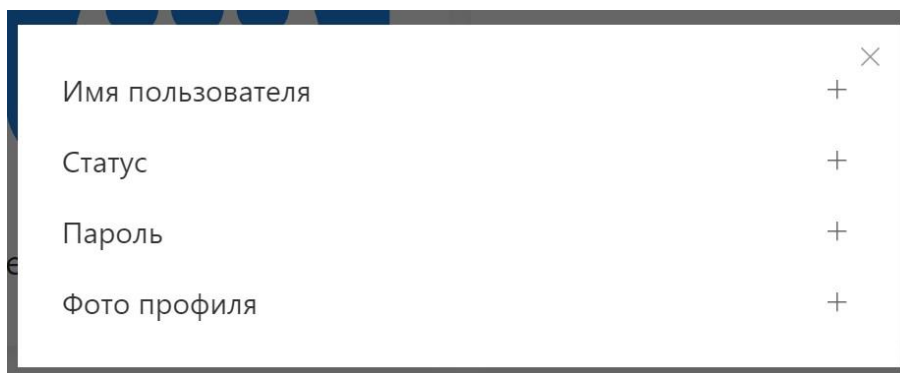


Рисунок 46 - Диалоговое окно редактирования профиля

Здесь пользователь может изменить свое имя, статус, пароль и фото профиля. Эти формы и действия продемонстрированы на рисунках 47-50.

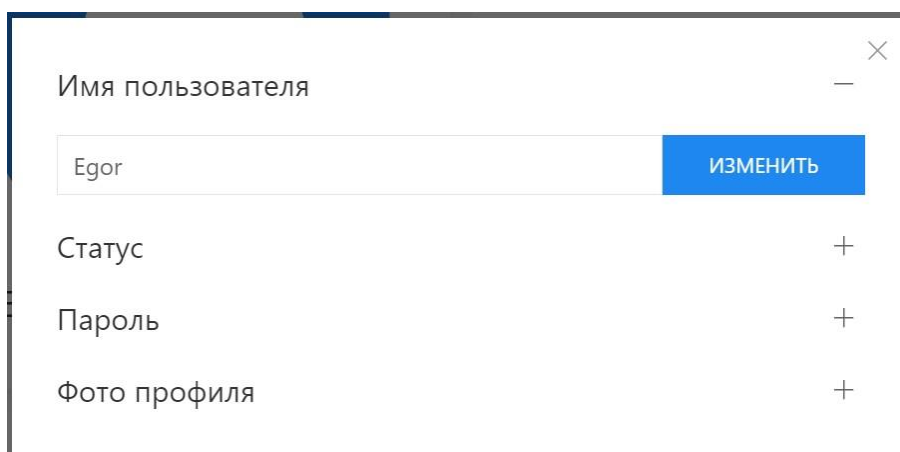


Рисунок 47 - Изменение имени пользователя

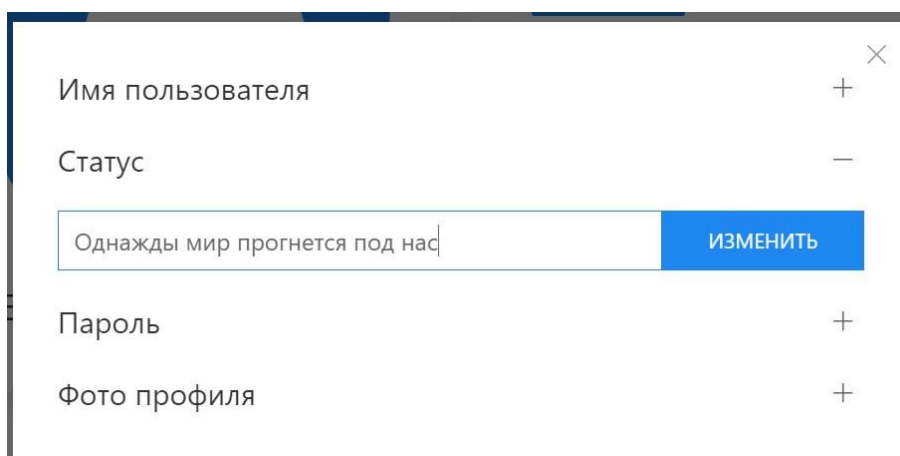


Рисунок 48 - Изменение статуса пользователя

A screenshot of a web form for updating a user profile. The form is enclosed in a dark border with a close button (X) in the top right corner. It contains several input fields and a button. The labels and their corresponding elements are: 'Имя пользователя' (Username) with a '+' icon; 'Статус' (Status) with a '+' icon; 'Пароль' (Password) with a '-' icon; 'Старый пароль' (Old password) with a text input field; 'Новый пароль' (New password) with a text input field; 'Повторите новый пароль' (Repeat new password) with a text input field; a wide button labeled 'ОБНОВИТЬ ПАРОЛЬ' (Update password); and 'Фото профиля' (Profile photo) with a '+' icon.

Рисунок 49 - Форма изменения пароля

A screenshot of a web form for updating a user profile, specifically for changing the profile photo. The form is enclosed in a dark border with a close button (X) in the top right corner. It contains labels for 'Имя пользователя' (Username), 'Статус' (Status), 'Пароль' (Password), and 'Фото профиля' (Profile photo), each with a '+' or '-' icon. Below these is a large light gray button labeled 'Выберите файл' (Select file). At the bottom are two buttons: 'ЗАГРУЗИТЬ' (Upload) on the left and 'ЗАКРЫТЬ' (Close) on the right.

Рисунок 50 - Форма изменения фото профиля

При изменении фото профиля, пользователь должен выбрать новое фото. После выбора файла на устройстве, новое фото отобразится в данной форме для предпросмотра и подтверждения о том, что именно это фото пользователь хотел выбрать (рисунок 51).

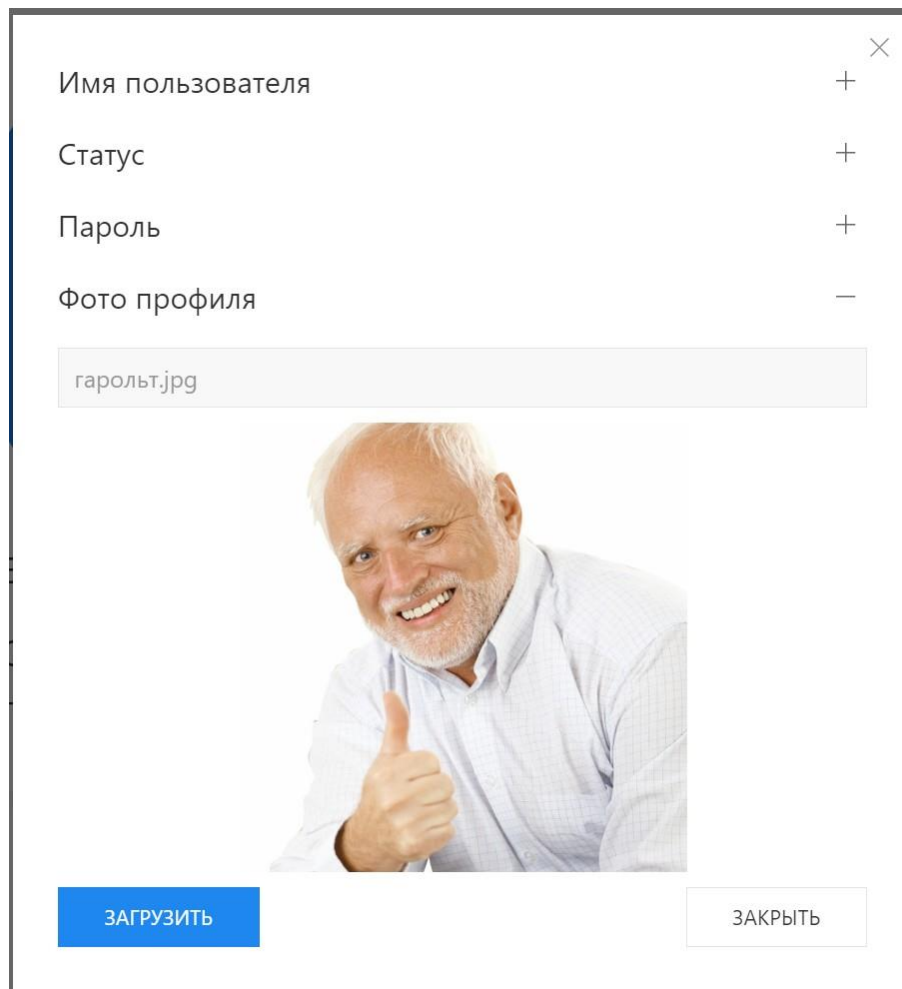


Рисунок 51 - Предпросмотр выбранного фото

Если пользователя все устраивает, то после нажатия кнопки «загрузить» и успешной загрузки фото в хранилища, он увидит подтверждающее сообщение об успешной загрузки. Фото профиля изменено.

На карточке, на которой отображаются теги, тоже есть иконка карандаша. Нажав на нее, пользователь увидит диалоговое окно управления тегами. Здесь пользователь сможет добавить новый тег или удалить имеющийся.



Рисунок 52 - Диалоговое окно управления тегами

При выборе режима добавления, появляется текстовое поле и пустая панелька. При вводе в текстовое поле тега, в панельке будут появляться все теги, которые имеют совпадение с тем, что вводит пользователь. Пользователь может нажать на один из предложенных тегов, и тем самым добавить его к себе. На рисунке 53 продемонстрирован случай, когда пользователь ввел букву «и», увидел все теги, которые на нее начинаются, и решил добавить тег «история», нажав на него.

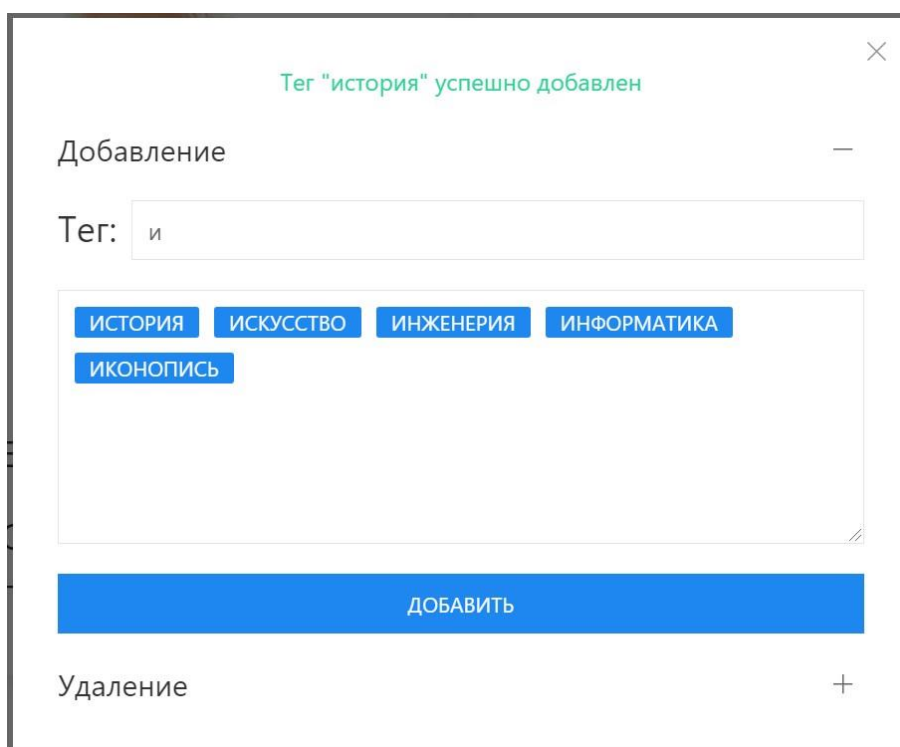


Рисунок 53 - Добавление тега из предложенных

Если на панельки не окажется тегов, значит такого тега раньше не было. Пользователь может ввести любое слово и нажать на кнопку «добавить». После этого этот тег добавиться в базу данных и присоединиться к пользователю. В дальнейшем, этот тег будет появляться при поиске. На рисунке 54, продемонстрирован случай, когда в базе данных не нашелся тег, совпадающий с словом «автоваз».

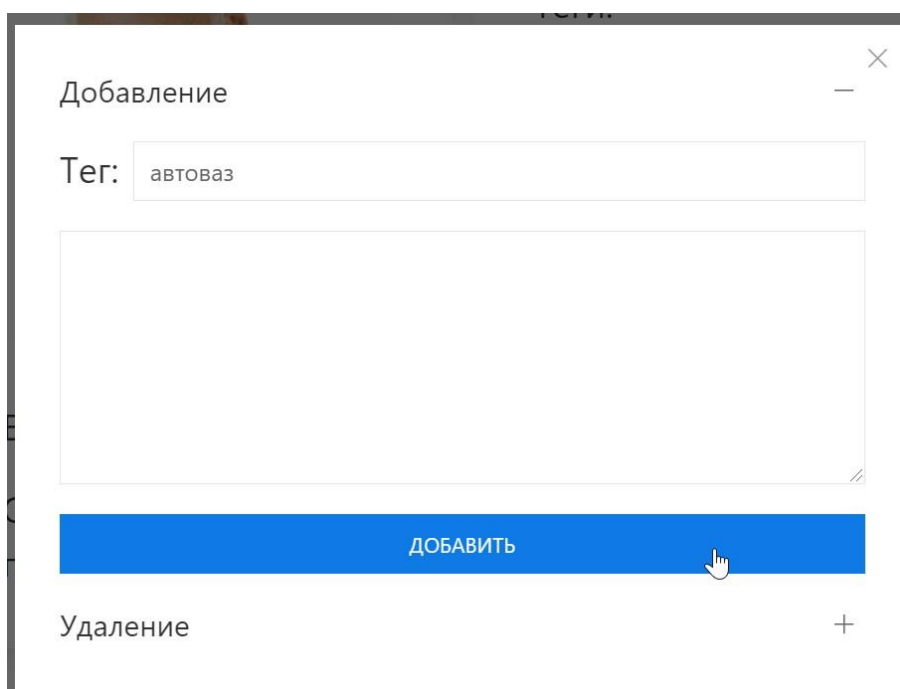


Рисунок 54 - Добавления тега, которого раньше не было

Выбрав режим удаления тегов, появляется панель, в которой отображаются все теги пользователя, но тут они окрашены в красный цвет (рисунок 55). Пользователь может выбрать тег, который хочет удалить, нажат на него, и он удалится из списка тегов пользователя.

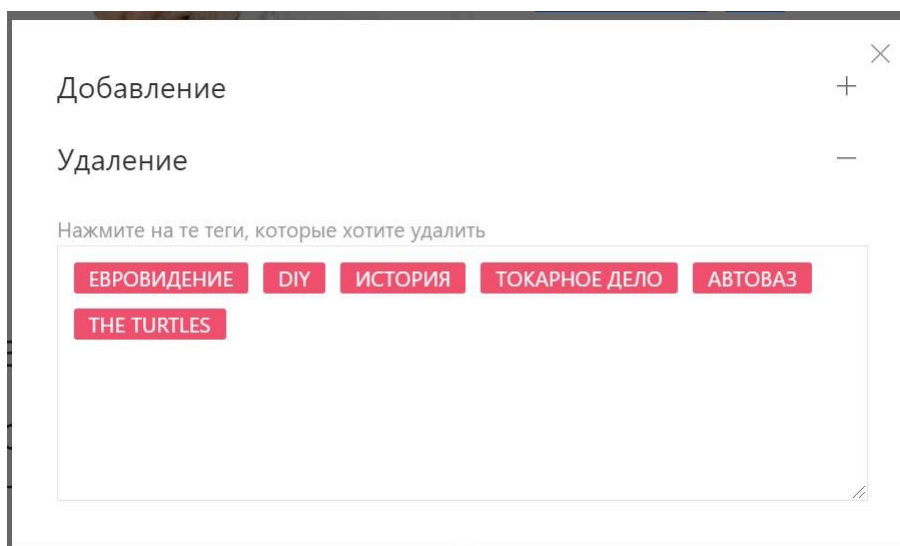


Рисунок 55 - Режим удаление тегов

На рисунке 56 продемонстрирован отредактированный профиль пользователя, после применения всех вышеупомянутых изменений.

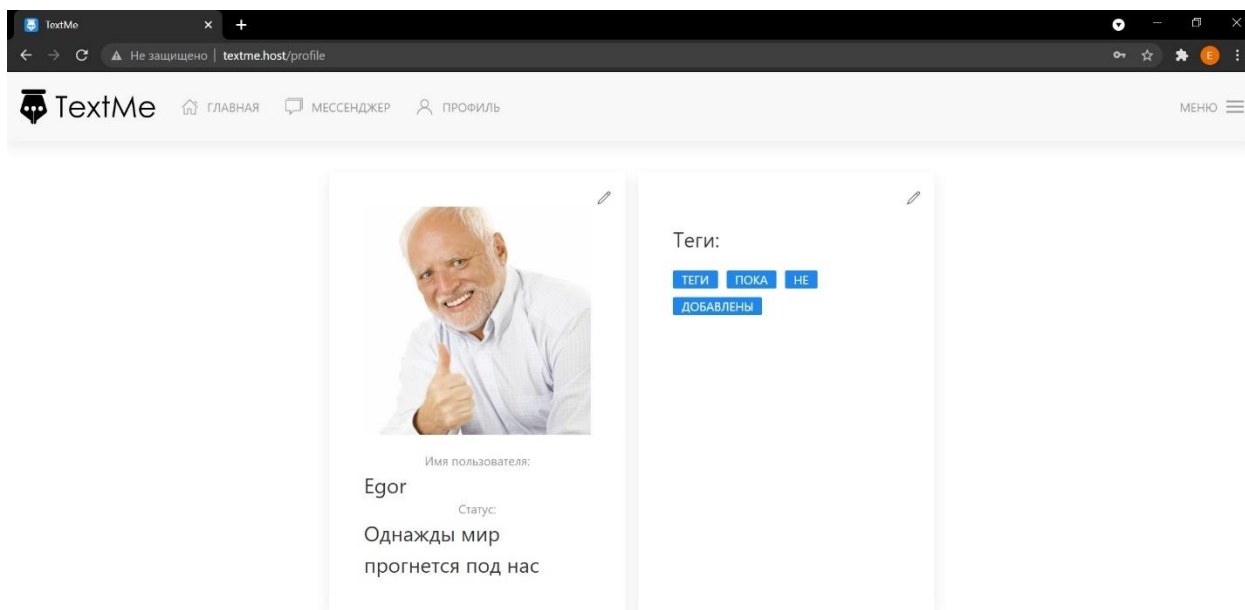


Рисунок 56 - Профиль после редактирования

### 3.3.7 Администрация

На странице профиля в правом верхнем углу находится кнопка «меню», нажав на которую пользователь может увидеть дополнительное навигационное меню, продемонстрированное на рисунке 57.

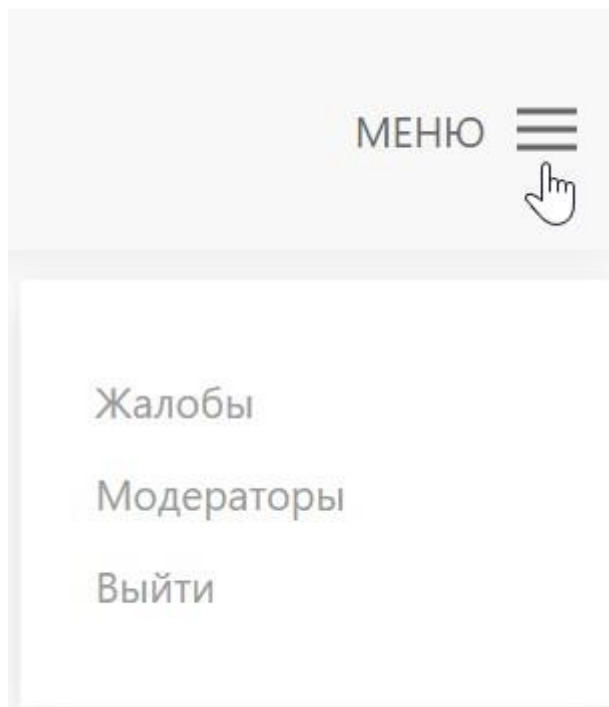


Рисунок 57 - Дополнительное навигационное меню



Если пользователь обладает правами администратора, то в этом меню он увидит ссылки на страницу с жалобами, на страницу управления модераторами и кнопку «выход», которая предназначена для выхода из аккаунта.

Если пользователь обладает правами модератора, то в том меню у него будет отсутствовать кнопка «Модераторы», которая ведет на страницу управления модераторами.

Если пользователь не обладает дополнительными правами, то у него не будет дополнительного меню, а в этом месте будет находиться кнопка «Выход».

Нажав на кнопку «жалобы», администратор или модератор попадает на страницу с жалобами, продемонстрированную на рисунке 58.

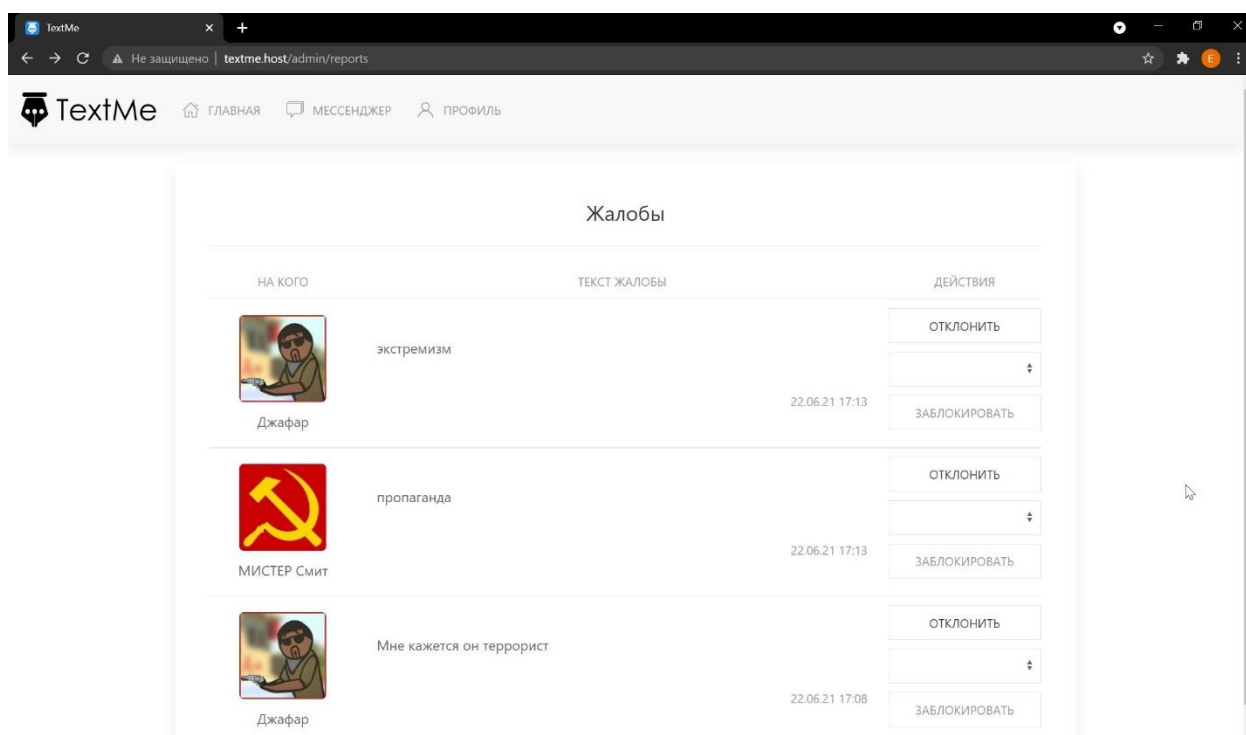


Рисунок 58 - Страница с жалобами

На данной странице в виде таблицы отображается следующая информация: фото профиля и имя пользователя, на которого пожаловались, текст жалобы и дата отправки жалобы, а также кнопки для быстрого решения жалобы.

Нажав на кнопку «отклонить», жалоба помечается как решенная и больше не появляется в этом списке.

Для того, чтобы заблокировать пользователя, необходимо в выпадающем списке (рисунок 59) выбрать одно из предложенных количеств дней блокировки: 1, 3, 7, 14, 30, 365 дней.

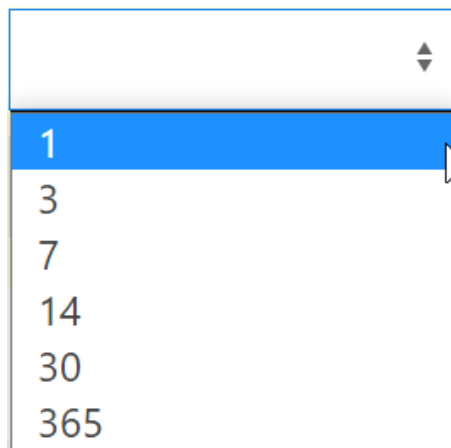


Рисунок 59 - Выбор продолжительности блокировки

После выбора продолжительности блокировки, кнопка «заблокировать» становится активной и после этого пользователь попадает в список заблокированных пользователей.

Если нажать на строку таблицы, то открывается страница с сводкой жалоб по одному выбранному пользователю. На рисунке 60 продемонстрирована сводка жалоб на пользователя «Джафар».

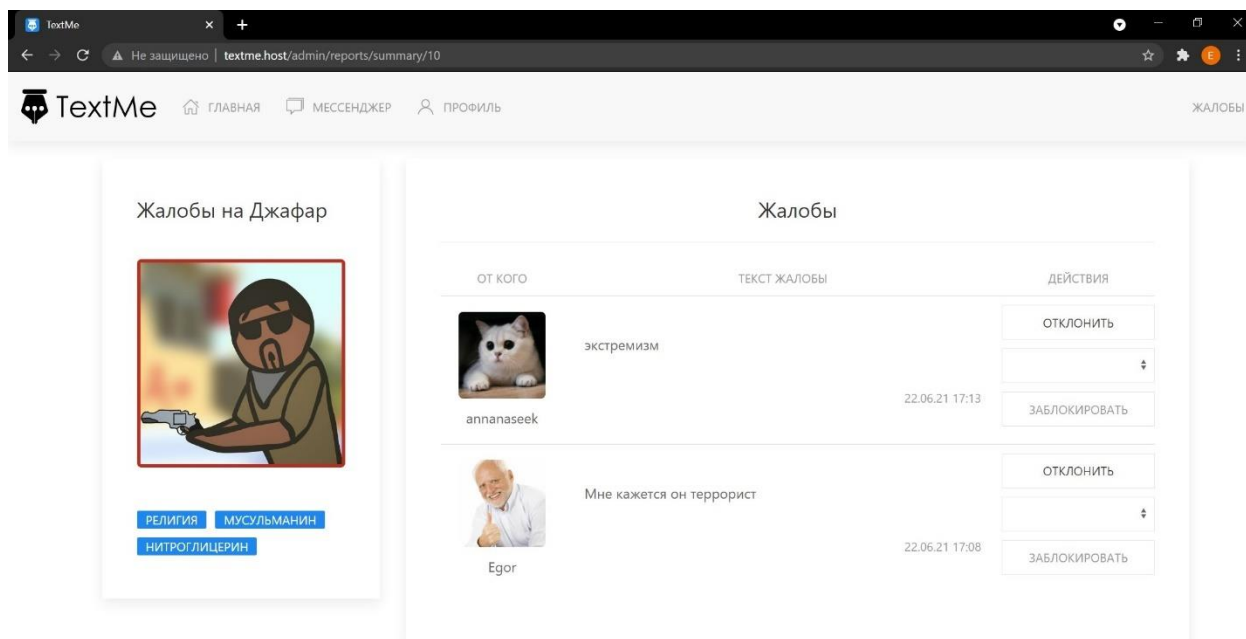


Рисунок 60 - Сводка жалоб по одному пользователю

В левой части этой страницы отображается карточка пользователя, на которого поступили жалобы. В правой части отображается таблица, схожая с таблицей на прошлой странице, но она отличается от предыдущей тем, что в левом столбце написано от кого поступила жалоба. Здесь администратор или модератор может посмотреть сразу все жалобы на пользователя и принять решение что с ними делать: отклонить или принять.

После того, как пользователь был заблокирован, доступ к сайту у него временно прекращается, и при попытке пройти авторизацию заблокированный пользователь видит сообщение, продемонстрированное на рисунке 61.



Рисунок 61 - Информация о том, что профиль заблокирован

Нажав на кнопку «модераторы», администратор попадает на страницу с модераторами, продемонстрированную на рисунке 62.

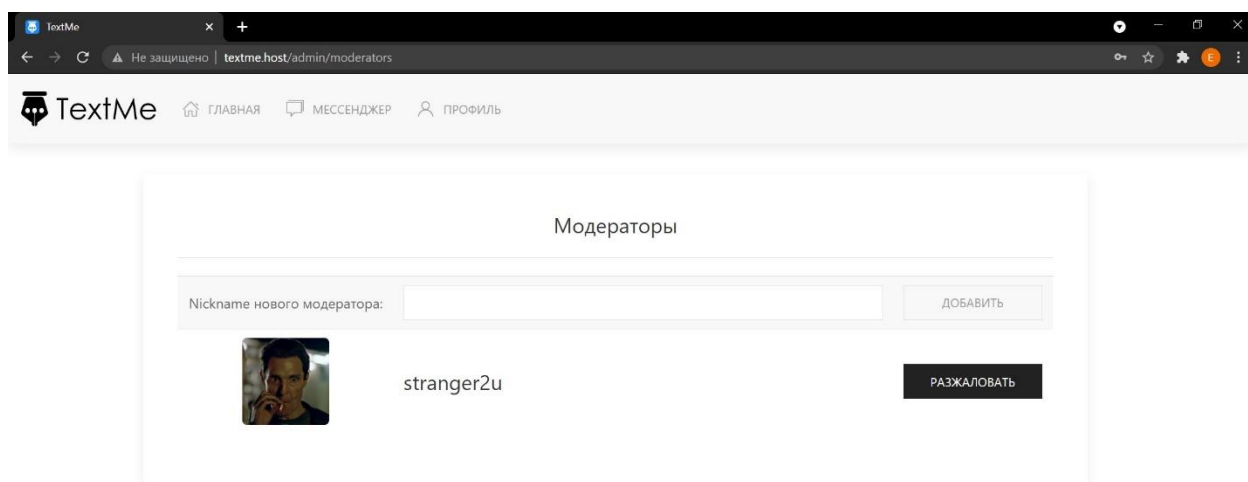


Рисунок 62 - Страница с модераторами

На данной странице в виде таблицы отображаются все пользователи с правами модераторами. Если у выбранного пользователя нажать на кнопку «разжаловать», то он перестает быть модератором.

Чтобы добавить нового модератора, необходимо написать его полное имя в текстовое поле вверху таблицы и после нажать кнопку «добавить». Если пользователь с таким именем есть, то ему добавятся права модератора и он отобразится в этой таблице. На рисунке 63 продемонстрировано успешное добавление нового модератора.

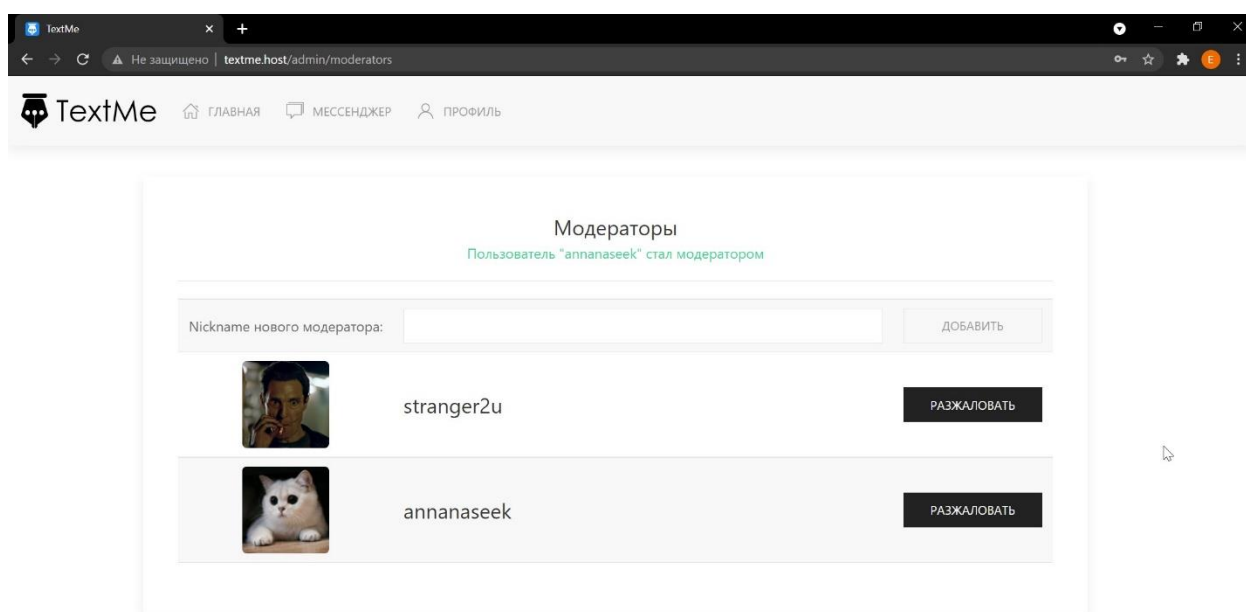


Рисунок 63 - Добавлен новый модератор

### 3.3.8 Отображение на мобильном устройстве

Интерфейс обладает свойством адаптивности и корректно отображается на мобильном устройстве. В качестве примера на рисунке 64 приведены скриншоты трех страниц, снятые с телефона.

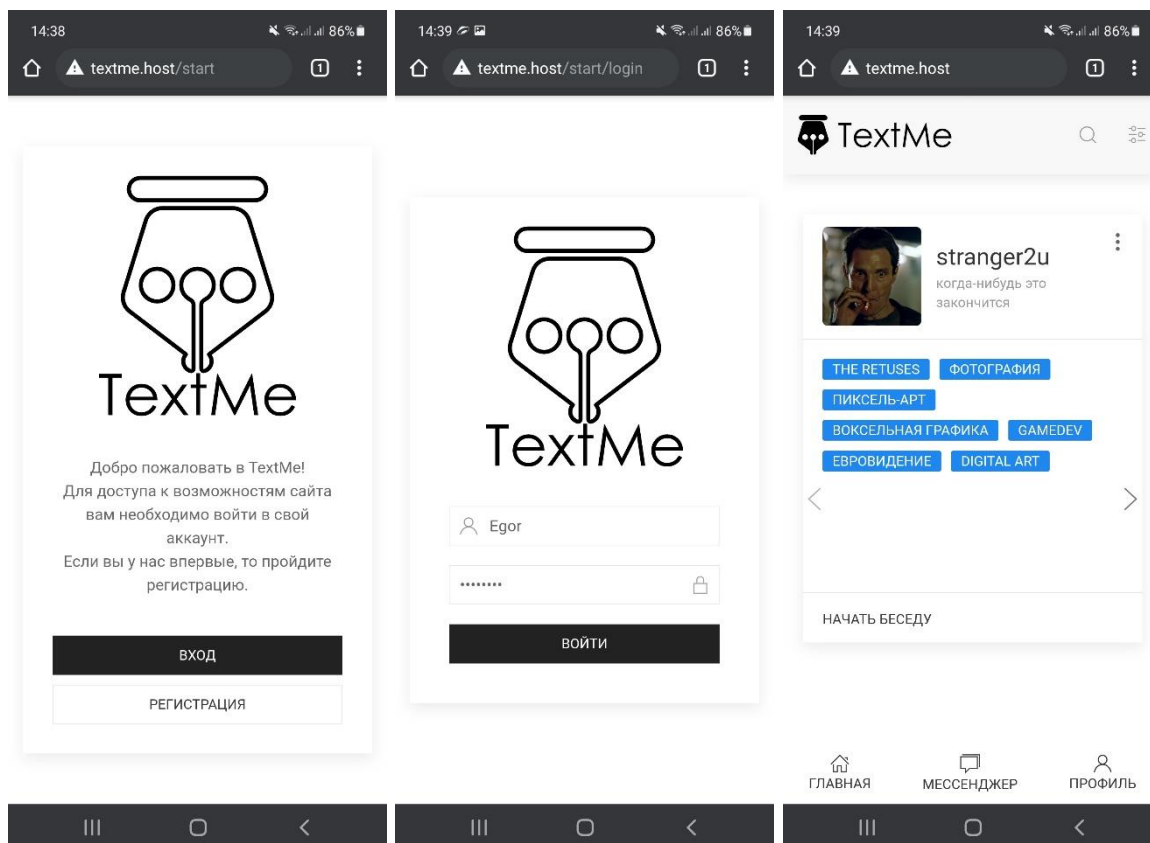


Рисунок 64 - Отображение на мобильном устройстве

## 4 Тестирование

### 4.1 Модульное тестирование

По ходу разработки приложения проводилось модульное тестирование java кода. Основные сценарии были направлены на проверку тяжелых динамических sql запросов к базе данных. Тестовые сценарии и их результаты приведены в таблице 1.

Таблица 1 – результаты модульного тестирования

Сценарий	Результат
Добавление тегов к карточке	Пройден
Удаление тегов у карточки	Пройден
Подбор карточек на основе тегов	Пройден
Поиск карточек по тегам	Пройден

### 4.2 Ручное функциональное тестирование

На последнем этапе разработки программного продукта было проведено ручное функциональное тестирование, нацеленное на проверку

работоспособности основной функциональности приложения. Тестовые случаи и их результаты приведены в таблице 2.

Таблица 2 – результаты ручного функционального тестирования

<b>Сценарий</b>	<b>Результат</b>
Регистрация пользователя	Пройден
Активация аккаунта через почту	Пройден
Вход в приложение	Пройден
Получение случайных карточек пользователей	Пройден
Получение подобранных карточек пользователей	Пройден
Добавление карточки в избранное	Пройден
Получение избранных карточек	Пройден
Поиск карточек по тегу	Пройден
Предложение совпадающих тегов во время поиска	Завален
Начало диалога с выбранным пользователем	Пройден
Отправка сообщения	Пройден
Чтение сообщения	Пройден
Редактирование сообщения	Пройден
Удаление сообщения	Пройден
Прикрепление файла к сообщению	Завален
Просмотр активных чатов	Пройден
Создание групповых чатов	Завален
Просмотр профиля пользователя	Пройден
Редактирование профиля пользователя	Пройден
Добавление тегов	Пройден
Удаление тегов	Пройден
Формирование жалобы	Пройден
Просмотр жалоб	Пройден
Отклонение жалобы	Пройден
Блокирование пользователя	Пройден

Продолжение таблицы 2

Просмотр модераторов	Пройден
Добавление модератора	Пройден
Удаление модератора	Пройден

### 4.3 Юзабилити-тестирование

Юзабилити-тестирование – это метод оценки интерфейса со стороны удобства и эффективности его использования. Чтобы получить результаты тестирования были привлечены четыре представителя целевой аудитории, а именно люди возрастом от 18 до 24 лет. Тестовые сценарии и их результаты продемонстрированы в таблице 3.

Таблица 3 – результаты юзабилити тестирования

	<b>Пользователь</b>			
<b>Сценарий</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>
Регистрация	Пройден	Пройден	Пройден	Пройден
Активация аккаунта	Пройден	Пройден	Пройден	Пройден
Вход в приложение	Пройден	Пройден	Пройден	Пройден
Заполнение профиля	Пройден	Пройден	Пройден	Пройден
Поиск собеседника	Пройден	Пройден	Пройден	Пройден
Ведение переписки	Пройден	Пройден	Пройден	Пройден
Формирование жалобы	Пройден	Пройден	Пройден	Пройден



## **Заключение**

В ходе выполнения курсового проекта командой был разработан мессенджер, соответствующий поставленным задачам.

Перед разработкой был проведен анализ предметной области, в результате которого были выявлены преимущества и недостатки найденных аналогов, который были учтены в ходе разработки данного мессенджера.

По ходу разработки проводилось тестирование с целью проверки работоспособности системы.

В процессе работы были выполнены следующие задачи:

- Обеспечение поиска собеседников на основе их карточек;
- Обеспечение возможности текстовой коммуникации;
- Обеспечение возможности подачи жалобы на пользователя;
- Обеспечение возможности создания необходимо подробной карточки для поиска собеседников;
- Обеспечение возможности для администратора и модератора просмотра списков всех поступающих жалоб;
- Обеспечение возможности назначения администратором модераторов;

## Список литературы

1. Документация фреймворка Angular [Электронный ресурс]. – Режим доступа: <https://angular.io/docs>
2. Документация фреймворка Spring Boot [Электронный ресурс]. – Режим доступа: <https://spring.io/projects/spring-boot>
3. Документация СУБД PostgreSQL [Электронный ресурс]. – Режим доступа: <https://www.postgresql.org/docs/>
4. Документация фреймворка MyBatis [Электронный ресурс]. – Режим доступа: <https://mybatis.org/mybatis-3/>
5. Документация системы управления версиями базы данных Liquibase [Электронный ресурс]. – Режим доступа: <https://docs.liquibase.com/>
6. Документация системы автоматической сборки Gradle [Электронный ресурс]. – Режим доступа: <https://docs.gradle.org/>
7. Документация CSS фреймворка UIKit [Электронный ресурс]. – Режим доступа: <https://3uikit.ru/>
8. Документация языка программирования TypeScript [Электронный ресурс]. – Режим доступа: <https://www.typescriptlang.org/docs/>
9. Документация jwt токенов [Электронный ресурс]. – Режим доступа: <https://jwt.io/introduction>
10. Документация протокола STOMP [Электронный ресурс]. – Режим доступа: <https://stomp.github.io/stomp-specification-1.2.html>
11. Документация сервиса Яндекс.Метрика [Электронный ресурс]. – Режим доступа: <https://yandex.com/support/metrika/>