

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

Факультет компьютерных наук

Кафедра технологий обработки и защиты информации

Разработка веб-приложения для ведения микроблогов

Курсовой проект

09.03.02 Информационные системы и технологии

Обработка информации и машинное обучение

Обучающийся _____ Д.А. Вакулин, 3 курс, д/о

Руководитель _____ В.С. Тарасов, ст. преподаватель

Воронеж 2022

Содержание

Содержание	2
Введение	3
1 Постановка задачи.....	4
2 Анализ предметной области	6
2.1 Терминология (гlossарий) предметной области.....	6
2.2 Обзор аналогов.....	8
2.2.1 Веб-приложение Twitter	8
2.2.2 Веб-приложение Tumblr	8
2.3 Графическое описание системы.....	9
2.3.1 Диаграмма прецедентов	9
2.3.2 Диаграмма последовательностей	10
2.3.3 Диаграмма состояний	11
2.3.4 Диаграмма развертывания.....	13
3 Реализация	15
3.1 Средства реализации.....	15
3.2 Архитектура приложения	15
3.3 Реализация БД.....	16
3.4 Реализация backend-составляющей приложения	17
Заключение	18
Список используемых источников.....	19

Введение

В настоящее время люди проводят значительную часть времени в социальных сетях. Они помогают не только в коммуникации с друзьями, коллегами, родственниками, но также позволяют публично высказывать свое мнение, узнавать новости и просто скоротать время.

В связи с обостренной ситуацией в мировой политике существует риск ограничения пока что доступных социальных сетей. Поэтому одной из задач является создание отечественного аналога сервиса микроблогов.

Все подобные приложения имеют, по большей части схожий функционал, поэтому на данный момент не существует доминирующей социальной сети или сервиса микроблогов. Каждый человек выбирает то приложение, которое ему подходит, или сразу несколько для различных ситуаций. Так, например, некоторые люди хотят иметь возможность закрыть доступ к чтению своих постов какой-либо группе: начальству; людям, несогласным с мнением автора или просто незнакомцам.

Данное приложение по ведению микроблогов будет иметь возможность ограничивать круг лиц, которым позволено читать посты.

Сказанное выше позволяет сделать вывод, что разработка настоящего веб-приложения имеет высокую актуальность по причине востребованности людей в этом.

1 Постановка задачи

Целью данного курсового проекта является разработка веб-приложения «MyBlog», которое предназначено для ведения разной степени приватности микроблогов, оставления комментариев и реакций.

К разрабатываемому приложению выдвинуты следующие функциональные требования:

- Возможность регистрации и авторизации пользователя в веб-приложении;
- Возможность просмотра ленты постов;
- Возможность просмотра комментариев к постам;
- Возможность авторизованного пользователя оставить комментарий;
- Возможность авторизованного пользователя поставить лайк посту;
- Возможность авторизованного пользователя написать свой пост с ограничением на круг лиц, позволенному читать пост;
- Возможность авторизованного пользователя просмотреть личный профиль;
- Возможность авторизованного пользователя сменить пароль и никнейм;
- Возможность авторизованного пользователя просмотреть профиль другого пользователя;
- Возможность авторизованного пользователя подписаться на другого пользователя;
- Возможность авторизованного пользователя открыть профиль другого пользователя по никнейму;
- Возможность администратора удалять посты;
- Возможность администратора удалять комментарии;

- Возможность администратора открыть профиль другого пользователя по никнейму;
- Возможность администратора блокировать пользователей;
- Возможность администратора удалять пользователей.

Для реализации данных требований необходимо выполнить следующие задачи:

- Проектирование веб-приложения.
- Разработка серверной части приложения:
 - 1) Проектирование и реализация базы данных;
 - 2) Реализация ролей в системе;
 - 3) Реализация системы авторизации и аутентификации пользователей.
 - 4) Реализация бизнес-логики;
- Разработка пользовательского интерфейса:
 - 1) Определение дизайна;
 - 2) Реализация пользовательского интерфейса в соответствии с установленным дизайном и функциональными требованиями;
 - 3) Реализация взаимодействия с серверной частью приложения.
- Тестирование веб-приложения.

2 Анализ предметной области

2.1 Терминология (гlossарий) предметной области

Авторизация – предоставление определенному лицу или группе лиц прав на выполнение определенных действий.

Администратор – специалист, уполномоченный управлять и администрировать определенный участок ИТ-среды.

Аккаунт – персональная страница пользователя или личный кабинет, который создается после регистрации на сайте.

Аутентификация – процедура проверки подлинности, например, проверка подлинности пользователя путем сравнения введенного им пароля с паролем, сохраненным в базе данных.

База данных – упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе.

Веб-приложение – клиент-серверное приложение, в котором клиент взаимодействует с веб-сервисом при помощи браузера.

Лайк – кнопка, с помощью которой публикация в приложении отмечается как понравившаяся, а также сама отметка.

Никнейм – псевдоним, используемый пользователем в сервисе.

Пользователь – лицо, которое использует приложение для выполнения конкретной функции.

Пост – информационный блок, размещенный пользователем в социальной сети или блоге.

Сервер – выделенный или специализированный компьютер для выполнения сервисного программного обеспечения.

СУБД – система управления базами данных. Комплекс программного-языковых средств, позволяющих создать базы данных и управлять данными.

Хостинг – услуга по предоставлению ресурсов для размещения информации на сервере, постоянно имеющем доступ к сети (обычно Интернет).

Яндекс-метрика – бесплатный интернет-сервис компании Яндекс, предназначенный для оценки посещаемости веб-сайтов и анализа поведения пользователей.

API – программный интерфейс приложения. Описание способов, которыми одна компьютерная программа может взаимодействовать с другой программой.

Bootstrap – свободный набор инструментов для создания сайтов и веб-приложений, включающий в себя HTML- и CSS-шаблоны

CSS – формальный язык описания внешнего документа, написанного с использованием языка разметки.

GitHub – крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки.

HTML – стандартизированный язык разметки документов для просмотра веб-страниц в браузере.

Java – строго-типизированный объектно-ориентированный язык программирования общего назначения.

JavaScript – мультипарадигменный язык программирования, поддерживающий объектно-ориентированный, императивный и функциональные стили.

Miro – интерактивная доска для совместной работы распределенных команд.

PostgreSQL – свободная объектно-реляционная система управления базами данных.

Swagger – язык описания интерфейса для описания API.

Trello – облачная программа для управления проектами небольших групп.

2.2 Обзор аналогов

2.2.1 Веб-приложение Twitter

Twitter – американский сервис микроблогов и социальная сеть, в которой пользователи публикуют сообщения, известные как «твиты», и взаимодействуют с ними.

Пользователи взаимодействуют с Twitter через веб-браузер, мобильное приложение или API.

Преимущества:

- Бесплатная социальная сеть;
- Огромная аудитория;
- Простой интерфейс.

Недостатки:

- В данный момент заблокирован на территории РФ;
- Несмотря на то, что Twitter имеет политику против спама, многие учетные записи продолжают эту практику;
- Ограничение поста в 280 символов;
- Большое количество неактивных и заброшенных аккаунтов.

2.2.2 Веб-приложение Tumblr

Tumblr является службой микроблогов, включающей в себя множество картинок, статей, видео и gif-изображений по разным тематикам.

Пользователь может подписываться на блоги других пользователей, после чего их записи будут появляться в его ленте новостей.

Достоинства:

- Кастомизация интерфейса;
- Высокое качество картинок из-за низкого уровня сжатия;
- Возможность ограничивать круг лиц, имеющих доступ к посту.

Недостатки:

- Отсутствие русскоязычной службы поддержки;
- Наличие большого количества спама;
- Сложность восприятия интерфейса.

2.3 Графическое описание системы

2.3.1 Диаграмма прецедентов

На Рисунке 1 изображена диаграмма прецедентов показывает сценарии, которые может выполнить каждый вид пользователей.

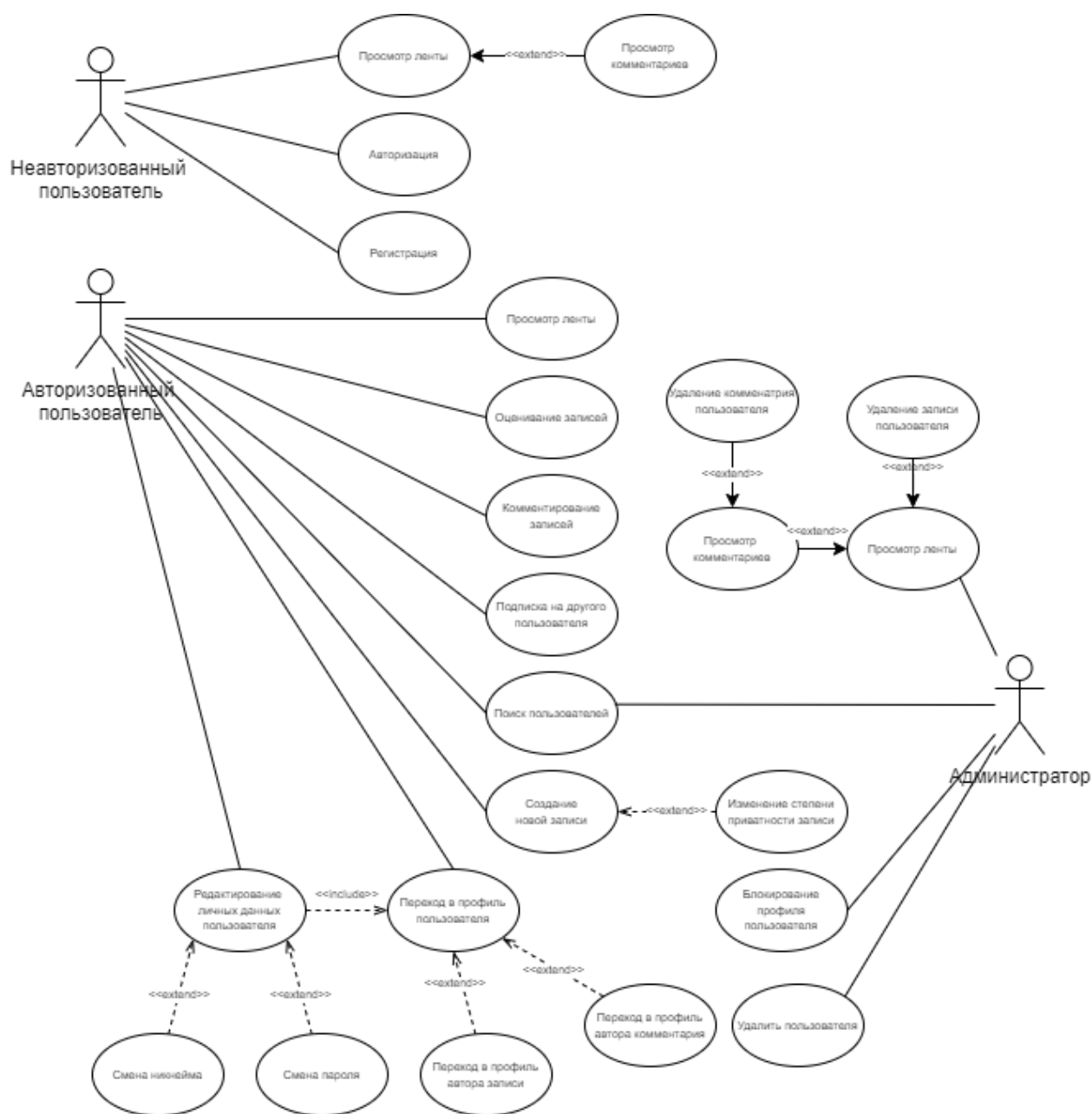


Рисунок 1 - Диаграмма прецедентов

2.3.2 Диаграмма последовательностей

На Рисунке 2 изображена диаграмма последовательностей регистрации пользователя.

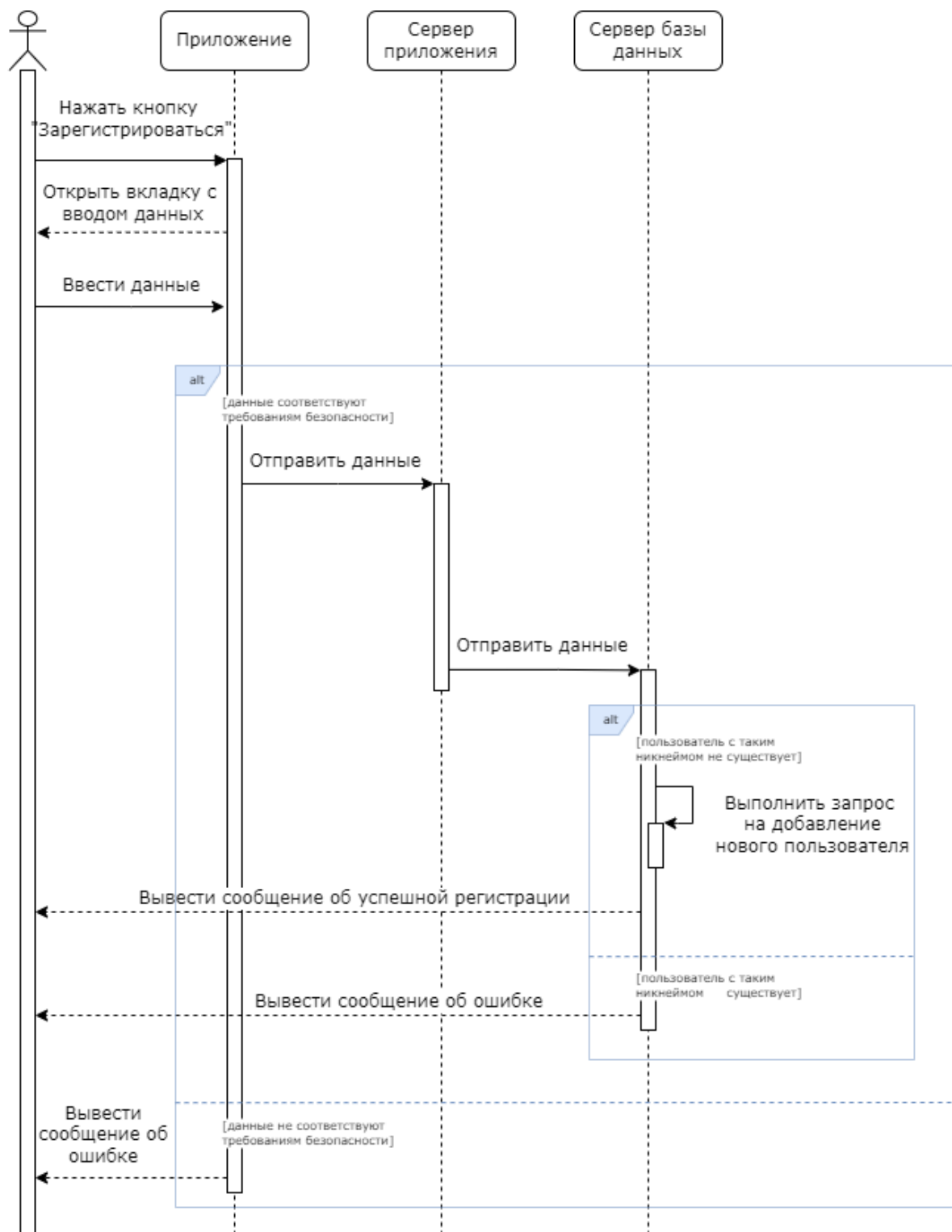


Рисунок 2 - Диаграмма последовательностей при регистрации

На Рисунке 3 изображена диаграмма последовательностей при авторизации пользователя.

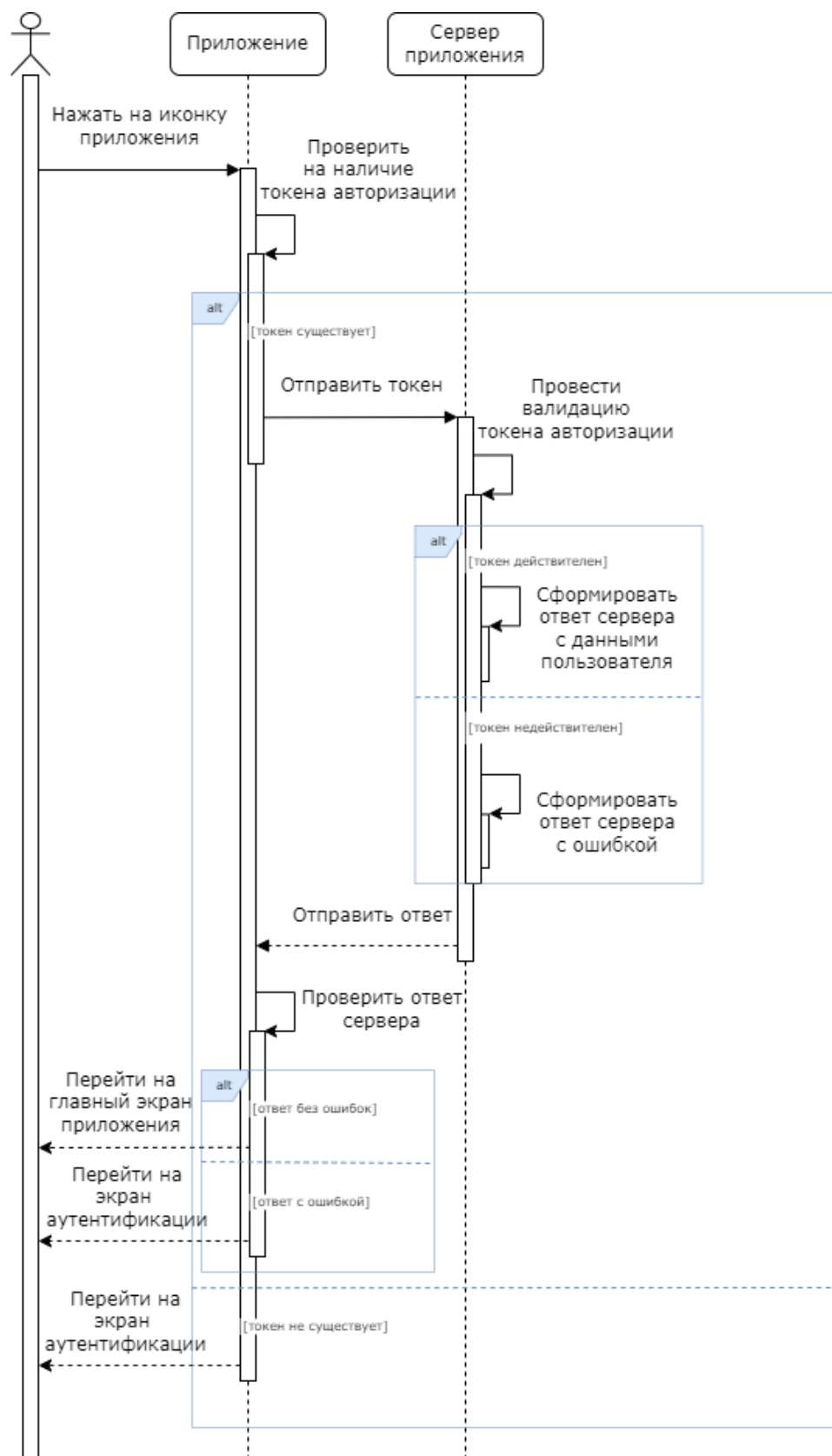


Рисунок 3 - Диаграмма последовательностей при авторизации

2.3.3 Диаграмма состояний

На Рисунке 4 изображена диаграмма состояний при авторизации пользователя.

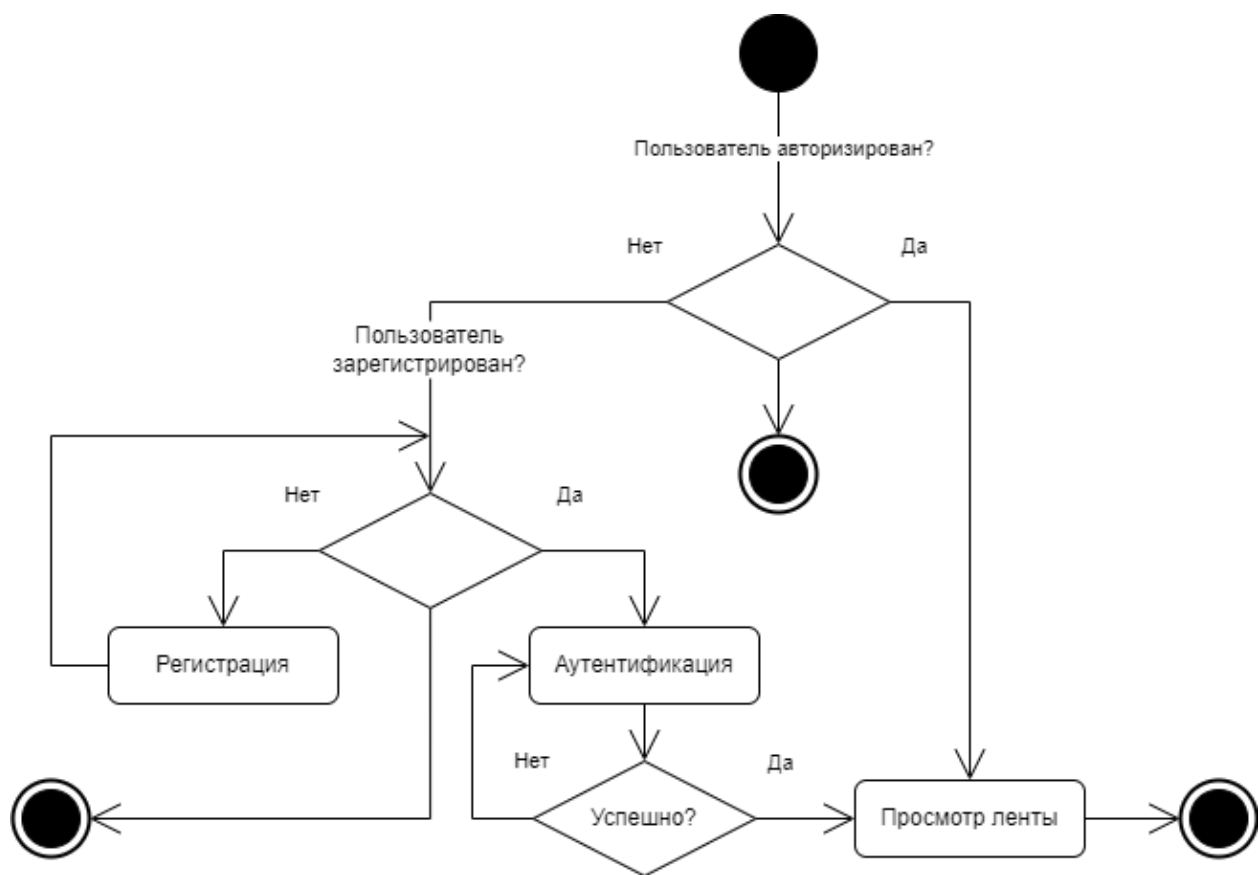


Рисунок 4 - Диаграмма состояний при авторизации пользователя

На Рисунке 5 изображена диаграмма состояний авторизованного пользователя.

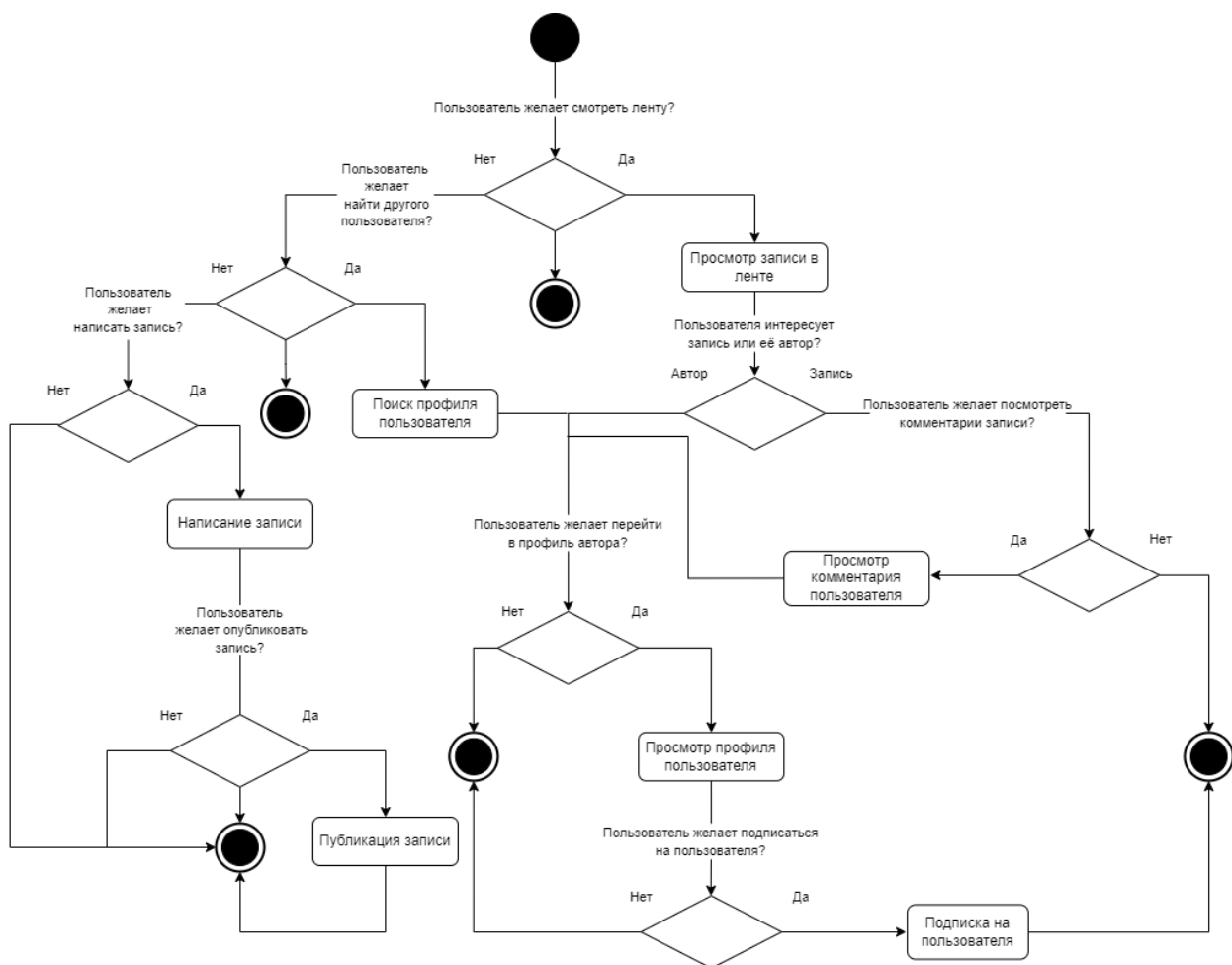


Рисунок 5 - Диаграмма состояний авторизованного пользователя

2.3.4 Диаграмма развертывания

На Рисунке 6 представлена диаграмма развертывания, визуализирующая компоненты и элементы системы.

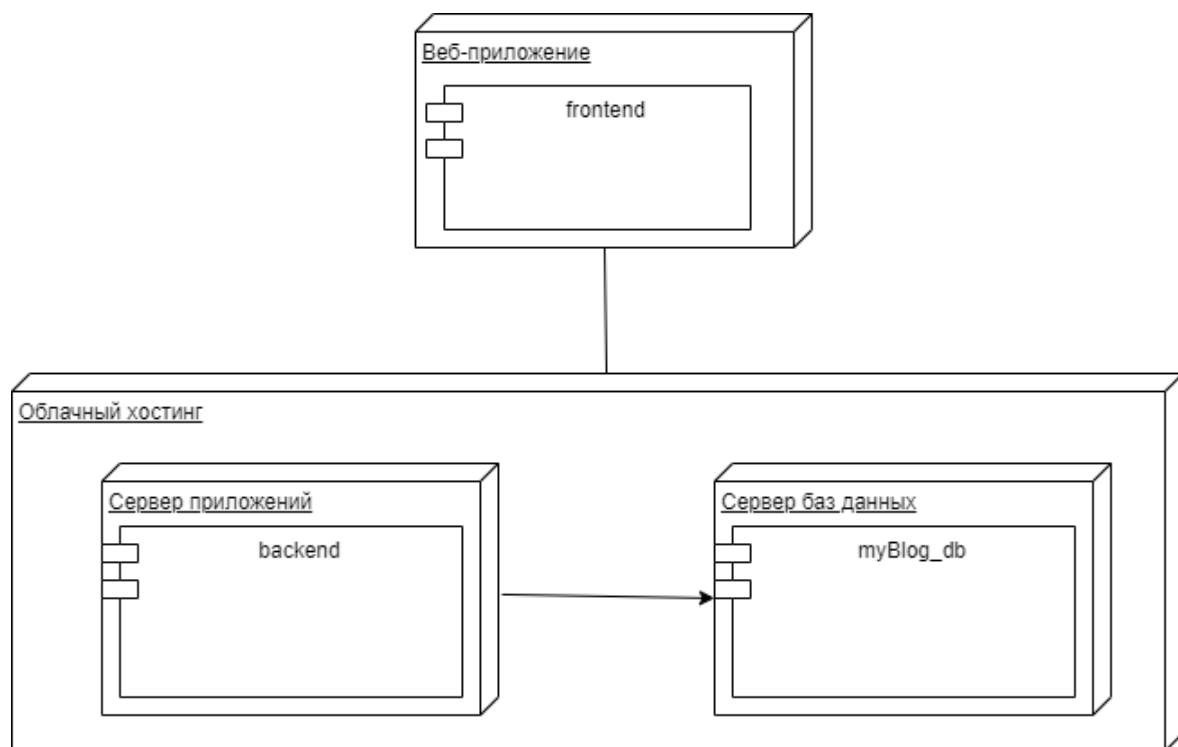


Рисунок 6 - Диаграмма развертывания приложения

3 Реализация

3.1 Средства реализации

Приложение реализовано с помощью следующих средств:

- Java – строго типизированный объектно-ориентированный язык программирования. Был выбран в качестве основного, так как он успел зарекомендовать себя как надежная и легко масштабируемая платформа разработки и до сих пор не потерял своей актуальности. К тому же существует огромное количество фреймворков и библиотек, написанных на Java, которые в перспективе можно легко интегрировать в проект;
- Spring Framework – универсальный фреймворк с открытым исходным кодом для Java-платформы. Был выбран, так как предоставляет мощные и удобные механизмы построения клиент-серверных приложений, в связи с чем пользуется огромным спросом и является фактически стандартом в построении приложений на Java;
- PostgreSQL – объектно-реляционная система управления базами данных. Является продуктом с открытым исходным кодом, который поддерживается многими серверами, в связи с чем и был выбран;
- Flyway – продукт с открытым исходным кодом для обеспечения миграций баз данных. Был выбран, так как легко интегрируется со Spring Framework и поддерживает PostgreSQL;
- Swagger – фреймворк для спецификации REST API.

3.2 Архитектура приложения

Приложение реализовано по трехуровневой архитектуре (Рисунок 7).

Клиентом выступает пользовательская часть приложения. На сервере развернута backend-часть приложения, которая обеспечивает доступ к третьему уровню архитектуры – серверу базы данных, который обеспечивает хранение данных.

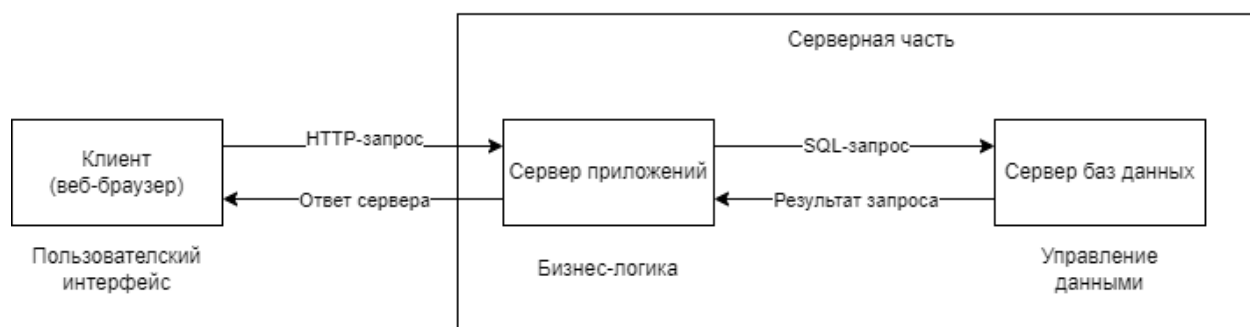


Рисунок 7 - Архитектура приложения

3.3 Реализация БД

В рамках проекта спроектирована и реализована база данных. Все таблицы представляют сущности предметной области (Рисунок 8).

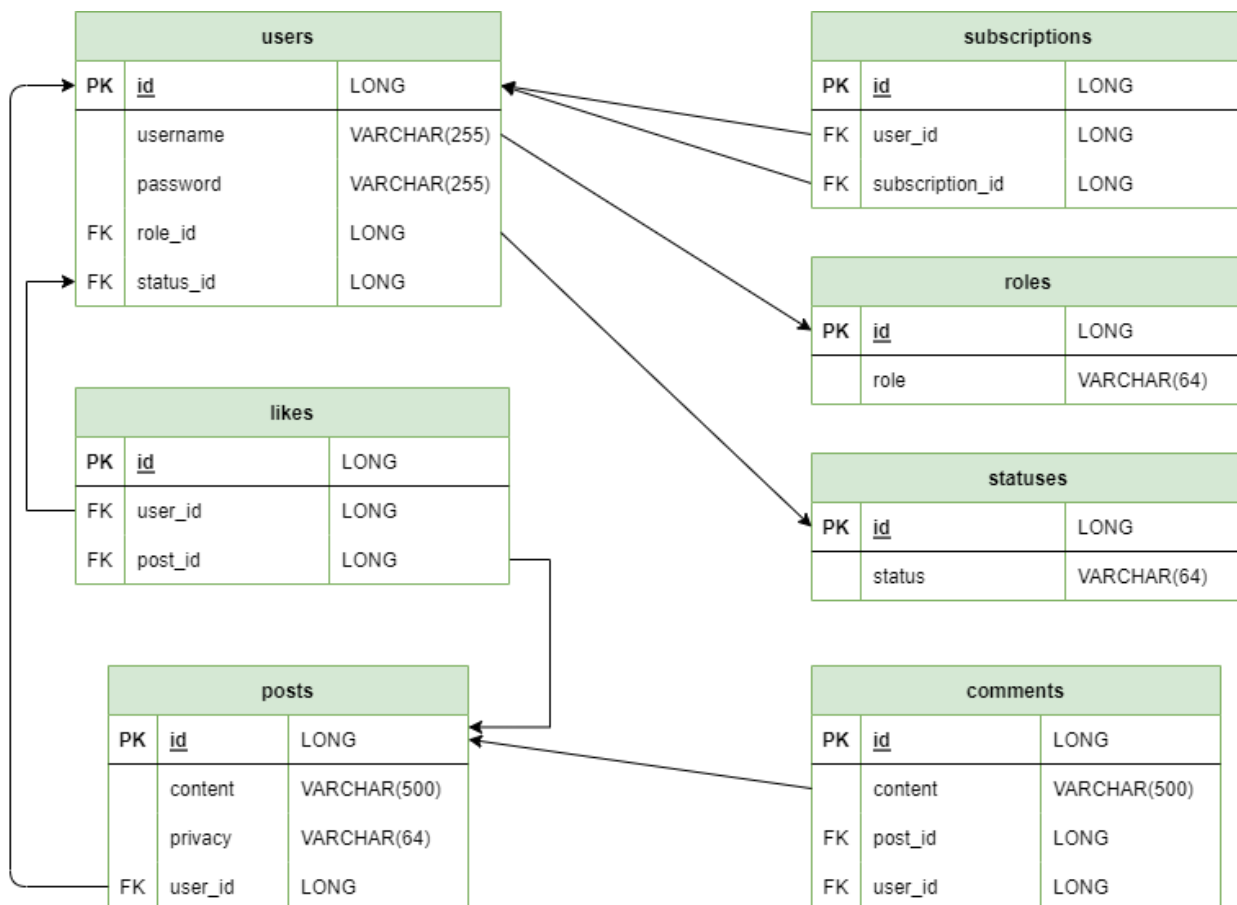


Рисунок 8 - Физическая модель данных

3.4 Реализация backend-составляющей приложения

Backend-составляющая приложения написана на языке Java с использованием компонентов Spring Framework. Архитектура представлена следующими слоями:

- Слой репозитория;
- Слой сервисов;
- Слой контроллеров.

Слой репозитория отвечает за поиск, сохранение, удаление и редактирование информации из базы данных. Базовые запросы представлены методами интерфейса `JpaRepository`. Более сложные запросы можно объявлять вручную посредством установки специальной аннотации над методом.

Слой сервисов отвечает за бизнес-логику приложения. В них реализован функционал, для удовлетворения потребностей пользователей.

Слой контроллеров отвечает за взаимодействие сервера с клиентом. Контроллеры представляют собой набор эндпоинтов REST API. Контроллер обрабатывает HTTP-запросы от клиента и передает его в слой сервисов. Полученный ответ контроллер отправляет в виде HTTP-ответа.

Заключение

В результате работы над проектом было реализовано веб-приложение, которое позволяет пользователям:

- Вести микроблоги;
- Ограничивать круг лиц, имеющих доступ к посту;
- Смотреть ленту постов;
- Ставить постам лайки;
- Оставлять комментарии под постами;
- Осуществлять поиск пользователей;
- Смотреть профили пользователей;
- Подписываться на пользователей;
- Смотреть личный профиль;
- Изменять пароль и никнейм.

Веб-приложение позволяет администраторам:

- Удалять посты;
- Удалять комментарии;
- Блокировать пользователей.

В ходе работы были выполнены следующие задачи:

- Собрана необходимая информация об исследуемой предметной области;
- Разработаны техническое задание и определенная им документация;
- Спроектирована и разработана база данных;
- Разработано веб-приложение;
- Проведено тестирование веб-приложения;
- Веб-приложение и база данных развернуты на удаленном сервере.

Цель курсовой работы считается достигнутой.

Список используемых источников

1. Spring Framework Documentation [Электронный ресурс]. – Режим доступа: <https://docs.spring.io/spring-framework/docs/current/reference/html/>. – Заглавие с экрана. – (Дата обращения: 16.09.2022).
2. PostgreSQL Documentation [Электронный ресурс]. – Режим доступа: <https://www.postgresql.org/docs/>. – Заглавие с экрана. – (Дата обращения: 16.09.2022).
3. Documentation Thymeleaf [Электронный ресурс]. – Режим доступа: <https://www.thymeleaf.org/documentation.html>. – Заглавие с экрана. – (Дата обращения: 16.09.2022).
4. Documentation Heroku Dev Center [Электронный ресурс]. – Режим доступа: <https://devcenter.heroku.com/categories/reference>. – Заглавие с экрана. – (Дата обращения: 16.09.2022).