Изображение выглядит как графическая вставка, Графика, мультфильм

Автоматически созданное описание

Зам. директора КМПО

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_С.Ф. Гасанов

«\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2024 г.

ДИПЛОМНЫЙ ПРОЕКТ

Тема: «Социальная сеть обмена информацией «Ganz»»

Специальность 09.02.07 Информационные системы и программирование

|  |  |  |
| --- | --- | --- |
| Выполнил студент(ка) группы 41ис-20 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | А.М-Б.Бацаев |
| Руководитель | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Н.А. Бабаева |
| Старший консультант | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | О.А. Калашникова |
| Консультант по технико-экономическому обоснованию работы | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | М.М. Трифонова |
| Нормоконтролер | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Н.А. Бабаева |

Москва 2024

**РЕФЕРАТ**

В данной расчетно-пояснительной записке, посвященной дипломному проекту на тему «Разработка социальной сети "Ganz"», подробно излагаются результаты работы: она содержит 80 страниц, на которых представлены 54 иллюстрации, размещены 2 таблицы, упомянуты 24 источника и включены 4 приложения.

Проект направлен на создание социальной сети, предназначенной для усовершенствования и оптимизации интерактивности между пользователями; эта платформа предусмотрена как средство для генерации и обмена контентом, а также как инструмент, обеспечивающий комфортное и защищенное пространство для коммуникации.

В аналитическом разделе осуществляется предпроектное обследование: проводится детальный анализ текущего состояния социальных сетей, выявляются существующие недостатки и формулируются требования к новой системе. На основе полученных данных формируются задачи, которые должны быть решены в рамках разработки данной социальной сети; к числу этих задач относятся как функциональные, так и нефункциональные требования, а также обоснование необходимости создания данной системы.

Проектный раздел документа описывает этапы эскизного и технического проектирования, включая разработку базы данных и подготовку рабочих чертежей. В рамках этого раздела формируется общая концепция системы, выбираются архитектурные и технологические решения, создается структура данных и ER-диаграмма; а также подготавливаются детализированные чертежи и схемы для фактической реализации системы. Для разработки интерфейса используются технологии HTML, CSS, JavaScript и фреймворк Bootstrap.

Экономический раздел включает анализ трудоемкости разработки, определение численности команды проекта, расчет фонда оплаты труда, себестоимости проектных работ и рентабельности предполагаемой системы. В этом разделе проводится также сравнение затрат и потенциальных выгод от внедрения социальной сети, чем обосновывается экономическая эффективность разработки.

Заключительным образом, можно утверждать, что данная записка является многоаспектным исследованием и проектированием социальной сети, включающим глубокий анализ, продуманное проектирование и всестороннюю экономическую оценку предстоящего внедрения.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 5](#_Toc168575995)

[1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛОСТИ 7](#_Toc168575996)

[1.1 Анализ существующих социальных сетей обмена информацией 7](#_Toc168575997)

[1.2 Описание основных функциональных возможностей социальной сети "Ganz" 11](#_Toc168575998)

[1.3 Определение основных целей и задач создания социальной сети "Ganz" 12](#_Toc168575999)

[2. ПРОЕКТИРОВАНИЕ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ 14](#_Toc168576000)

[2.1 Разработка предварительных проектных решений 14](#_Toc168576001)

[2.1.1 Разработка структуры сайта и основных функциональных модулей 16](#_Toc168576002)

[2.1.2 Определение требований к интерфейсу пользователя 18](#_Toc168576003)

[2.2 Проектирование базы данных 22](#_Toc168576004)

[2.2.1 Создание концептуальной модели данных 23](#_Toc168576005)

[2.2.2 Разработка логической модели базы данных 24](#_Toc168576006)

[2.3 Разработка проектных решений 26](#_Toc168576007)

[2.3.1 Разработка архитектуры системы 27](#_Toc168576008)

[2.3.2 Разработка интерфейса администратора 28](#_Toc168576009)

[2.3.3 Разработка интерфейса пользователя 29](#_Toc168576010)

[3. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ 32](#_Toc168576011)

[3.1 Настройка серверного окружения 32](#_Toc168576012)

[3.2 Создание базы данных 36](#_Toc168576013)

[3.3 Написание бэкенд-логики на Python с использованием Django 43](#_Toc168576014)

[3.4 Разработка фронтенд-части с использованием HTML, CSS, JavaScript и Bootstrap 48](#_Toc168576015)

[4 ТЕСТИРОВАНИЕ И ОТЛАДКА 59](#_Toc168576016)

[4.1 Проведение функционального тестирования 59](#_Toc168576017)

[4.2 Отладка ошибок и недочетов 64](#_Toc168576018)

[5. ТЕХНИКО-ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ 67](#_Toc168576019)

[ЗАКЛЮЧЕНИЕ 71](#_Toc168576020)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 73](#_Toc168576021)

[ПРИЛОЖЕНИЕ А 75](#_Toc168576022)

[ПРИЛОЖЕНИЕ Б 76](#_Toc168576023)

[ПРИЛОЖЕНИЕ В 77](#_Toc168576024)

[ПРИЛОЖЕНИЕ Г 78](#_Toc168576025)

# **ВВЕДЕНИЕ**

В современном информационном обществе социальные сети играют, несомненно, важную роль в повседневной жизни многих людей. Они являются не только средством коммуникации, но и ключевыми инструментами для обмена информацией, создания сообществ и формирования общественного мнения. За последнее десятилетие социальные платформы продемонстрировали удивительную способность объединять людей из разных уголков мира и давать им возможность выражать свои мысли, идеи и убеждения. Однако, несмотря на огромную важность и обширные преимущества социальных сетей, они сталкиваются с множеством непреодолимых проблем. В частности, многие платформы имеют ограниченный функционал или неудобный интерфейс, что может сильно затруднить их использование и снизить эффективность общения. Кроме того, некоторые социальные сети сталкиваются с проблемами с безопасностью данных или цензурой контента, что серьезно подрывает доверие пользователей к этим платформам. В этом контексте возникает острая необходимость в разработке новых и инновационных социальных сетей, отвечающих современным технологическим стандартам, удовлетворяющих потребности пользователей и способствующих обмену информацией и идеями. Целью данного исследования является разработка и внедрение новой социальной сети обмена информацией под названием «Ganz». Цель проекта — создание удобной, практичной и инновационной платформы, способной удовлетворить потребности пользователей в обмене информацией и общении в онлайн-среде. Для достижения этой цели были определены следующие ключевые моменты:

Проведение анализа существующих социальных сетей для обмена информацией: Планируется провести детальное исследование популярных социальных платформ, таких как Twitter, Reddit и других, с целью выявления их основных особенностей, преимуществ и недостатков. Этот анализ позволяет нам лучше понять потребности пользователей и выявить лучшие практики в этой области.

Описание основных функций социальной сети «Ganz»: Планируется подробное описание основных функций, предлагаемых платформой. Это включает в себя возможность аутентификации и регистрации пользователей, создания и обсуждения тем, поиска информации, а также создания профилей пользователей и управления ими.

Проектирование и реализация: На основе проведенного анализа существующих платформ и выявленных целей будет разработан детальный план создания и внедрения социальной сети "Ganz". Это включает в себя разработку архитектуры системы, выбор технологических решений, разработку пользовательского интерфейса и функционала, а также тестирование и оптимизацию работы платформы. Тестирование и оценка результатов: После завершения разработки планируется тестирование платформы на предмет ее функциональности, производительности и безопасности.

# 1. **ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛОСТИ**

## **Анализ существующих социальных сетей обмена информацией**

В наше время социальные медиа представляют собой один из ключевых каналов для обмена информацией, коммуникации и формирования общественного мнения. На фоне многочисленных платформ, предназначенных для обмена информацией, выделяются несколько, отличающихся особенностями функционала и популярностью среди пользователей. В данном разделе будут рассмотрены основные характеристики и особенности известных социальных сетей обмена информацией, включая Twitter и Reddit, а также другие популярные платформы.

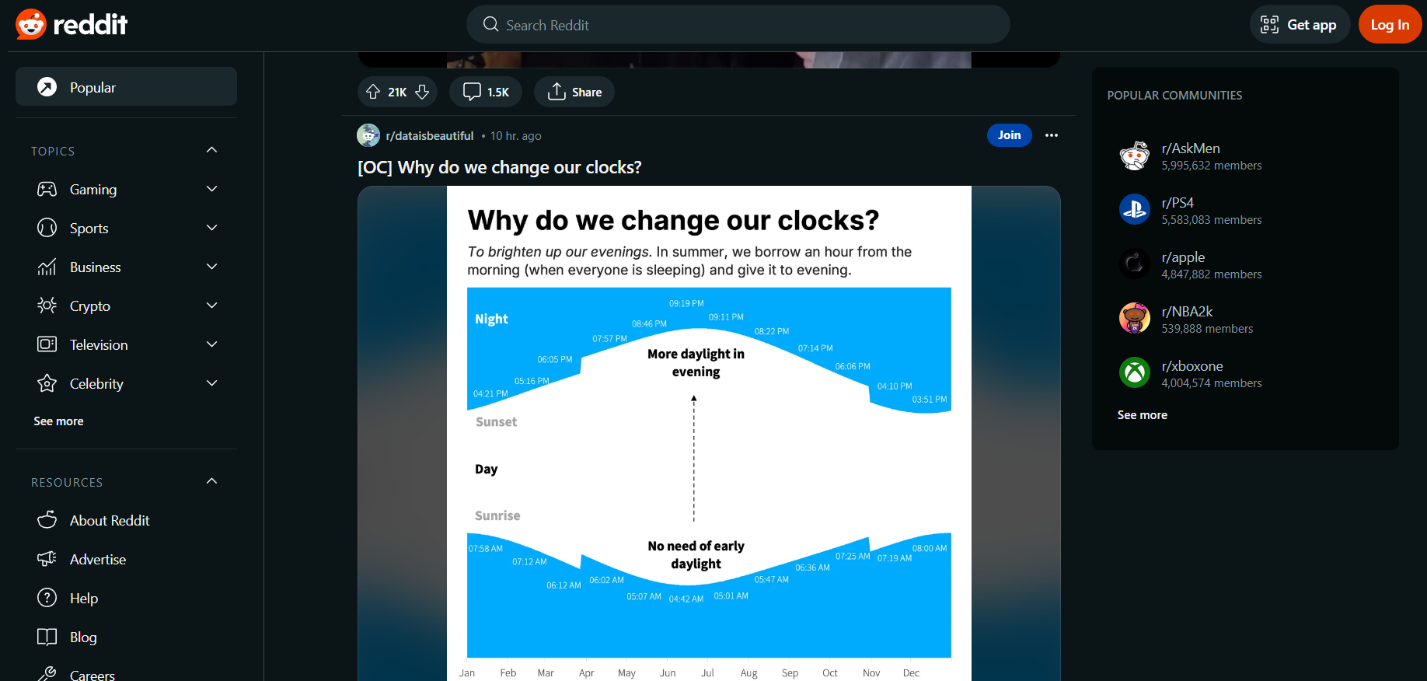
Twitter является одним из наиболее востребованных мировых социальных медиа для обмена информацией. Одной из его главных особенностей является ограничение на длину сообщения до 280 символов, что способствует краткости и лаконичности передачи мыслей. На упомянутой интерактивной платформе, обозначаемой как "Twitter", участникам предоставляется возможность эффективно публиковать короткие сообщения, известные как "твиты". Уникальные особенности данной платформы также включают в себя подписку на аккаунты иных лиц, просмотр и анализ публикуемых ими твитов, а также интерактивное взаимодействие с контентом — как путём реагирования на сообщения, так и через использование хештегов и упоминаний других участников сети. Вследствие широкого спектра функций, в частности возможности делиться контентом, данный ресурс выступает как мощный канал для мгновенного распространения новостной информации и других сообщений, что способствует его популярности среди пользователей, предпочитающих обсуждать актуальные новости. (Рисунок 1).

Благодаря возможности оперативных репостов, Twitter обладает выдающейся способностью быстро распространять информацию, что делает его неотъемлемым инструментом в арсенале современных средств массовой информации и оказывает заметное влияние на формирование общественного мнения. Эта функциональность не только способствует бурному обмену мнениями, но и представляет значительную ценность для оперативного получения и распространения свежих новостных данных, что подчёркивает его статус важной площадки для ведения общественных дискуссий.

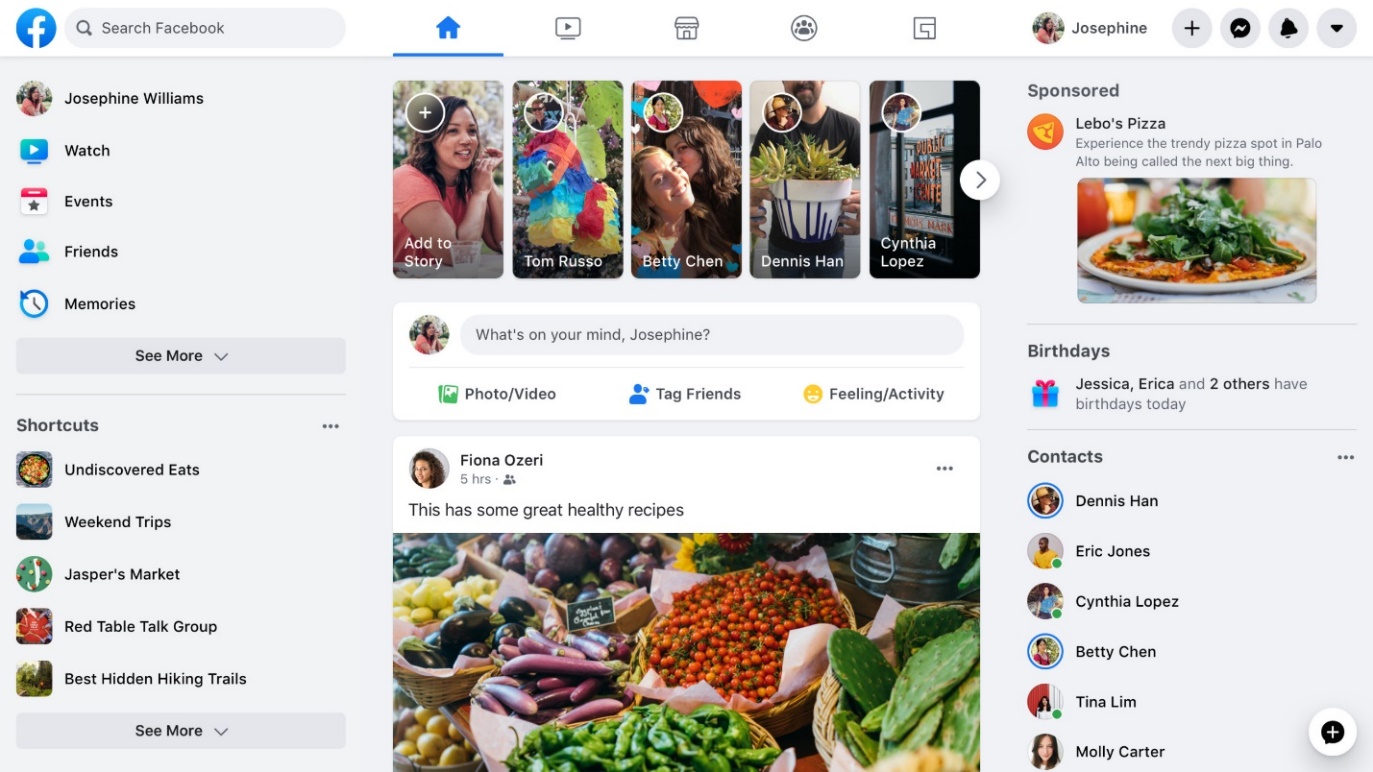
Рисунок 1 - Twitter

Reddit, представляющий собой социальную сеть, служит многофункциональной платформой для обсуждения разнообразных тем. Участники данного сообщества имеют возможность инициировать новые дискуссии или активно участвовать в уже существующих, выражая свои мнения в комментариях к темам, поднятым другими пользователями. Кроме того, они могут выбирать интересы, которые находят специальное отражение в их предпочтениях.

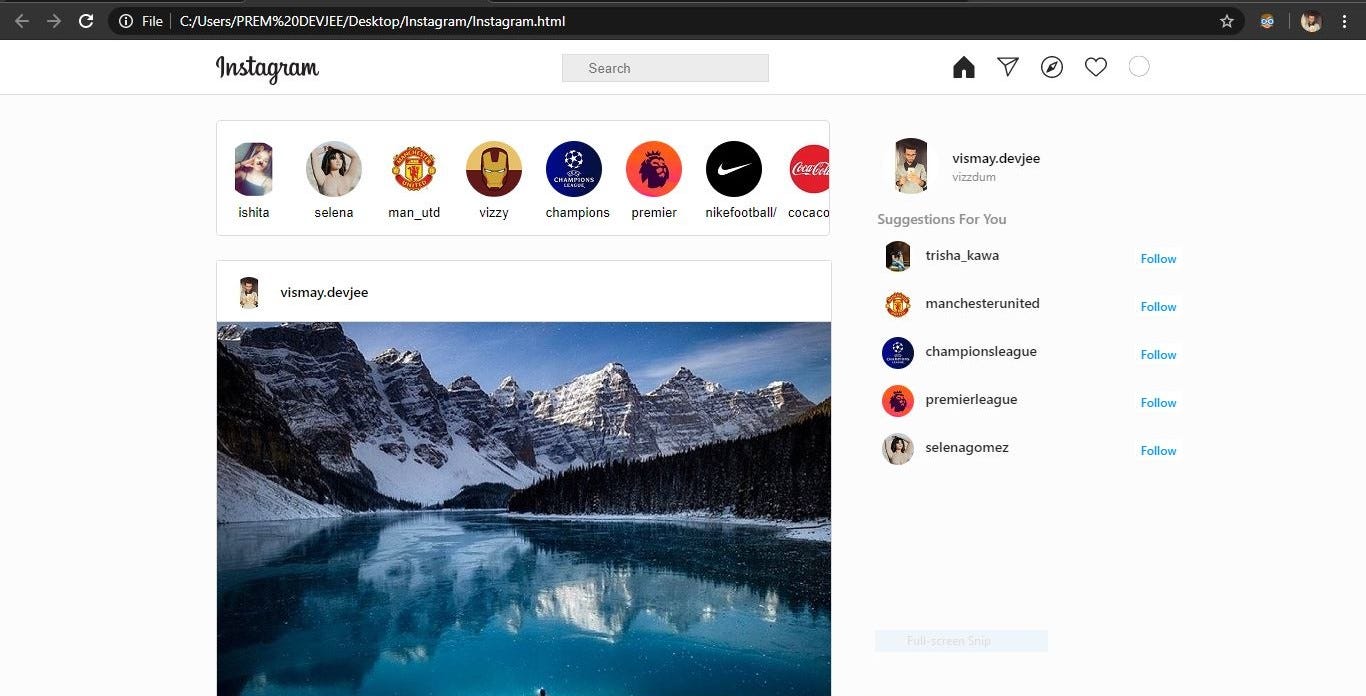
Обширный диапазон обсуждаемых на Reddit тем включает в себя новости, политику и даже мемы, что, несомненно, способствует популярности данной платформы. Уникальная атмосфера, складывающаяся благодаря активному взаимодействию пользователей, делает эту социальную сеть особенно привлекательной для ведения разнообразных дискуссий. (Рисунок 2).

Рисунок 2 - Reddit 

В соответствии с действующими канонами в области информационных технологий, неоспоримым является статус Facebook как одной из преобладающих глобальных сетевых платформ, осуществляющей объединение миллиардов индивидов по всему миру в единое информационное пространство. Данная платформа предоставляет пользователям разнообразные возможности, начиная от конфигурации индивидуальных аккаунтов до публикации материалов, расширения социальных связей путём добавления контактов, обмена сообщениями, а также просмотра обновлений в ленте новостей. Особое внимание стоит уделить функционалу, позволяющему формировать группы, что способствует более тесному взаимодействию пользователей и эффективному обмену информацией. (См. Рисунок 3).

Рисунок 3 – Facebook 

Instagram представляет собой мобильное приложение, которое служит платформой для обмена фотографиями и видеоматериалами. Эта цифровая среда позволяет пользователям не только демонстрировать собственные визуальные произведения широкой аудитории, но и просматривать, а также оценивать контент, размещённый другими участниками сети, через механизмы такие, как "лайки" и комментарии. Дополнительно, приложение обладает инструментарием для редактирования, предоставляющим различные фильтры и настройки, что способствует улучшению визуальной аттрактивности фотографий и видеозаписей. (См. Рисунок 4).

Рисунок 4 – Instagram **

Данные социальные сети представляют собой лишь незначительный сегмент многообразия онлайн-платформ, предназначенных для обмена информацией между пользователями. Каждая из этих сетей обладает уникальными характеристиками, которые находят отклик в предпочтениях их пользователей. Однако, неизменно возникает необходимость в внедрении нововведений и интересных решений для адаптации к динамически изменяющимся запросам аудитории.

## **Описание основных функциональных возможностей социальной сети** **"Ganz"**

Социальная платформа "Ganz" является пространством для коммуникаций, предоставляя широкий спектр функциональных возможностей для обмена информацией и взаимодействия с другими пользователями. В числе ключевых функций выделяются:

Процесс регистрации и авторизации пользователей: возможность формирования индивидуальных аккаунтов, аутентификации в системе и восстановления парольных данных в случае их утраты — каждому участнику доступна опция конфигурации личного профиля с возможностью добавления персональной информации.

Менеджмент публикаций: пользователи имеют право на создание, модификацию и удаление постов, отправку и приём текстовых сообщений, что способствует динамичному взаимодействию в рамках сети.

Интерактив в комментариях и экспрессия мнений через "лайки": платформа обеспечивает возможность оставлять комментарии и выражать своё отношение к материалам других участников, что стимулирует обмен мнениями и поддерживает активность в сообществе.

Система личных сообщений и поисковые функции: интегрированные мессенджеры позволяют пользователям осуществлять прямое общение, а наличие функций поиска облегчает навигацию по интересующим темам и следование за активностями других участников.

Данные функции делают "Ganz" не только удобным, но и привлекательным ресурсом для коммуникации и обмена информацией в современном цифровом пространстве.

## **1.3 Определение основных целей и задач создания социальной сети "Ganz"**

Социальная платформа Ganz разработана с целью облегчения процесса взаимного обмена информацией, идеями, мнениями и медийными материалами между пользователями, что способствует активизации их взаимодействия. В эру глобальной информатизации цифровые социальные сети становятся интегральной частью существования современного человека, вписываясь в его ежедневную реальность как незаменимый элемент. Они не только фасилитируют самовыражение, но и расширяют горизонты коммуникации, знакомства и дискуссий актуальных вопросов, тем самым создавая условия для обширного социального взаимодействия и обмена мнениями по множеству тем.

Инициатива развития социальной сети Ganz представляет собой важный проект, направленный на усовершенствование информационного пространства и активизацию социальных коммуникаций. Проект "Ganz" ставит перед собой ряд задач: от разработки эффективной платформы для обмена данными по широкому спектру тематик до формирования многообразного сообщества, которое способствует сотрудничеству и взаимной поддержке участников. Особое значение придается предоставлению пользователям инструментов для общения и интерактивного взаимодействия, что является критически важным. Важнейшей миссией Ganz является стимулирование креативного самовыражения среди участников, а также создание безопасного и дружественного интернет-пространства, где каждый член сообщества мог бы чувствовать себя защищенным и комфортно.

В рамках реализации стратегических направлений Ganz определены основные задачи, включая разработку системы регистрации и аутентификации пользователей, что обеспечит безопасность и комфорт при использовании платформы. Ключевым аспектом также является разработка функционала для управления профилями участников. Планируется внедрение возможностей для создания тем, комментирования и выражения благодарности за контент. Приоритетом остается создание эффективных поисковых механизмов и инструментов фильтрации информации, что значительно упростит доступ к актуальным данным. Предусмотрено также развитие ресурсов для управления контентом, модерации сообщества и оперативного устранения возможных неполадок, что в целом улучшит безопасность и эффективность взаимодействий внутри платформы. В конечном итоге, стремление к максимальной масштабируемости и повышению производительности системы направлено на обеспечение ее стабильности и высокой скорости реакции интерфейса.

# **2. ПРОЕКТИРОВАНИЕ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ**

## **2.1 Разработка предварительных проектных решений**

В сфере разработки архитектурных объектов, индустриального оборудования или программных изделий, наивысшее значение приобретает инициализирующая стадия проектирования — тот период, когда концептуальные замыслы обретают визуальную форму и когда определяются фундаментальные директивы для предстоящей реализации. Данный этап не просто предоставляет оценку потенциального развития проекта, но и задаёт его курс и направления. Специфический процесс концептуализации социальной сети Ganz охватывает формулирование стратегических аспектов, комплексный анализ и сбор данных, разработку и визуализацию основополагающей идеи, а также последующую фиксацию результатов в документальном виде.

На первой стадии проекта «Ganz», важно не только определить цели и задачи, которые подразумевают создание пространства для свободного обмена информацией, мнениями, идеями между участниками, но и построить уникальную онлайн-платформу для взаимодействия единомышленников. Эта площадка предназначена для дискуссий по актуальным темам, обмена опытом и общения по различным вопросам; такой виртуальный форум способствует развитию цивилизованных отношений с обществом, укреплению социальных связей и формированию международного сообщества, что становится критически важной задачей проекта.

Особая роль в функционировании платформы Ganz отводится механизмам создания дискуссионных тем и возможностям комментирования — инструментам, благодаря которым участники обмениваются мнениями и позициями. Предвидится, что функциональность платформы позволит пользователям не только порождать новые тематические направления, но и обсуждать уже существующие, оценивая содержание, предложенное другими участниками.

Не менее значимыми являются разработка и реализация систем авторизации и управления профилями, благодаря которым каждому пользователю обеспечивается доступ к возможностям платформы и предоставляется возможность персонализации собственного профиля, включая данные о личных интересах и предпочтениях. Вопросы модерации и обеспечения безопасности — ключевые аспекты, сохраняющие целостность платформы Ganz, включают механизмы, направленные на устранение нежелательного контента, борьбу с вредоносными действиями и защиту конфиденциальности личных данных пользователей.

Создание интуитивно понятного интерфейса, сопровождающего лёгкость навигации по платформе и доступность её ключевых функций, требует разработки визуально привлекательного дизайна, удобного для использования как на стационарных, так и на портативных устройствах. Поддержание динамичного развития и адаптация к инновациям — долгосрочные задачи проекта Ganz, решение которых предусматривает разработку модульной архитектуры и применение передовых технологий для повышения производительности и эффективности системы.

Анализ информационных потоков из аналогичных социальных сетей, таких как Twitter и Reddit, привёл к выявлению критических характеристик и потребностей пользователей; это исследование помогло определить ключевые модули и функции проекта, ставшие неотъемлемой частью его успешной реализации. В проект включены механизмы авторизации и аутентификации, создание тем для дискуссий и их управление, возможности комментирования и интерактивного взаимодействия, настройка профилей с возможностью добавления друзей и обмена сообщениями, а также административный интерфейс для мониторинга контента и пользователей. Визуализация концепции проекта демонстрируется через разработанные пользовательские интерфейсы и панели управления, позволяющие оценить предварительно предполагаемый визуальный стиль и функциональность платформы, способствующие дальнейшему усовершенствованию её дизайна. Подготовка документации проекта, включая подробные описания функций, опций и руководств для пользователей и администраторов, будет завершена и представлена до официального запуска приложения для публичного использования.

## **2.1.1 Разработка структуры сайта и основных функциональных модулей**

В рамках разрабатываемого проекта веб-платформы и ключевых функциональных элементов социальной сети Ganz особое внимание, безусловно, следует уделять детализации основных компонентов интерфейса. Одной из первостепенных задач является организация системной взаимосвязи между этими элементами. Цель такой структурированности — не только удовлетворение непосредственных потребностей пользователей, но и реализация стратегических целей данной платформы. Так, важность проработки каждого аспекта интерфейса и архитектуры системы не может быть переоценена: это залог успешной адаптации платформы к изменяющимся условиям виртуального пространства, обеспечивая при этом её стабильное функционирование и эффективное взаимодействие с пользователем.

В рамках разработки веб-платформы социальной сети Ganz, абсолютным приоритетом является детальное рассмотрение и анализ исходных элементов дизайна вместе с их внутренними взаимосвязями. Это действие нацелено на достижение удовлетворения запросов пользователей и реализацию стратегических амбиций данной платформы. Применяется следующая структура сайта, обусловленная ключевыми компонентами его функциональных модулей:

На домашней странице предлагается — как бы мимоходом — ознакомление с текущими событиями платформы, включая список фаворитных и недавно обсуждавшихся тем; не исключены и самые последние комментарии. Функционал входа и регистрации, куда входят механизмы авторизации, обеспечивает возможность входа на сайт либо создания новой учетной записи для тех, кто ранее не был зарегистрирован.

Профиль пользователя — это пространство, где представлена возможность модификации персональных данных, включая, но не ограничиваясь, описанием человека, его контактной информацией и другими атрибутами. Дополнительно, предусмотрена функция добавления других пользователей в список друзей. Лента активности отображает онлайн-действия участников, такие как комментирование и создание тем.

Мотивы участия углубляются через создание тем для обсуждений, с выбором подходящих категорий и возможностью фильтрации доступных дискуссий. Участие в голосовании, направленное на выявление популярных тем, также является частью интерактивности. Поиск по ключевым словам значительно упрощает навигацию по платформе.

Комментарии и их последующие ответы способствуют глубине дискуссий, стимулируя обмен мнениями. Система обмена сообщениями поддерживает приватный диалог между участниками, а также управление полученной корреспонденцией.

Процедура входа на сайт сопровождается использованием логина и пароля, тогда как процесс регистрации новых пользователей включает заполнение обязательных полей и верификацию по электронной почте, обеспечивая таким образом безопасность и контроль доступа.

Каждый модуль данной системы обладает индивидуальной логикой и предназначен для взаимодействия с другими элементами, что способствует созданию удобного, интерактивного и многогранного пользовательского интерфейса. Таким образом, эти компоненты являются основополагающими для функционирования социальной сети «Ganz».

## **2.1.2 Определение требований к интерфейсу пользователя**

Формулировка критериев для пользовательского интерфейса, безусловно, олицетворяет один из наиболее значимых этапов в разработке любой социальной сети, подобной Ganz. В данном контексте, структура интерфейса веб-приложения выполняет роль критического элемента, который определяет первое впечатление о продукте и его функциональной простоте. Императивно, чтобы интерфейс оказался не только интуитивно понятным, но и функциональным, дабы обеспечить высокий уровень пользовательского взаимодействия на разнообразных стадиях управления аналогичными платформами.

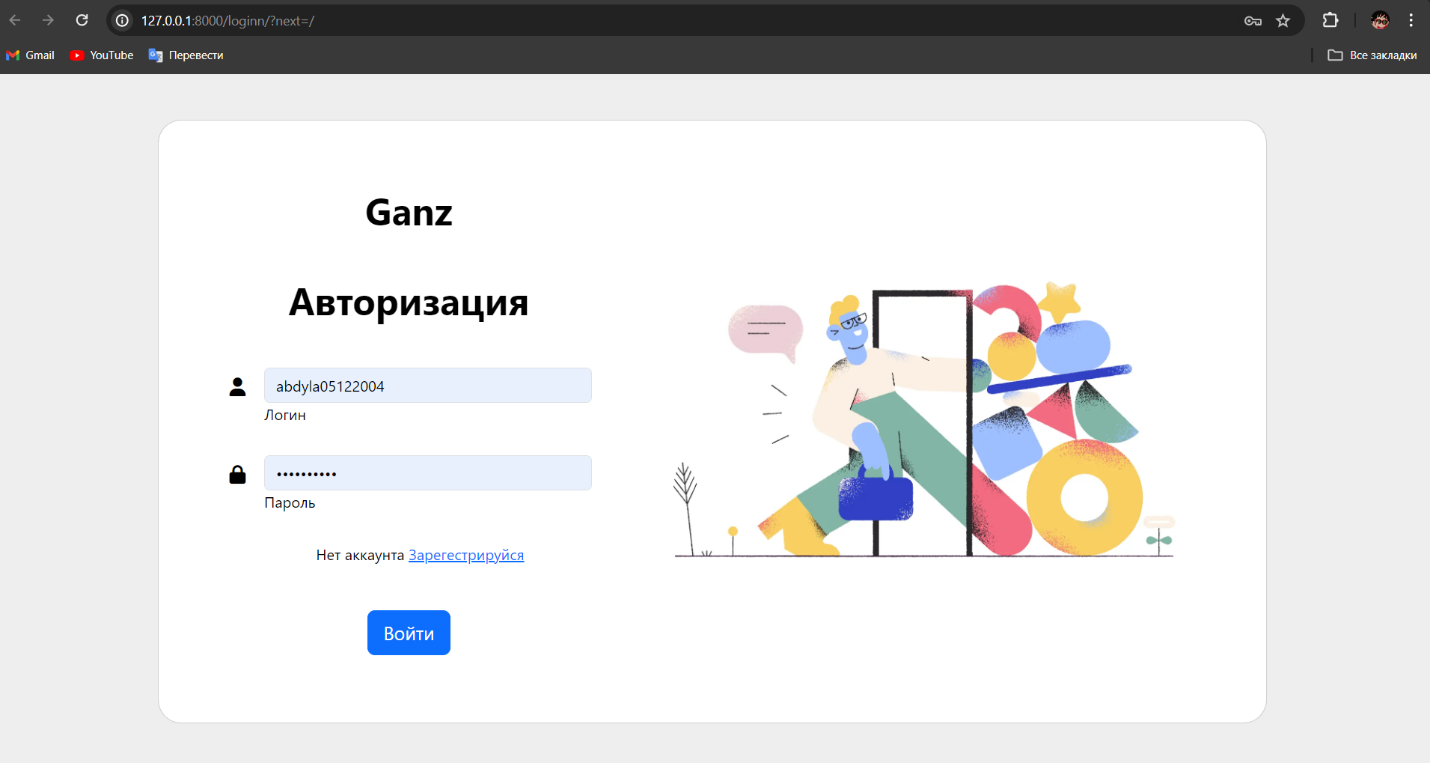
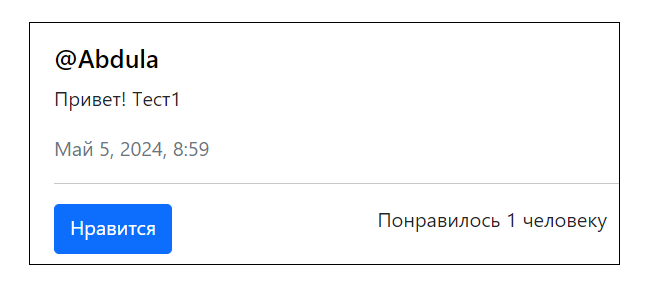
Компоненты процесса регистрации и аутентификации занимают центральное место в разработке интерфейса. Разработка удобной формы для входа в систему и регистрации новых учётных записей — необходимое условие. Аутентификационная форма должна включать поля для ввода идентификатора и пароля; необходимы также элементы управления, такие как кнопка активации входа и ссылка на страницу восстановления пароля в случае его утраты. Форма регистрации должна предоставлять возможность ввода основной информации — имени пользователя, адреса электронной почты, пароля — и произвести верификацию уникальности идентификатора и адреса электронной почты. Смотреть Рисунок 4.

Рисунок 4 - Пример страницы "Авторизация"

В роли «лица» виртуального пространства социальной сети, главная страница должна отображать актуальные новости по наиболее востребованным темам. Она должна включать в себя не только функционал для инициирования новой темы или публикации поста, но и разнообразные элементы навигации — эти элементы крайне необходимы для упрощения доступа ко всем сегментам приложения. Весьма значимо, чтобы данная страница отличалась не только визуальной привлекательностью, но и информационной насыщенностью, что привлекает внимание пользователей и стимулирует их к глубокому изучению контента.

Основные функции, такие как создание тем и комментирование, являются стержневыми для социальной сети Ganz; таким образом, интерфейс должен способствовать простому доступу к этим функциям. Предполагается, что для инициации новой темы или поста пользователю будет предложена специализированная форма, оборудованная полями для ввода названия и текста сообщения. Участникам сети будет предоставлена возможность выражать свои мысли через комментарии к опубликованным темам. Форма комментариев, будучи интуитивно понятной и простой в использовании, должна позволять пользователям с лёгкостью делиться своими взглядами или задавать вопросы по интересующим их темам.

Дизайнерская концепция и визуальная реализация интерфейсов занимают доминирующие позиции в процессе разработки привлекательных пользовательских систем; необходимость адаптации к современным веяниям в сфере веб-дизайна и следование передовым практикам становится ключевым условием для гарантирования эффективного пользовательского взаимодействия. В центре внимания находится создание эстетически привлекательного внешнего вида, который не только формирует положительные впечатления среди пользователей, но и гармонично интегрируется в стилистические рамки бренда, при этом используется Bootstrap как инструментарий для реализации визуально притягательных и функциональных интерфейсов — см. рисунок 5.

Рисунок 5 – Иллюстрация поста пользователя.

Основополагающим фактором дизайна является его отзывчивость, адаптированность к разнообразным типам устройств и размерам экранов. Применение технологий адаптивного дизайна, таких как CSS медиазапросы и адаптивные сетки, обеспечивает необходимую гибкость и универсальность доступа. Критическую значимость имеет производительность и скорость реакции интерфейса — факторы, которые не подлежат недооценке. Ожидания современных пользователей касательно отзывчивости и плавности работы приложения предписывают оптимизацию кода, применение методик сжатия и минимизации файлов CSS и JavaScript, а также стратегии кэширования ресурсов для достижения высочайших стандартов производительности.

## **2.2 Проектирование базы данных**

В процессе инициации разработки информационной структуры социальной сети «Ganz», первостепенное внимание уделяется идентификации и анализу основополагающих компонентов данной системы — к примеру, таковыми являются пользователи, темы обсуждений, комментарии, а также возможные дополнительные элементы, включающие в себя группы участников или метки. Отмечается необходимость чёткого определения не только ключевых функций каждого из перечисленных элементов, но и механизмов их взаимодействия с другими составляющими данной сети.

В дополнение к вышеуказанному, учитывая функциональные требования, представляется необходимым внедрение новых таблиц и полей в базу данных для реализации специфических операций, таких как интеграция меток или группирование пользователей. Эта мера предназначена для обеспечения гибкости и расширяемости функционала социальной сети, позволяя ей адаптироваться к изменяющимся запросам и предпочтениям её пользователей. На основе определенных атрибутов и функций данных элементов, следующий шаг включает разработку структуры их взаимосвязей: таким образом, каждый комментарий должен быть ассоциирован с определенным пользователем и темой через внешние ключи, ссылающиеся на соответствующие записи в таблицах пользователей и тем.

Ключевым аспектом является также гарантирование целостности данных и предотвращение их избыточности, что достигается посредством процедуры нормализации базы данных, последующей за первичным этапом формирования структуры. Кроме того, для повышения производительности приложения, необходимо оптимизировать обработку запросов.

Разработка базы данных для "Ganz" представляет собой многоэтапный и сложный процесс, начиная от тщательного анализа требований приложения и заканчивая установлением оптимальной структуры данных. Этот процесс должен привести к созданию хорошо спроектированной и эффективно функционирующей базы данных, что, в свою очередь, обеспечит высокую производительность всей системы.

## **2.2.1 Создание концептуальной модели данных**

Формулирование концептуальной модели данных для социальной сети "Ganz" является принципиальным этапом в процессе конструирования информационной архитектуры; ведь указанная модель закладывает не только фундамент для определения ключевых элементов системы, но и создаёт карту их многоуровневых взаимодействий. Она обеспечивает детализацию структуры информации, предназначенной для аккумуляции в базе данных, при этом исключая из рассмотрения технические аспекты реализации на уровне таблиц и полей.

Предварительный анализ требований к функционалу социальной сети "Ganz" остаётся необходимым для выявления основных функций, поддерживаемых базой данных. Среди вероятных ключевых элементов следует выделить: пользователей, темы, посты и комментарии, что подчёркивает сложность задачи по созданию системы, аналогичной Reddit.

Пользователи: Являются краеугольным камнем любой социальной сети. Для каждого из них обязательна информация: уникальный идентификатор (ID), имя (Username), электронная почта (Email), пароль (Password) и дата регистрации (Registration Date). Кроме того, следует учитывать разнообразие ролей пользователей – от обычных до администраторов – поскольку от их статуса зависит доступность функций приложения.

Посты: Представляют собой текстовые или мультимедийные материалы, размещаемые пользователями на платформе. Каждый пост содержит уникальный идентификатор (ID), заголовок (Title), информацию о содержании (Content), авторе (Author), связанном с таблицей пользователей, и дату создания (Creation Date).

Комментарии: Позволяют пользователям обсуждать посты и делиться мнениями. Каждый комментарий содержит уникальный идентификатор (ID), текст (Content), информацию об авторе (Author), связанном с таблицей пользователей, и ссылку на тему (Topic), а также дату создания (Creation Date).

Понимание взаимосвязей между этими сущностями открывает понимание структуры социальной сети: пользователи могут создавать темы, посты и комментарии; подписываться на других пользователей или темы. Посты и комментарии формируют иерархическую структуру, где комментарии могут быть ответами на другие комментарии.

## **2.2.2 Разработка логической модели базы данных**

Процесс разработки логической модели базы данных для социальной сети «Ganz» занимает центральное место в рамках проектных работ; этот этап не только требует всестороннего анализа, но и подразумевает тщательное, детализированное предварительное планирование. Логическая модель, которая формулирует сущности, атрибуты и многоуровневые взаимосвязи между ними, служит зеркальным отображением структуры базы данных, охватывая её логическую организацию.

Данная модель не просто складывается из абстрактных элементов, а структурируется как хорошо отлаженный механизм, где каждая деталь — будь то сущность или атрибут — имеет своё назначение в обеспечении целостности и функциональности базы данных. Это предполагает, что каждый аспект модели должен быть взвешен и продуман до мельчайших подробностей, обеспечивая таким образом гармоничное и эффективное взаимодействие всех компонентов системы.

В контексте проектирования логической модели для базы данных «Ganz» немаловажно учитывать специфику социальных сетей и их функциональные возможности. Учитывая, что социальная сеть «Ganz» предоставляет пользователям возможности для создания тем и оставления комментариев, необходимо включить в базу данных таблицы, которые будут аккумулировать информацию о пользователях, темах, комментариях и других критически важных элементах, обеспечивающих функционирование сервиса.

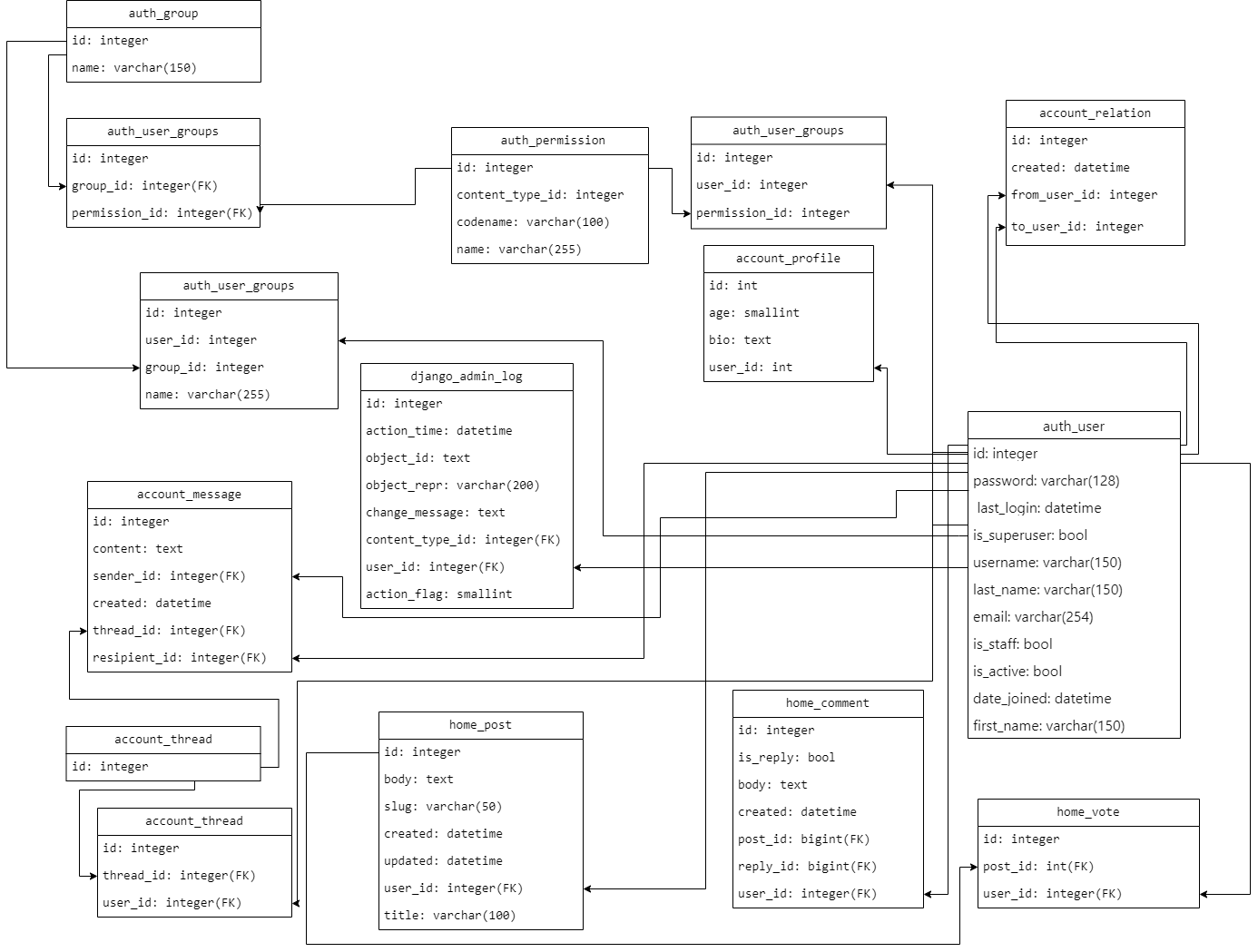
Логическая модель базы данных социальной сети «Ganz» структурирована таким образом, что она включает в себя таблицы, предназначенные для хранения данных о пользователях (auth\_user), группах пользователей (auth\_group), разрешениях (auth\_permission), отслеживании действий администраторов (django\_admin\_log) и управлении сессиями пользователей (django\_session).

Таблица auth\_user аккумулирует основную информацию о пользователях, включая такие данные, как идентификационный номер, логин, пароль, дату регистрации и другие сведения, критически необходимые для аутентификации и управления учетными записями. Для регулирования прав доступа целесообразно применять таблицу auth\_group, в которой собраны данные о различных группах пользователей. В таблице auth\_permission фиксируется информация о разрешениях, которые могут быть предоставлены как отдельным пользователям, так и группам для выполнения конкретных действий внутри системы.

Таблицы django\_session и django\_admin\_log задействованы для регистрации действий администраторов и управления сессиями пользователей соответственно, выполняя ключевую роль в обеспечении безопасности системы и мониторинге активности пользователей.

Не менее важно в рамках логической модели учитывать взаимосвязи между различными сущностями. Так, таблицы auth\_user\_groups и auth\_user\_user\_permissions связаны с таблицей auth\_user посредством внешних ключей, что позволяет контролировать принадлежность пользователей к определенным группам и доступ к соответствующим разрешениям.

В заключение, разработка логической модели базы данных «Ganz» требует тщательного учета всех необходимых сущностей и их связей, что является обязательным условием для эффективного хранения и управления данными, необходимыми для надлежащего функционирования социальной сети. Рисунок 6.

 Рисунок 6 - Логическая модель базы данных

## **2.3 Разработка проектных решений**

Проектирование архитектурных решений для социальной сети «Ganz» предполагает глубокое и всестороннее планирование, охватывающее многообразие аспектов — от структуры системы до пользовательского интерфейса. На данном этапе проекта задачей становится определение ключевых принципов, которые будут лежать в основе функционирования и дизайна предстоящего веб-приложения.

## **2.3.1 Разработка архитектуры системы**

В рамках представленного проекта, реализация которого целиком и полностью зависит от архитектонической конструкции системы, последняя определяет механизмы взаимодействия различных компонентов. Таким образом, обеспечивается необходимая производительность, масштабируемость и безусловная безопасность функционирования приложения. Именно архитектура служит каркасом, который диктует параметры синергии между элементами системы.

Серверная часть приложения «Ganz», социальной сети нового поколения, была разработана на языке Python, применяя фреймворк Django — инструмент, способствующий ускоренной разработке веб-приложений. Django, в свою очередь, предоставляет широкий спектр инструментов и библиотек, существенно упрощающих работу с базами данных, аутентификацией пользователей и системой маршрутизации запросов.

Реляционная база данных SQLite3 была выбрана в качестве основы данных. Она поддерживает широкий спектр функциональностей, необходимых для выполнения многочисленных задач.

Разработка клиентской части приложения включает использование технологий HTML, CSS и JavaScript. Будет применена рамка для создания современного адаптивного дизайна, что обеспечит корректное отображение на разнообразных устройствах.

Взаимодействие между клиентской и серверной частями приложения будет осуществляться через RESTful API. Django REST Framework (DRF) предоставляет обширные возможности для документирования и создания API, значительно упрощая тем самым процесс разработки и последующую поддержку.

Аспект безопасности занимает ключевую роль в дизайне проекта. Применение современных средств защиты, таких как HTTPS для шифрования данных, защита от атак CSRF и XSS, а также обеспечение надлежащей аутентификации и авторизации пользователей, подтверждает это утверждение.

## **2.3.2 Разработка интерфейса администратора**

Административный интерфейс, который задуман как инструментарий для модерации контента и осуществления контроля над системой, обязан отличаться простотой в освоении и эксплуатации, а также обеспечивать обширный спектр функций для управления приложением. Для разработки упомянутой платформы предполагается использование встроенной панели управления Django; данная система славится своей функциональностью и удобством в настройке, что делает её идеальной для задач администрирования. Эффективность такого подхода лежит в его способности сократить время на внедрение и увеличить точность управления, чем и заслуживает особого внимания.

Административный интерфейс, который будет внедрён, предусматривает реализацию ряда функциональных возможностей. В рамках обеспечения контроля над пользователями данной системы, необходимо подчеркнуть важность возможностей администратора: просмотр списка участников, редактирование представленной ими информации и, в определённых ситуациях, блокирование или удаление аккаунтов. Для упрощения этих операций используется интерфейс Django Admin, оснащённый функциональными средствами для фильтрации и поиска пользователей, основанными на множестве параметров.

В контексте модерации контента, административный аппарат вправе осуществлять контроль за темами, комментариями и другими элементами контента, проверяя их на соответствие установленным стандартам платформы, а также классифицируя их как спам или нарушения правил. Управленческая панель Django способствует эффективности и простоте управления разнообразными моделями данных, что, в свою очередь, оптимизирует процесс модерации.

Особенное внимание заслуживает специализированный раздел настроек приложения, доступный администратору. Этот раздел предоставляет возможности для управления параметрами безопасности, доступности, уведомлениями и другими настройками, которые можно модифицировать для повышения функциональности приложения. Все упомянутые параметры настраиваются через интерфейс Django Admin, предлагающий необходимые инструменты для этой цели.

Нельзя упускать из виду аналитику и отчётность как важнейшие элементы административной деятельности, поскольку администратору необходим доступ к обширной аналитической информации, связанной с деятельностью приложения. Дашборды, отображающие ключевые показатели использования, активности пользователей и отчёты о производительности и безопасности, незаменимы для принятия обоснованных управленческих решений. Для расширения функциональности аналитических возможностей, Django Admin может быть дополнен за счёт интеграции дополнительных приложений и виджетов.

Критическую важность представляет система уведомлений, обеспечивающая своевременное реагирование на события в приложении, такие как регистрация новых пользователей, обращения в службу поддержки или выявленные проблемы безопасности. Интеграция с различными системами уведомлений через платформу Django предоставляет администратору возможности настройки автоматизированных уведомлений, что является неотъемлемой частью управления приложением.

## **2.3.3 Разработка интерфейса пользователя**

В свете последних достижений в сфере информационных технологий, эффективность веб-приложений во многом зависит от качества разработки пользовательского интерфейса. Следовательно, для достижения высокого уровня функциональной удобности и полноты удовлетворения потребительских ожиданий необходимо, чтобы данный интерфейс не только обладал функциональной насыщенностью и скоростью реакции, но и отличался эстетической привлекательностью, а также интуитивной понятностью — таковы непреложные требования современности.

С точки зрения концептуальных подходов к разработке пользовательских интерфейсов, особую значимость приобретает адаптация главной страницы социальной сети «Ganz» к вершинам критериев привлекательности и информационной насыщенности. Ответственность за это возлагается на тех, кто занимается интеграцией визуальных и функциональных аспектов, дабы обеспечить как визуальную привлекательность, так и комплексное удовлетворение информационных запросов пользователей. Так, она должна включать в себя не только актуализированный перечень наиболее востребованных тем, но и интуитивно понятные элементы навигации — это позволит пользователям с максимальной оперативностью осуществлять переходы между различными разделами приложения.

Переходя к странице темы, пользователь встречается с содержанием, заголовком и комментариями к теме. В этом контексте он получает возможность не только прочитать уже имеющиеся мнения, но и активно участвовать в обсуждении, оставляя собственные отзывы или реагируя на мнения других.

Что касается страницы профиля, здесь пользователь обнаружит не только свои личные данные, созданные им темы и комментарии, но и сможет вносить изменения в свои данные и настройки, что значительно упрощает процесс персонализации взаимодействия с приложением.

Необходимо особо отметить, что формы ввода — будь то аутентификация, регистрация, создание темы или комментирование — должны быть максимально упрощены для пользователя. Они должны содержать все необходимые поля ввода и кнопки отправки, при этом обеспечивая проверку корректности введённых данных.

Навигация и поиск в приложении должны быть настолько понятными и легкими, чтобы пользователи могли без труда перемещаться по разделам, а также быстро находить необходимую информацию по интересующим их темам или ключевым словам.

В процессе разработки интерфейсов, — именно такого рода деятельности, где уделяется внимание мельчайшим деталям, — принципиальным остаётся постулат о необходимости глубокого и всестороннего учета амбиций и предпочтений тех, кто в конечном итоге будет использовать продукт: конечных пользователей. Следовательно, каждый элемент, каждая деталь интерфейса должны быть исполнены таким образом, чтобы их использование было интуитивно понятным, легкодоступным и высокофункциональным.

Далее, если погрузиться в аспекты разработки интерфейса, можно выявить, что его кардинальная цель заключается в создании такого продукта, который, преодолевая рамки технических параметров, достигает гармонии с ожиданиями и предпочтениями пользователей, предоставляя им не только необходимые функции, но и даря максимальное удовлетворение от взаимодействия с приложением. Оттого, способность создавать продукты, которые являются не только конкурентоспособными, но и желанными на рынке, проистекает из непрерывного процесса совершенствования, адаптации интерфейса к динамично меняющимся и разнообразным запросам пользовательской аудитории.

# **3. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ**

## **3.1 Настройка серверного окружения**

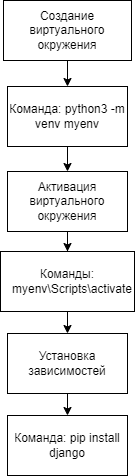
Настройка серверного окружения — начальная фаза в процессе создания и последующего развертывания социальной сети "Ganz"; процесс, который предусматривает подготовку необходимых программных решений и инструментов, целью которой является обеспечение стабильности и безопасности функционирования системы. В ходе данного этапа, осуществляется установка и конфигурирование виртуального окружения, что позволяет эффективно изолировать зависимости проекта, исключая риск конфликтов между библиотеками, применяемыми в рамках различных проектных инициатив.

Рисунок 7 - Диаграмма создания виртуального окружения

Создание и настройка проекта Django:

В рамках предварительного этапа проектной деятельности, имеющей отношение к разработке на платформе Django, происходит формирование нового проекта после адаптации виртуальной среды, что влечёт за собой создание соответствующих приложений. Организация проектной структуры предполагает чёткое логическое разграничение функционала — действие, способствующее не только оптимизации разработки, но и облегчению процесса поддержки кодовой базы.

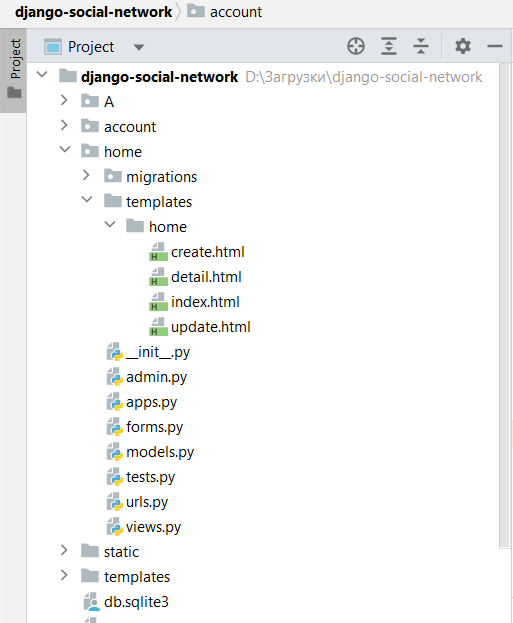


Рисунок 8 - Структура проекта Django

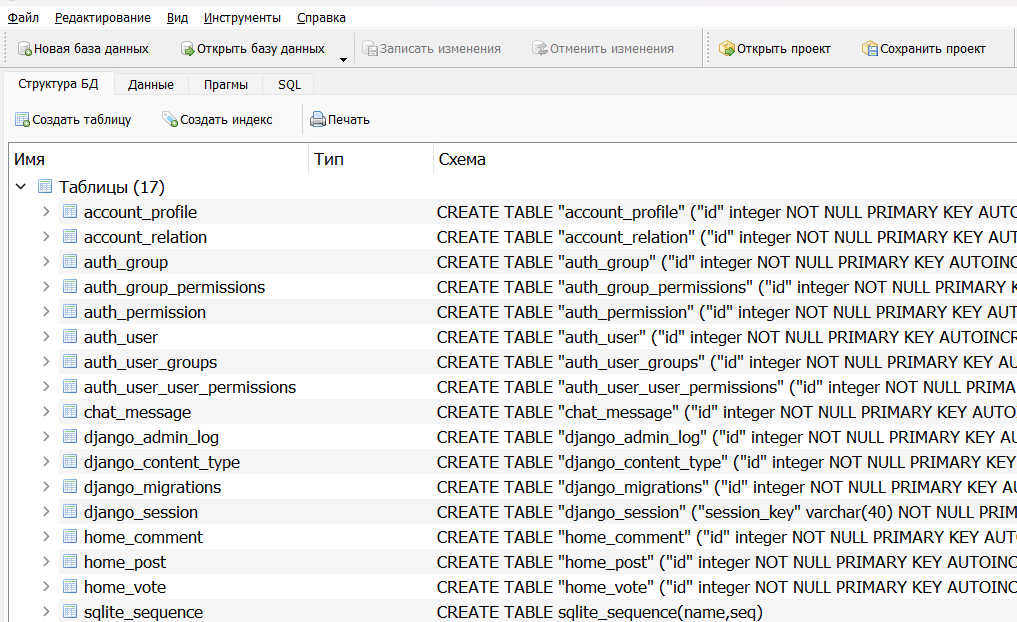
 В качестве инструмента управления базами данных выбрана SQLite3; данная система обладает преимуществами в контексте разработки и тестирования: установка и конфигурация дополнительных компонентов не требуется.

Рисунок 9 - ПО для работы с SQLite3

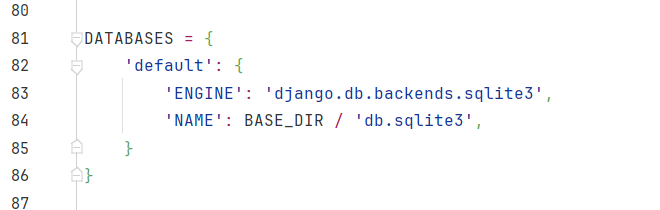
Настройка базы данных происходит в файле settings.py, что позволяет интегрировать базу данных в общую структуру проекта без излишних усилий

Рисунок 10 - Настройка базы данных в settings.py

Настройка URL маршрутизации:

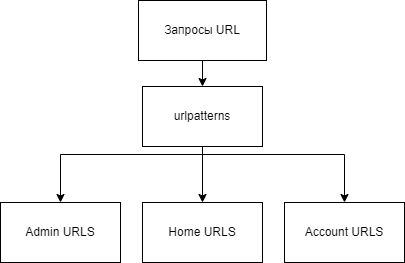
Что касается настройки URL-маршрутизации, то она, как ключевой элемент архитектуры веб-приложения, гарантирует правильное распределение запросов между различными компонентами системы (например, регистрация, авторизация, создание темы, комментирование), тем самым облегчая интерактивность взаимодействия с пользователем.

Рисунок 11 - Маршрутизации URL

Настройка шаблонов и статики:

Настройка шаблонов и статических файлов выполняет важную функцию в динамической генерации HTML-страниц, обеспечивая при этом стандартизацию внешнего вида и структуры интерфейсов, что способствует улучшению пользовательского опыта. Сохранение шаблонов и статических файлов в предназначенных для этого директориях способствует созданию единообразного и когерентного стиля оформления всех страниц приложения.

Шаблоны используются для динамической генерации HTML страниц. Для этого в проекте настроены директории для хранения шаблонов и статических файлов, что обеспечивает единообразный стиль и структуру всех страниц приложения.

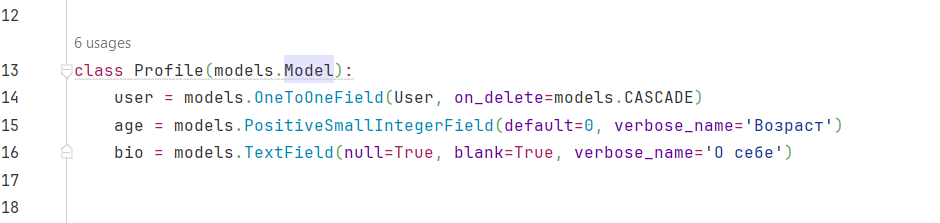
Рисунок 12 - Настройки шаблонов

## **3.2 Создание базы данных**

В процессе разработки базовой базы данных социальной сети ««Ganz»» ее создание является ключевым этапом – она собирает и обеспечивает безопасность всех критически важных данных, необходимых для адекватного функционирования всей системы. Эти данные включают информацию о пользователях, обсуждаемые темы и комментарии. Описанный здесь процесс включает в себя не только разработку с использованием SQLite и Django ORM, но и этапы от точной спецификации моделей данных до их тщательной реализации

Концептуализация моделей данных — процесс, охватывающий детальное определение и структурирование информационных блоков, которые предназначены для основы системы хранения данных в социальной сети "Ganz". С целью реализации функциональных возможностей указанной социальной платформы были специфицированы следующие модели данных:

Пользователь и Профиль: Модель «User», базирующаяся на встроенной в Django модели, включает в себя ключевую информацию о пользователях. Модель «Profile», дополняя её, вносит дополнительные поля — например, возраст и биографию. Рисунок 13 иллюстрирует модель «Profile».

Рисунок 13 – Модель Profile

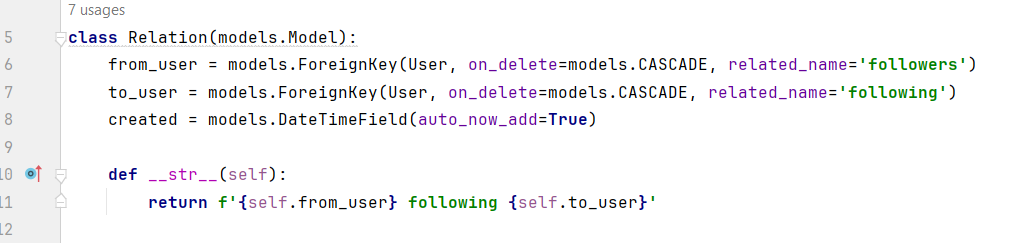
Отношения между пользователями: Модель «Relation» отвечает за реализацию связей между пользователями, обеспечивая возможность одним пользователям подписываться на других. Для наглядности представлена на Рисунке 14. Продолжение этой идеи визуализировано на Рисунке 14, который демонстрирует структуру модели «Relation».

Рисунок 14 – Модель Relation

Посты и Комментарии: Модель «Post» описывает публикации пользователей, включающие поля для текста, заголовка, даты создания и последующего обновления — Рисунок 15. Модель «Comment», предназначенная для описания комментариев к постам, содержит поля для текста комментария, данных о пользователе и ссылки на соответствующий пост. Отображено на Рисунке 16.

Рисунок 15 – Модель Post 

Рисунок 16 – Модель Comment

1. Голосования:

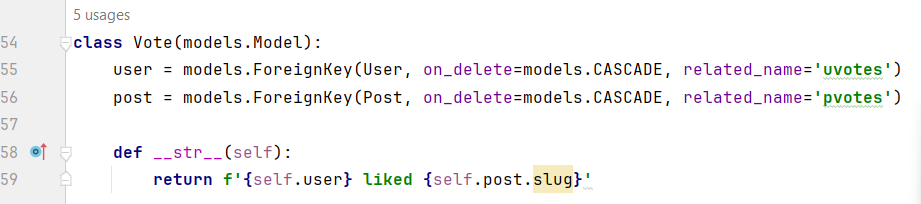
Модель Vote описывает систему лайков за посты. Рисунок 15.

Рисунок 16 – Модель Vote

1. Сообщения:

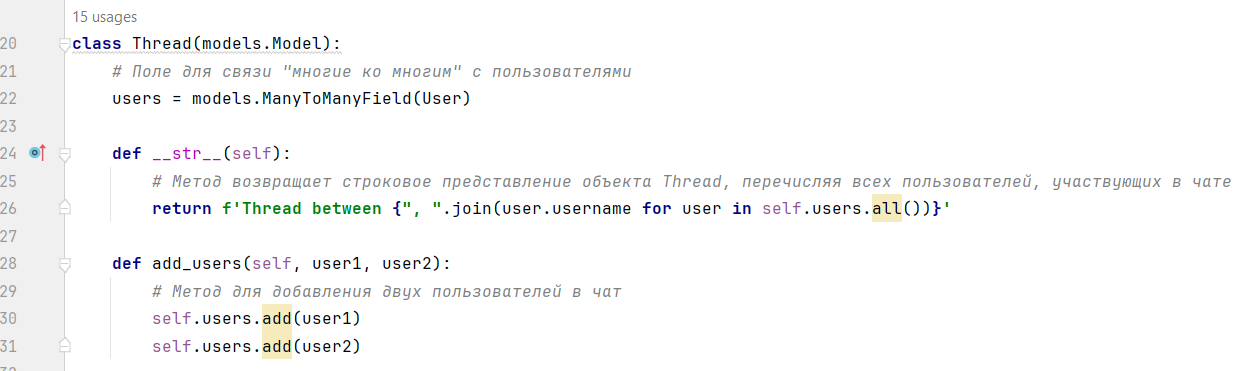
Модель Message и Thread реализуют систему личных сообщений между пользователями. Рисунок 17 и Рисунок 18

Рисунок 17 – Модель Thread

Рисунок 18 – Модель Message

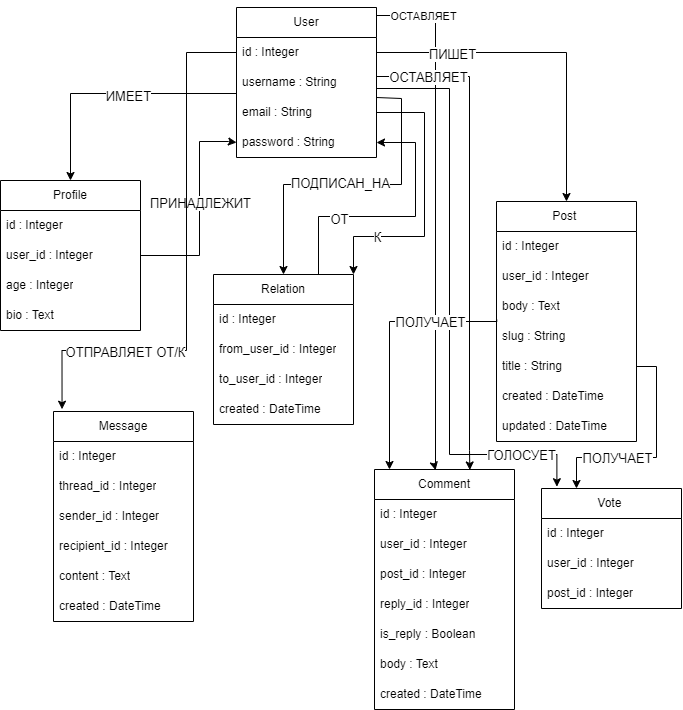


Рисунок 19 – Общая схема моделей данных

В разделе, посвященном моделированию данных, следующим логическим этапом стало проектирование и реализация миграционных процессов. Суть миграций заключается в трансформации предварительно определенных моделей в соответствующие SQL-запросы, цель которых — создание необходимых таблиц в базе данных. Для инициации процесса миграции использовалась следующая команда: *Python manage.py makemigration*, что стало кульминацией подготовительного этапа в разработке базы данных.

Проверка структуры базы данных

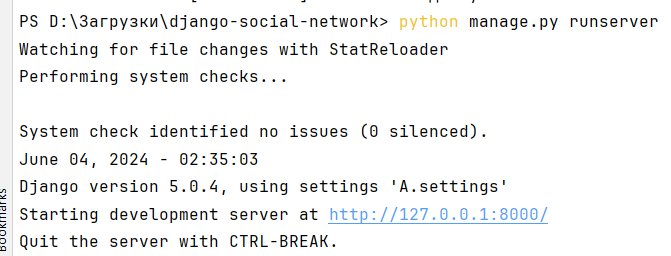
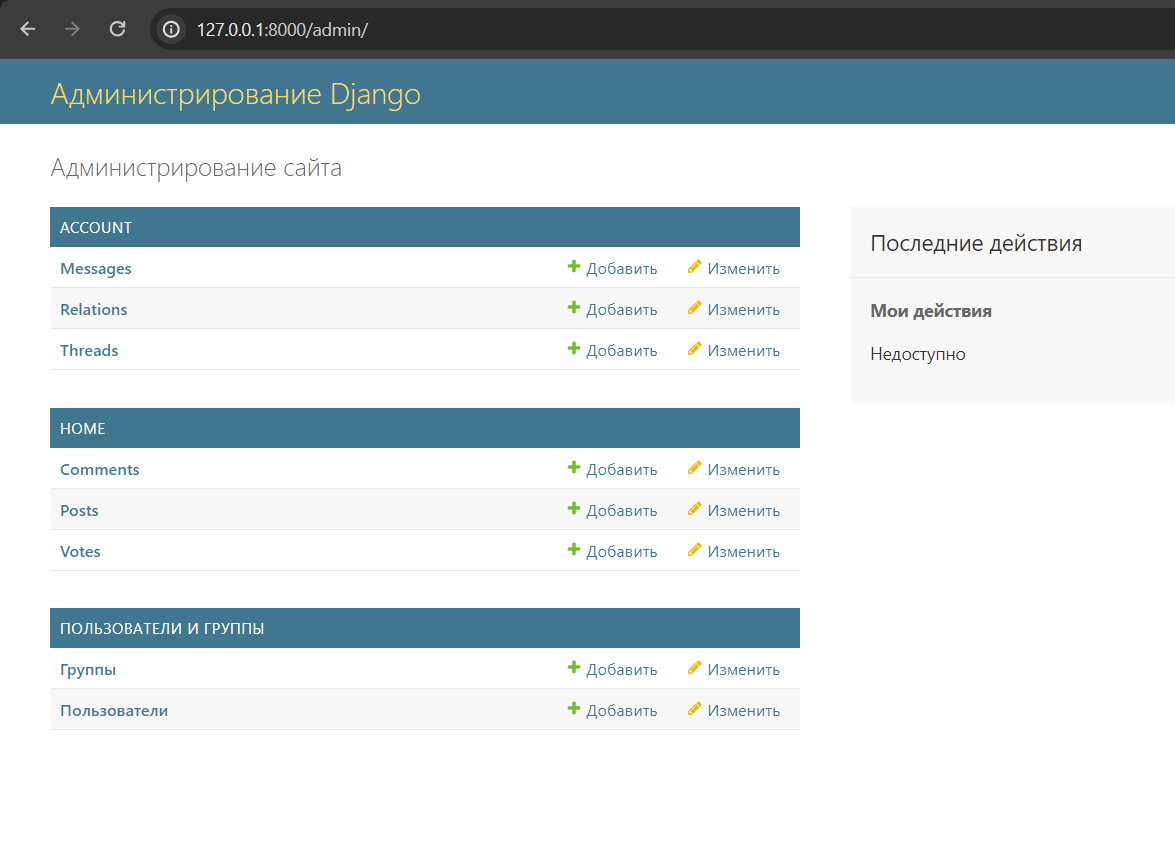
Последующая верификация структуры базы данных осуществлялась посредством административной панели Django, инструментарий которой обеспечивает не только функциональное управление данными, но и позволяет глубоко анализировать корректность формирования всех таблиц и полей базы данных. Для доступа к административной панели было необходимо запустить сервер разработки, что изображено на рисунке 20.

Рисунок 20 – Запуск сервера

Данный сервер стал доступен по адресу: http://127.0.0.1:8000/admin/. Авторизация в системе под учетной записью администратора открывала возможность просмотра всех сформированных моделей и аудита их структуры, что демонстрируется на рисунке 21.

Рисунок 21 – Административная панель 

В процессе разработки базы данных для социальной сети "Ganz", необходимо было сначала определить модели данных. Последующее создание и апробация миграций, а также тщательная проверка структуры самой базы данных стали критическими этапами, обеспечивающими надёжное хранение информации. Данные мероприятия гарантировали корректное функционирование всех ключевых возможностей системы. Среди них — регистрация пользователей, их аутентификация, формирование постов, комментирование, реализация механизмов голосования и обмен личными сообщениями. Каждый из этих этапов способствовал созданию устойчивой и функционально насыщенной инфраструктуры, обеспечивающей бесперебойную и безошибочную работу всех составляющих системы.

## **3.3 Написание бэкенд-логики на Python с использованием Django**

Разработка бэкенд-логики для социальной сети "Ganz" является важным этапом в процессе создания функционального и безопасного веб-приложения. Этот раздел подробно описывает процесс написания серверной части проекта с использованием фреймворка Django, включая настройку основных компонентов, разработку моделей, представлений (views) и сериализаторов (serializers), а также обеспечение взаимодействия с фронтендом.

Настройка проекта и создание приложений

Проект "Ganz" был инициализирован с использованием команды:

*—Django admin startproject ganz*

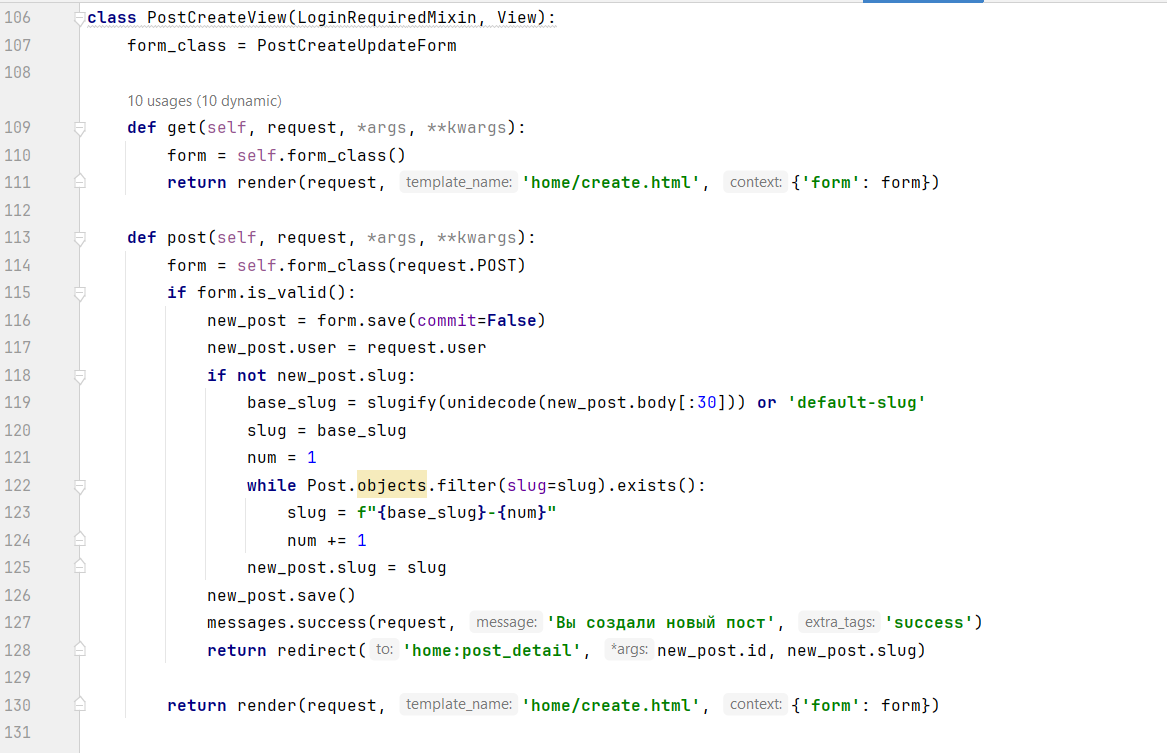
После создания основного проекта были созданы два приложения: account для управления пользователями и профилями, и home для работы с постами, комментариями и голосованиями.

*— python manage.py startapp account*

*— python manage.py startapp home*

Создание и настройка представлений (views)

Представления (views) обрабатывают запросы и возвращают соответствующие ответы. В приложении home были созданы представления для работы с постами и комментариями. На Рисунке 22 приведен пример создания представления для публикации нового поста.

Рисунок 22 – Представление для создания поста

Настройка маршрутизации (URLs)

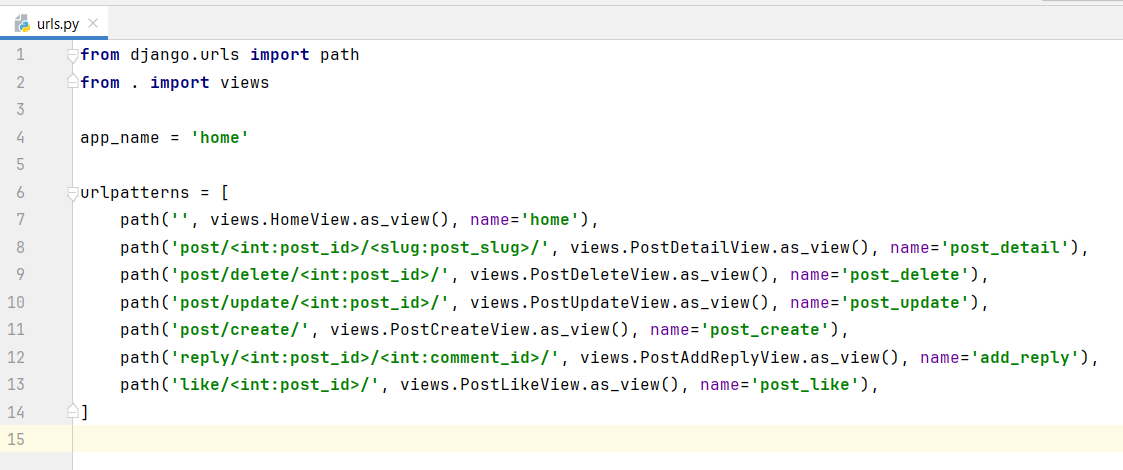
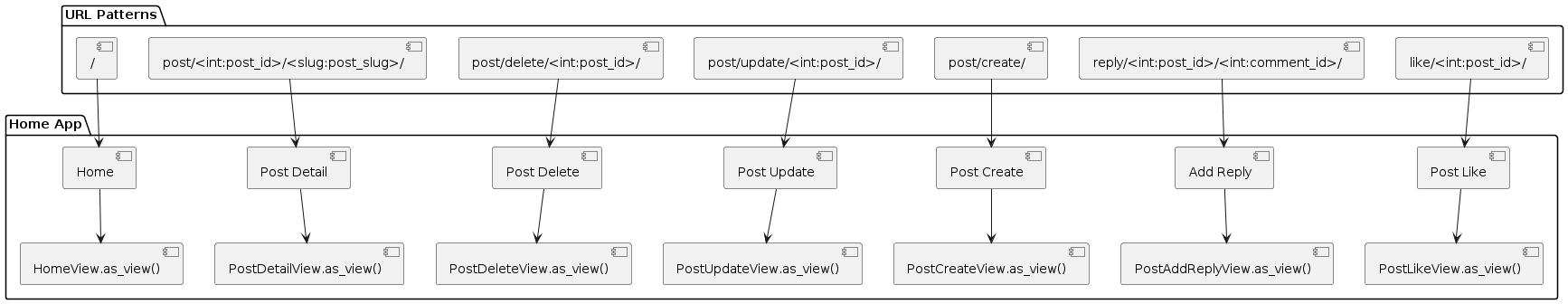
Для обработки запросов от пользователей были настроены URL-маршруты. Например, для приложения home был создан файл urls.py, который связывает запросы с соответствующими представлениями. Рисунок 23.

Рисунок 23 – Файл маршрутизации

Схема маршрутизации (URL routing) для моего кода отображает пути и связанные с ними представления (views) в приложении. На Рисунке 24 представлена визуализация схемы маршрутизации на основе файла urls.py.

Рисунок 24 – Схемы маршрутизации home

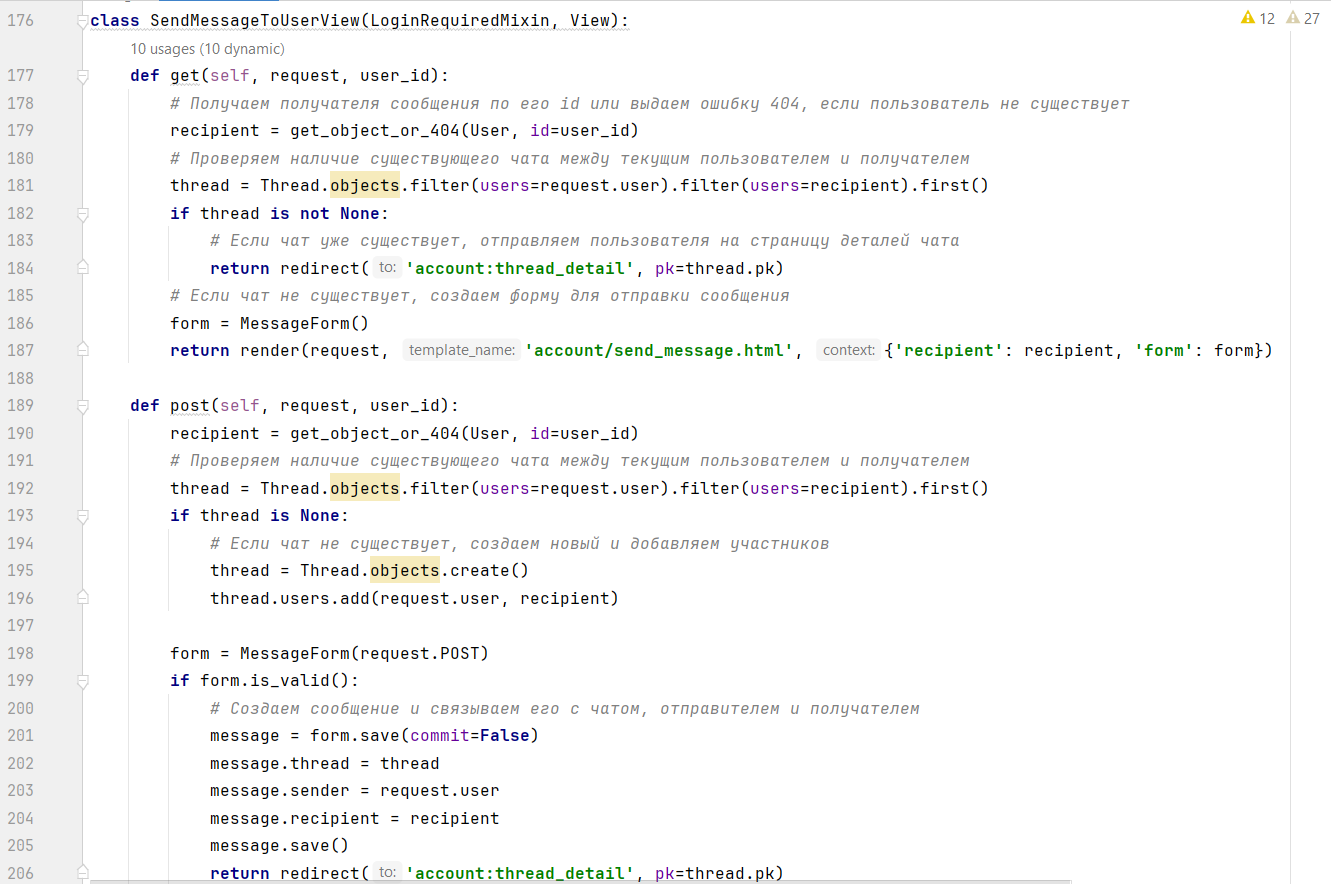
Реализация функционала авторизации и регистрации

Для управления процессом регистрации и авторизации пользователей использовались встроенные возможности Django, такие как форма регистрации и входа в систему. Рисунок 25.

Рисунок 25 – Код для регистрации пользователей

Обработка личных сообщений

Модели Thread и Message были разработаны для реализации системы личных сообщений. На Рисунке 26 приведен пример представления для отправки сообщения.

 Рисунок 26 – Код для отправки сообщения

Таким образом, написание бэкенд-логики на Python с использованием Django включало настройку проекта, создание моделей данных, представлений и маршрутов, а также реализацию функционала регистрации, авторизации и личных сообщений. Все эти шаги обеспечили надежное и эффективное функционирование серверной части социальной сети "Ganz".

## **3.4 Разработка фронтенд-части с использованием HTML, CSS, JavaScript и Bootstrap**

Фронтенд-часть социальной сети "Ganz" разработана с использованием HTML, CSS, JavaScript и Bootstrap. Эта комбинация технологий обеспечивает создание удобного, интуитивно понятного и эстетически приятного пользовательского интерфейса. В этом разделе будут основные элементы и компоненты, разработанные для фронтенд-части приложения, а также их интеграцию с бэкендом.

Основные файлы шаблонов.

В контексте разработки проекта активно применяются различные ключевые документы с расширением HTML, каждый из которых выполняет строго определённые функции в рамках создания пользовательского интерфейса. Следует отметить следующие файлы:

В пределах указанной директории — templates/404.html — заложена страница, отображающая ошибку 404. Она выполняет важную функцию информирования: пользователю становится ясно, что искомая страница отсутствует, что способствует избежанию его возможного разочарования и замешательства.

Файл templates/base.html служит фундаментальным основанием, на котором строится общая структура всех страниц проекта, включая подключение необходимых стилей и скриптов. Такая унификация дает возможность поддерживать консистентность и органичность дизайна на всех уровнях взаимодействия с пользователем.

В файле templates/messages.html реализована концепция вывода системных сообщений, что охватывает уведомления о различных событиях — от ошибок до успешных операций. Этот элемент взаимодействия предназначен для того, чтобы пользователь мог в реальном времени получать обратную связь о состоянии своих запросов и действий в системе.

Через файл templates/navbar.html предоставляется структура навигационной панели, что включает основные элементы управления для обеспечения доступа к различным секциям сайта. Этот файл играет ключевую роль в обеспечении удобства навигации по ресурсу и формировании его структуры доступности.

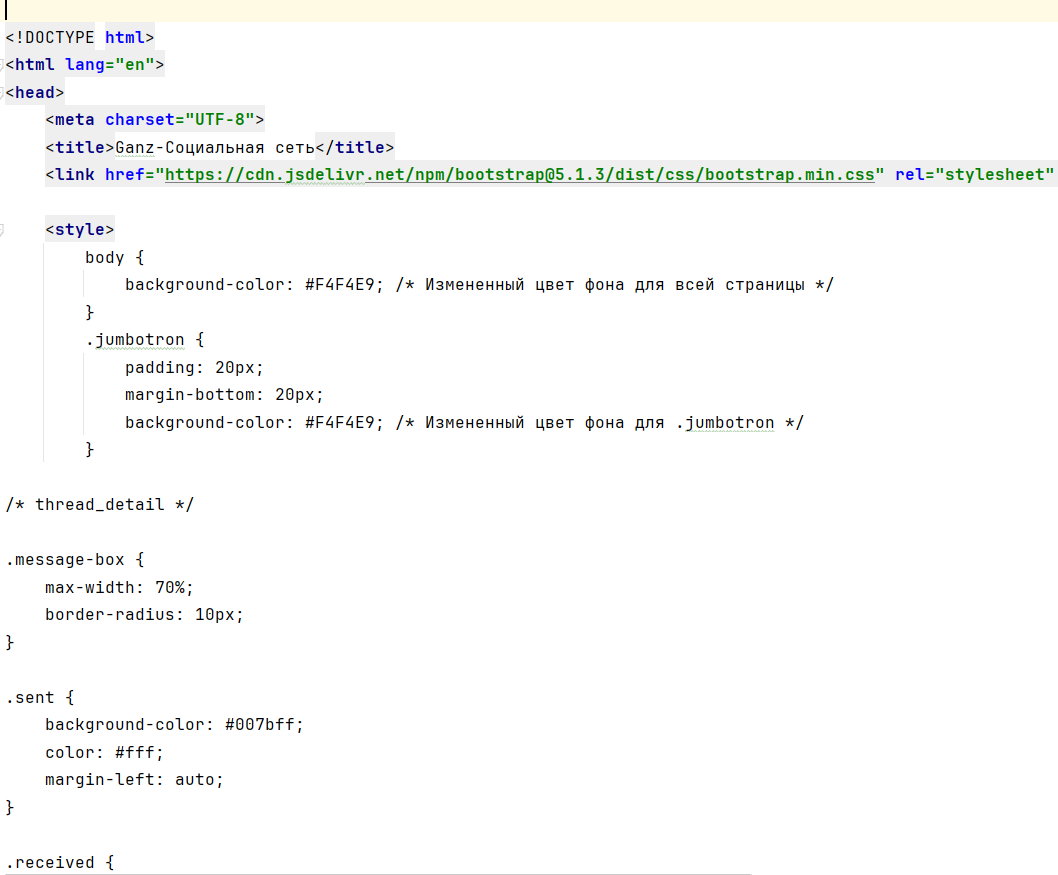
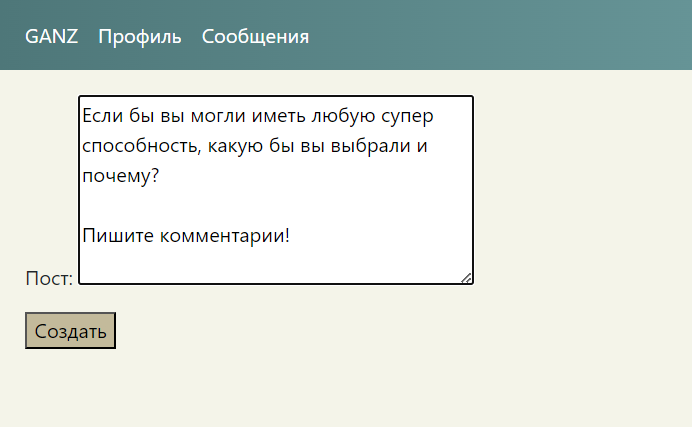
Все эти файлы вместе формируют единую систему, обеспечивающую не только структурную целостность, но и навигационную удобность интерфейса проекта, что существенно повышает его пользовательскую привлекательность и функциональную эффективность. Пример файла base.html, который служит в качестве базового шаблона, можно наблюдать на рисунке 27, иллюстрирующем его структуру и особенности применения.

Рисунок 27 - Базовый шаблон base.html

Данный документ описывает структуру файлов шаблонов, предназначенных для реализации главных страниц — интерфейс, который позволяет пользователям взаимодействовать с контентом в режиме создания, просмотра и редактирования. Например, файл home/create.html обеспечивает необходимую структуру для ввода заголовка и текста нового поста, иллюстрируя это действие с помощью Рисунка 28.

Рисунок 28 - Шаблон для создания нового поста 

В дополнение, шаблон home/detail.html функционирует как средство для детального отображения содержания поста, включая заголовок и комментарии пользователей (см. Рисунок 29). Параллельно, шаблон home/index.html служит для каталогизации и представления постов, систематизированных по дате их создания, что допускает переход к более детальному просмотру каждого из них; данная функция воплощена в Рисунке 30.

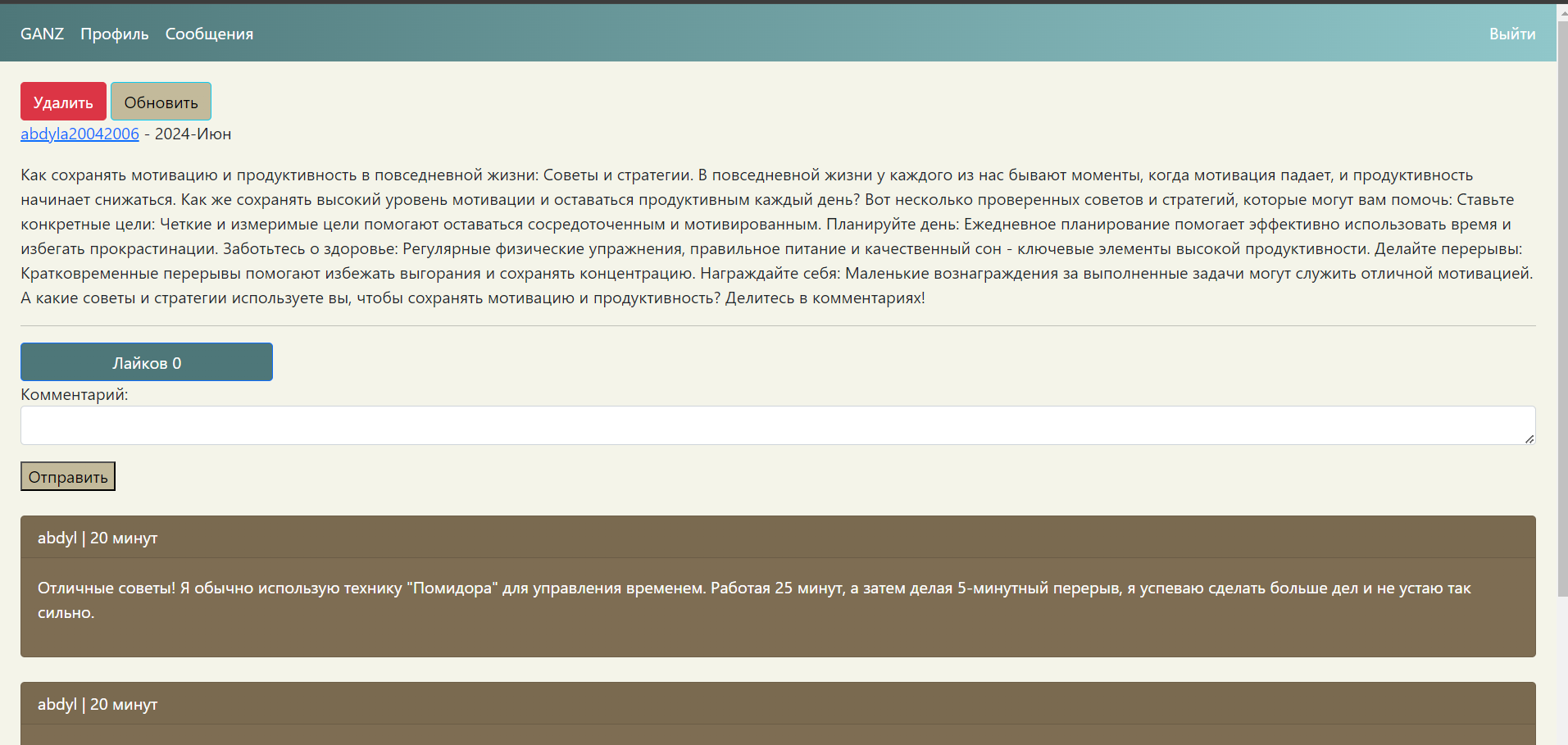
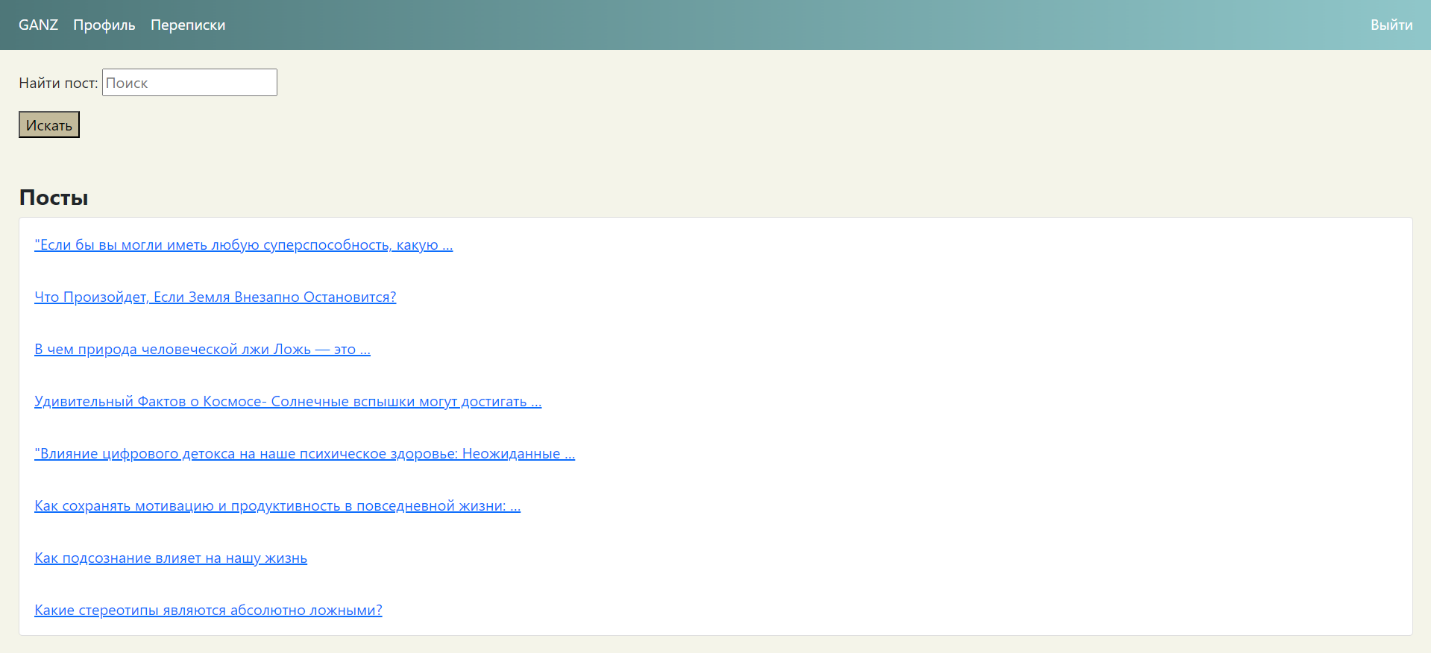


Рисунок 29 - Шаблон для детального просмотра поста

Рисунок 30 – Шаблон для отображения списка постов 

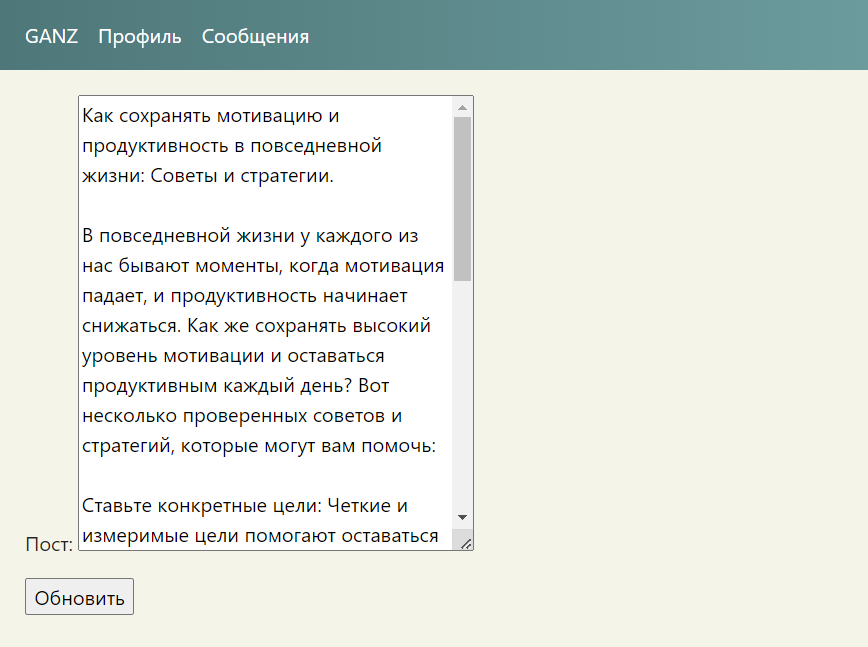
Кроме того, home/update.html представляет собой инструментарий для модификации уже существующих постов, предоставляя пользователю форму для корректировки заголовка и текста, что демонстрируется в Рисунке 31.

Рисунок 31 – Шаблон для обновления поста

Эти шаблоны обеспечивают интерфейс для работы с постами, позволяя пользователям создавать, просматривать и редактировать контент.

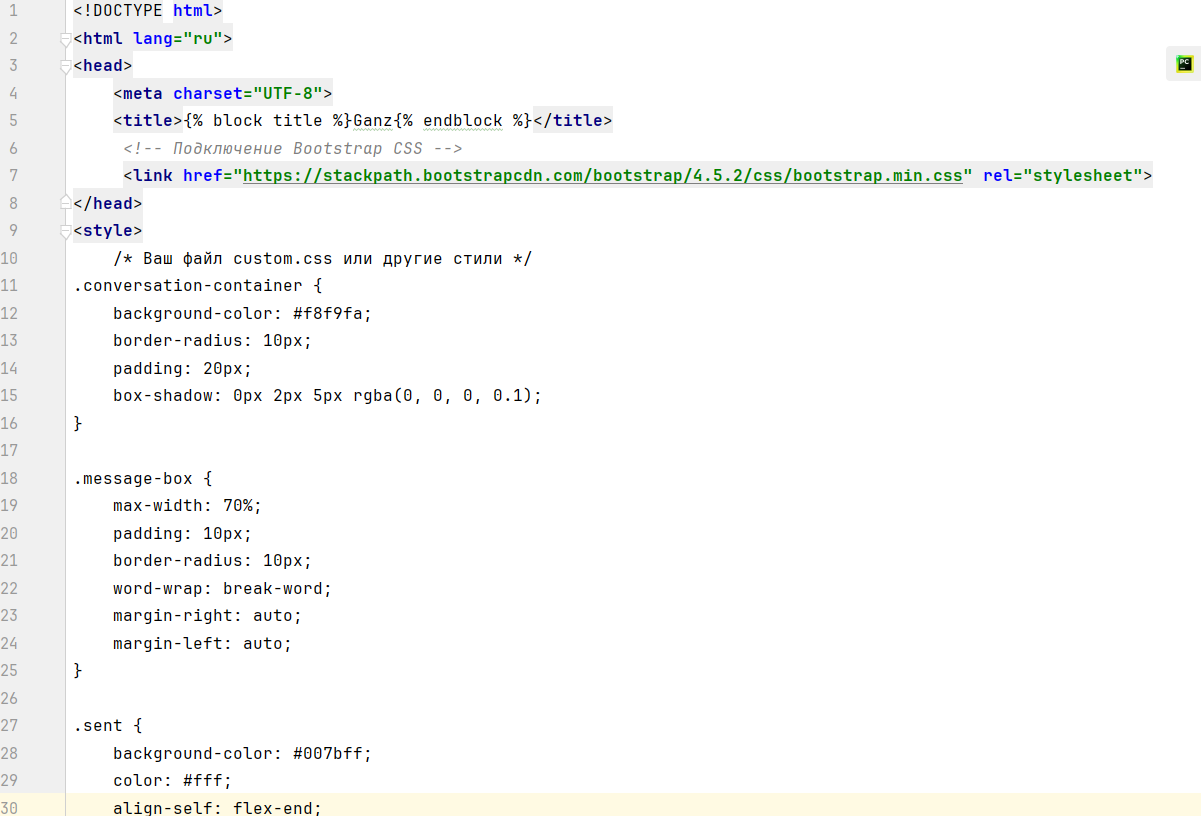
Таким образом, представленные шаблоны формируют структурную основу, которая допускает создание, визуализацию и корректировку контента в рамках определённой платформы. Переходя к управлению аккаунтом, шаблон account/base.html определяет базовую архитектуру для всех страниц, связанных с аккаунтом пользователя, включая его визуальное представление в Рисунке 32.

Рисунок 32 – Код для базового шаблона для страниц аккаунта

Шаблон account/thread\_detail\_ajax.html, предназначенный для детального просмотра потока сообщений с применением AJAX, обеспечивает динамическое обновление данных без необходимости перезагрузки страницы, как это отображено в Рисунке 33. Шаблон account/edit\_profile.html предлагает инструменты для изменения личных данных пользователя, что визуализировано в Рисунке 34. Подобно этому, account/thread\_detail.html и account/thread\_list.html функционируют как средства для отображения диалогов и переписки, обеспечивая пользователю возможность взаимодействия с активными потоками сообщений (см. Рисунки 35 и 36 соответственно).



Рисунок 33 – Код шаблона для динамического обновления сообщений

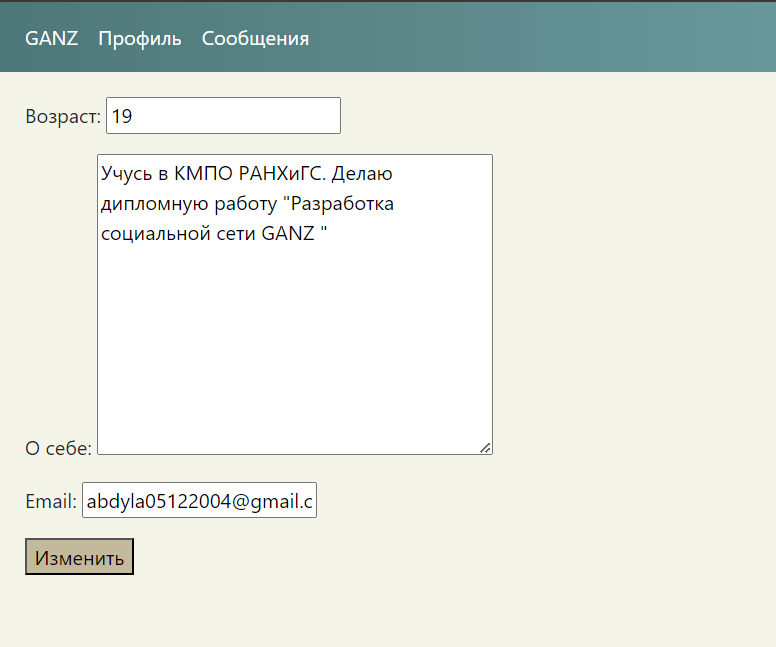


Рисунок 34 – Шаблон для редактирования профиля

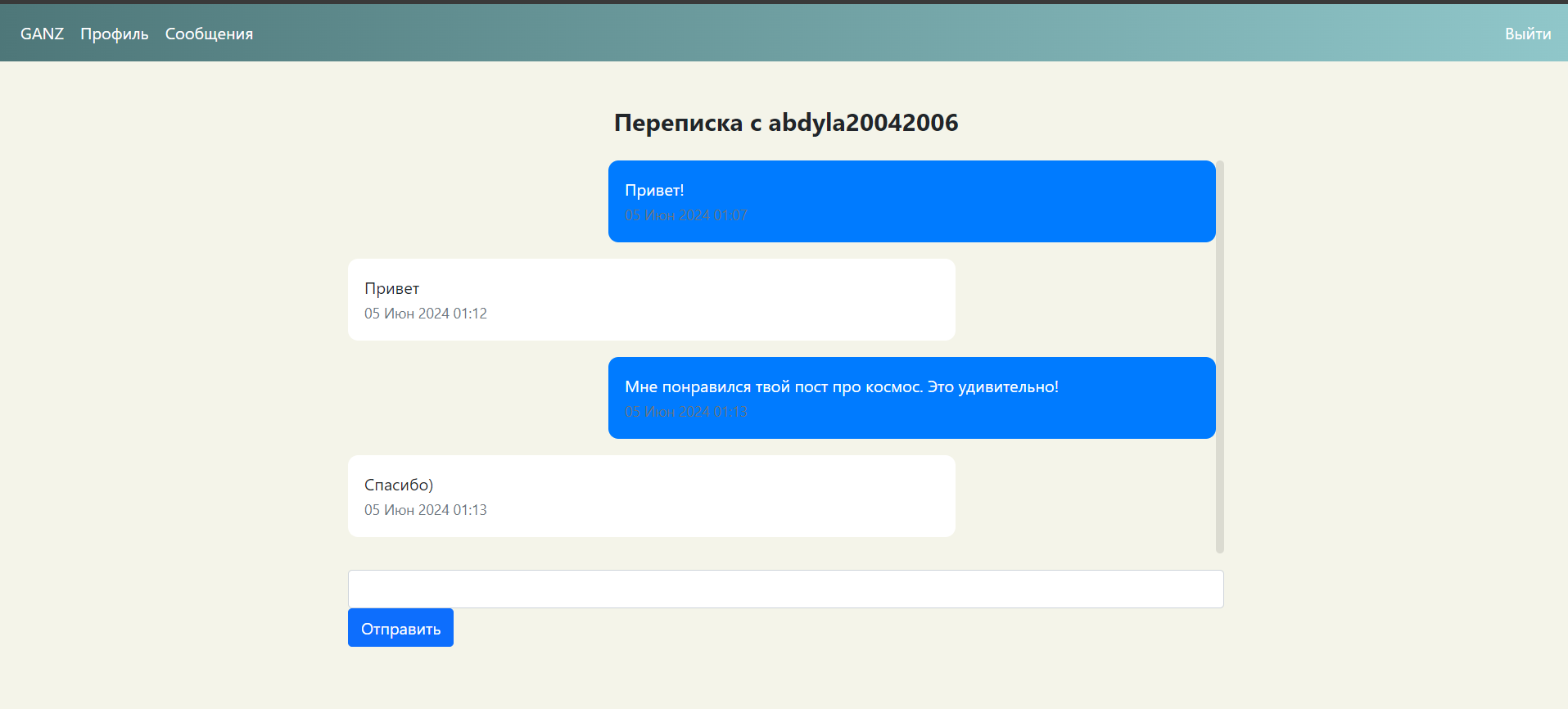


Рисунок 35 – Шаблон потока сообщений



Рисунок 36 – Шаблон для отображения списка диалогов

Завершающие шаблоны, account/register.html и account/login.html, предоставляют структурированные формы для регистрации новых пользователей и входа в систему, как это показано в Рисунках 37 и 38. Шаблон account/send\_message.html и account/profile.html облегчают обмен сообщениями и просмотр профильной информации, демонстрируемые в Рисунках 39 и 40.

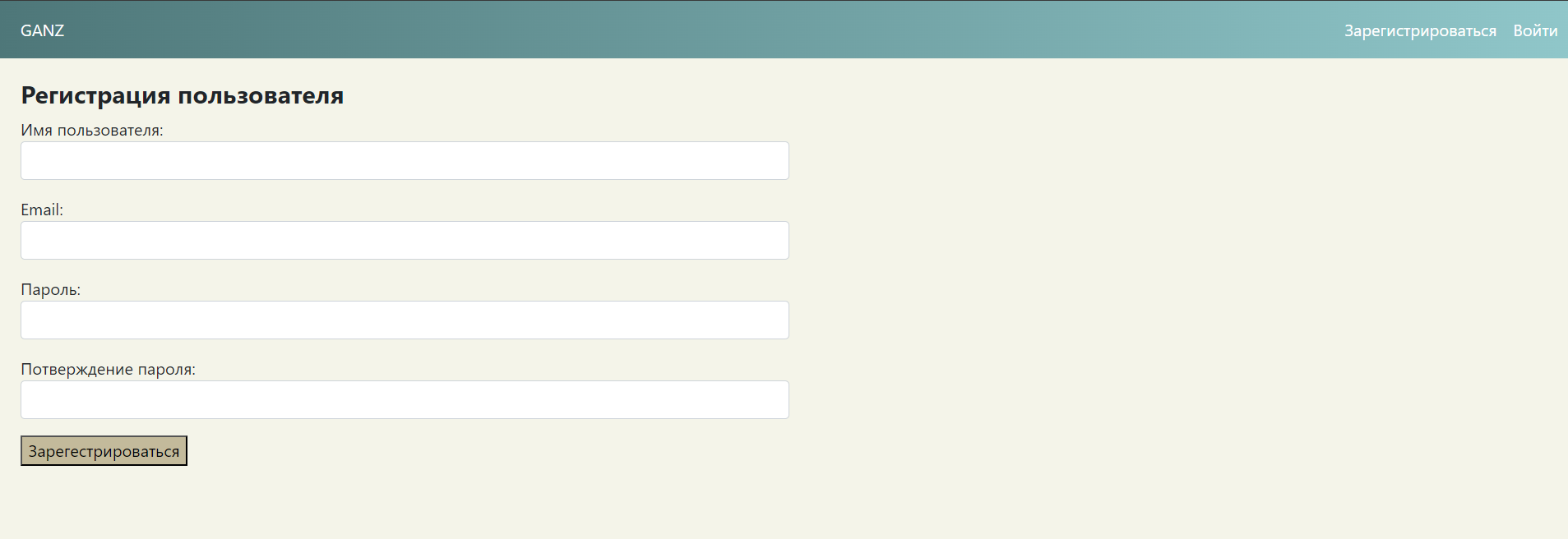


Рисунок 37 – Шаблон для регистрации нового пользователя

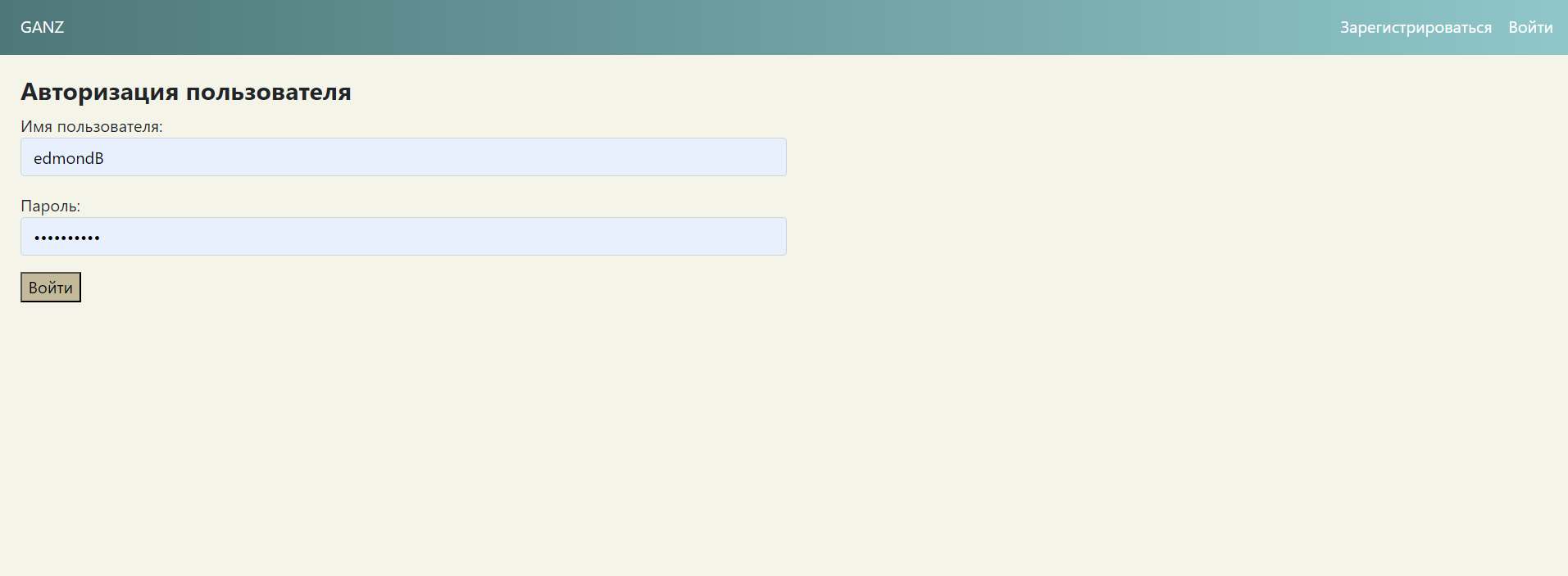


Рисунок 38 – Шаблон для страницы входа

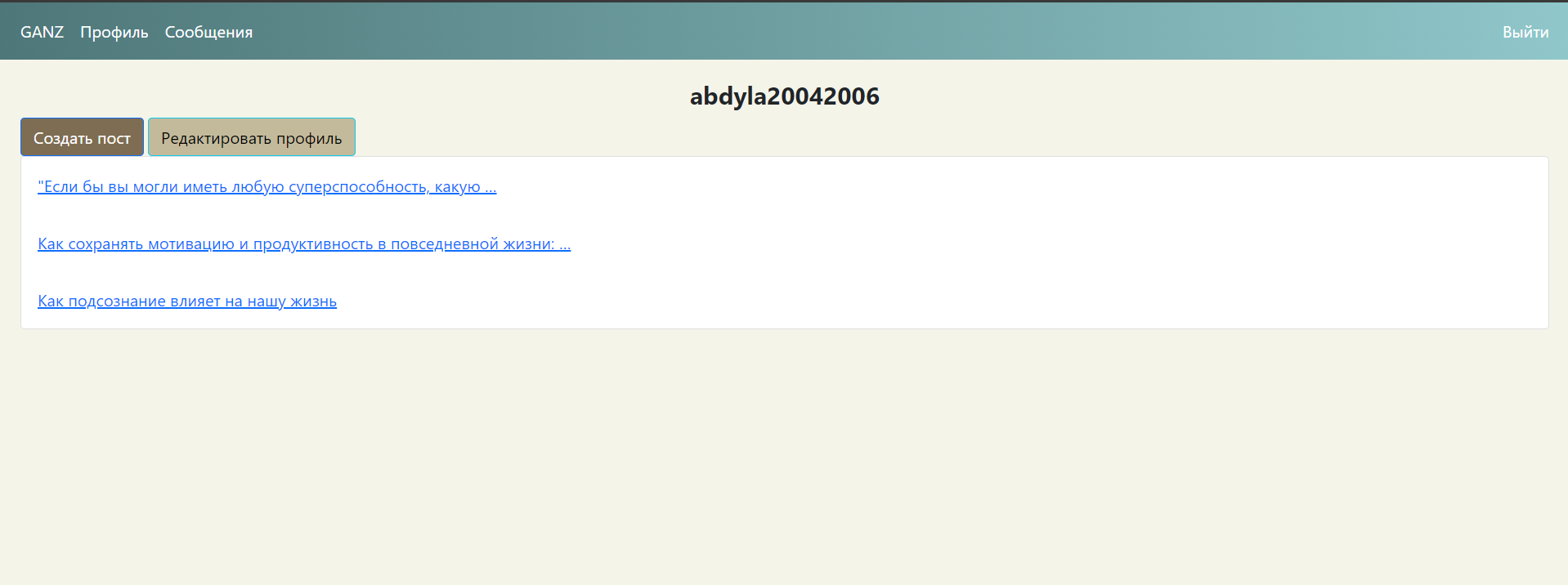


Рисунок 39 – Шаблон для просмотра своего профиля

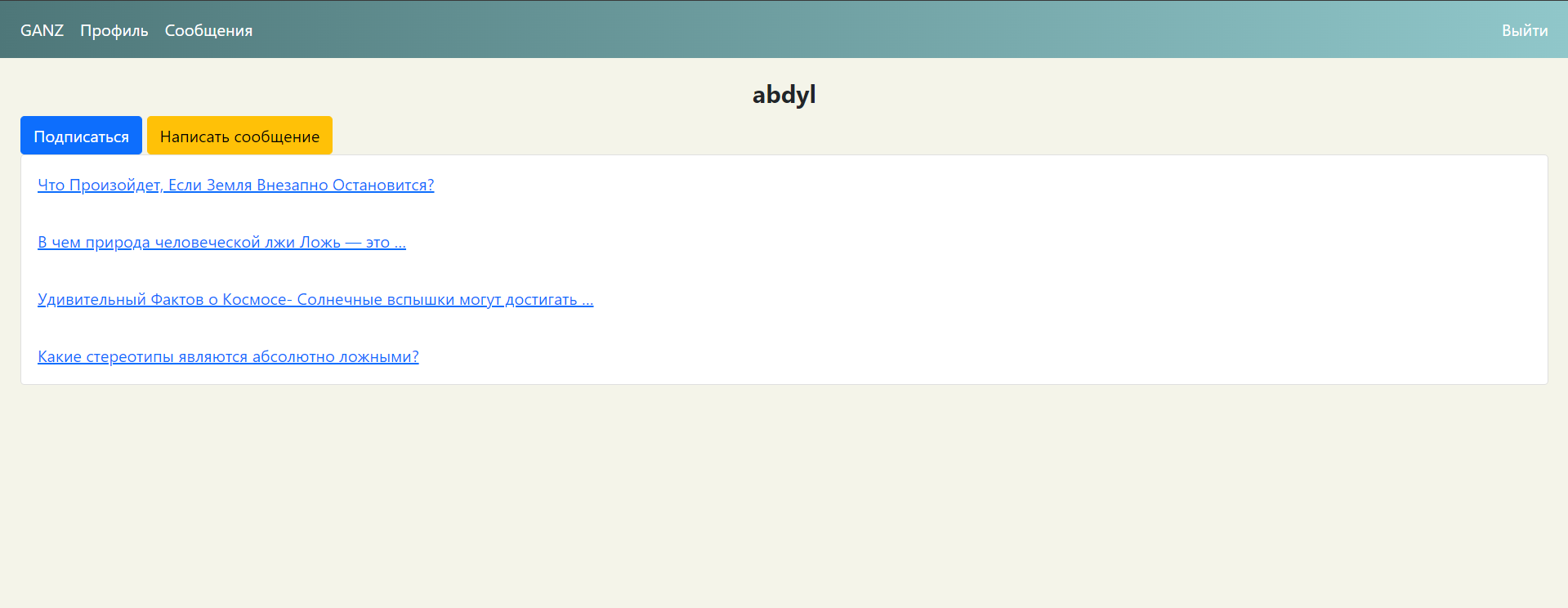
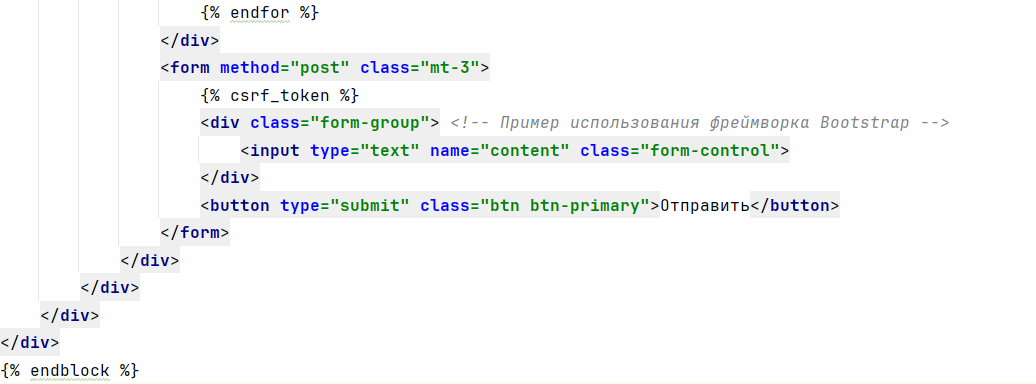


Рисунок 40 – Шаблон для просмотра профиля пользователя

Такова структура шаблонов, участвующих в процессе управления аккаунтом и взаимодействия пользователей между собой на платформе, обеспечивая эффективное и эргономичное использование системы.

Использование Bootstrap и jQuery.

Для создания адаптивного и привлекательного интерфейса используется фреймворк Bootstrap. Он позволяет быстро разрабатывать интерфейсы с использованием готовых компонентов и стилей. jQuery используется для упрощения работы с JavaScript и выполнения AJAX-запросов. На Рисунке 41 приведен пример использования фреймворка Bootstrap.

Рисунок 41 – Пример кода использования Bootstrap

Интеграция с бэкендом.

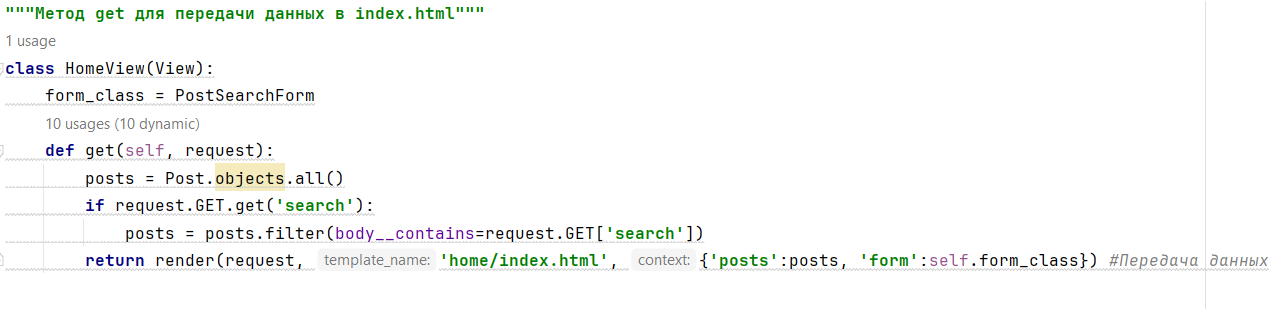
Фронтенд-часть интегрируется с бэкендом через представления (views), которые возвращают шаблоны с данными. Например, для отображения списка постов используется представление, которое передает данные постов в шаблон index.html. Рисунок 42.

Рисунок 42 – Код передачи данных в index.html

Таким образом, фронтенд-часть социальной сети "Ganz" разработана с использованием современных веб-технологий, обеспечивая удобный и интуитивно понятный интерфейс для пользователей. Использование Bootstrap и jQuery позволяет быстро создавать адаптивные и динамические интерфейсы, а интеграция с бэкендом на Django обеспечивает взаимодействие с данными и функциональностью приложения.

# **4 ТЕСТИРОВАНИЕ И ОТЛАДКА**

Тестирование и отладка являются ключевыми этапами разработки любого программного обеспечения, особенно социально значимого, как наша социальная сеть "Ganz". В этом разделе мы рассмотрим процессы функционального тестирования, проведенные для обеспечения корректной работы всех компонентов приложения. Функциональное тестирование было реализовано с помощью встроенных возможностей фреймворка Django.

## **4.1 Проведение функционального тестирования**

Функциональное тестирование направлено на проверку соответствия поведения системы её функциональным требованиям. В данном проекте тесты были сгруппированы по основным функциям приложения: регистрация и аутентификация пользователей, создание постов и комментариев.

Класс тестирования UserRegistrationTest

Этот класс тестирует процесс регистрации нового пользователя. В данном тесте используется метод self.client.post, который отправляет POST-запрос на URL, соответствующий маршруту регистрации пользователя. Данные для регистрации включают имя пользователя, адрес электронной почты и пароль. Рисунок 43.



Рисунок 43 – Класс тестирования UserRegistrationTest

Этот тест проверяет два ключевых аспекта процесса регистрации:

—Успешность выполнения POST-запроса на регистрацию (ожидаемый статус-код 302, указывающий на перенаправление).

—Создание нового пользователя в базе данных (путем проверки наличия пользователя с заданным именем).

В случае успешного выполнения всех проверок можно быть уверенным, что функциональность регистрации работает корректно и пользователи могут успешно регистрироваться в системе.

Класс тестирования UserLoginTest

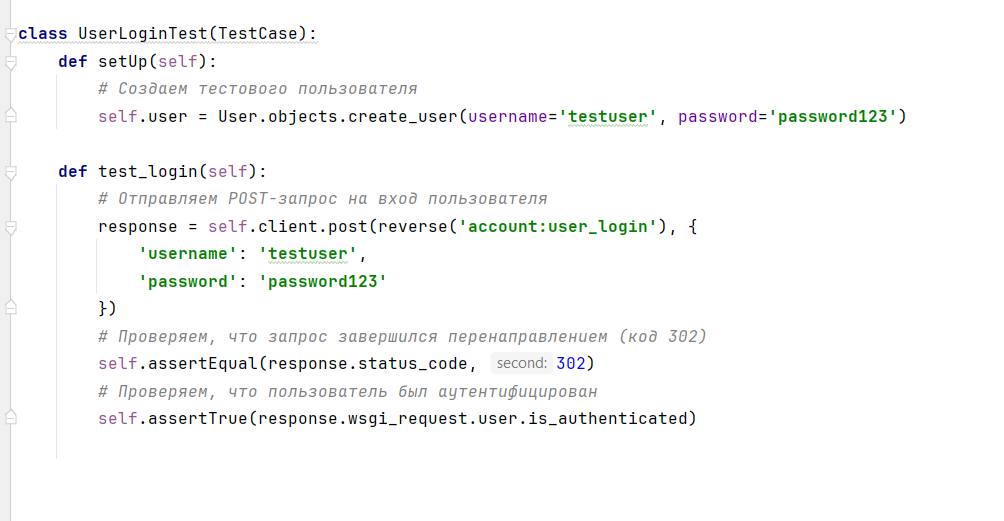
Этот класс тестирует процесс входа в систему для существующего пользователя. В методе setUp создается тестовый пользователь, который будет использоваться для проверки функциональности входа. Рисунок 44

Рисунок 44 – Класс тестирования UserLoginTest

Этот тест проверяет:

—Успешность выполнения POST-запроса на вход (ожидаемый статус-код 302, указывающий на перенаправление).

—Аутентификацию пользователя (путем проверки, что пользователь был успешно аутентифицирован и текущий пользователь wsgi\_request.user является аутентифицированным).

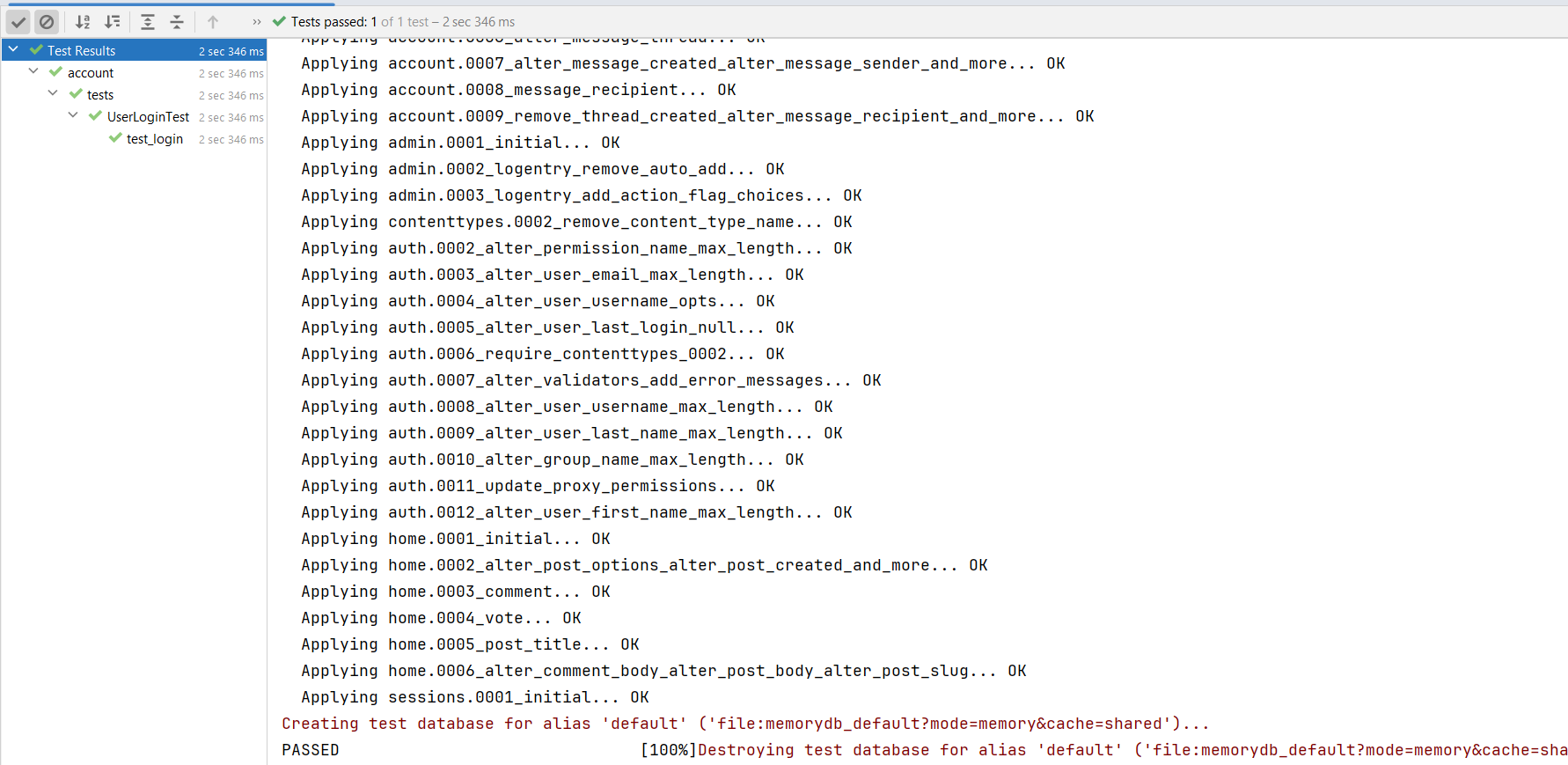
Успешное прохождение этих тестов подтверждает корректность реализации механизмов аутентификации и регистрации что является критически важным для безопасности и функциональности приложения. Рисунок 45-46.

Рисунок 45 – Успешное прохождение тестов

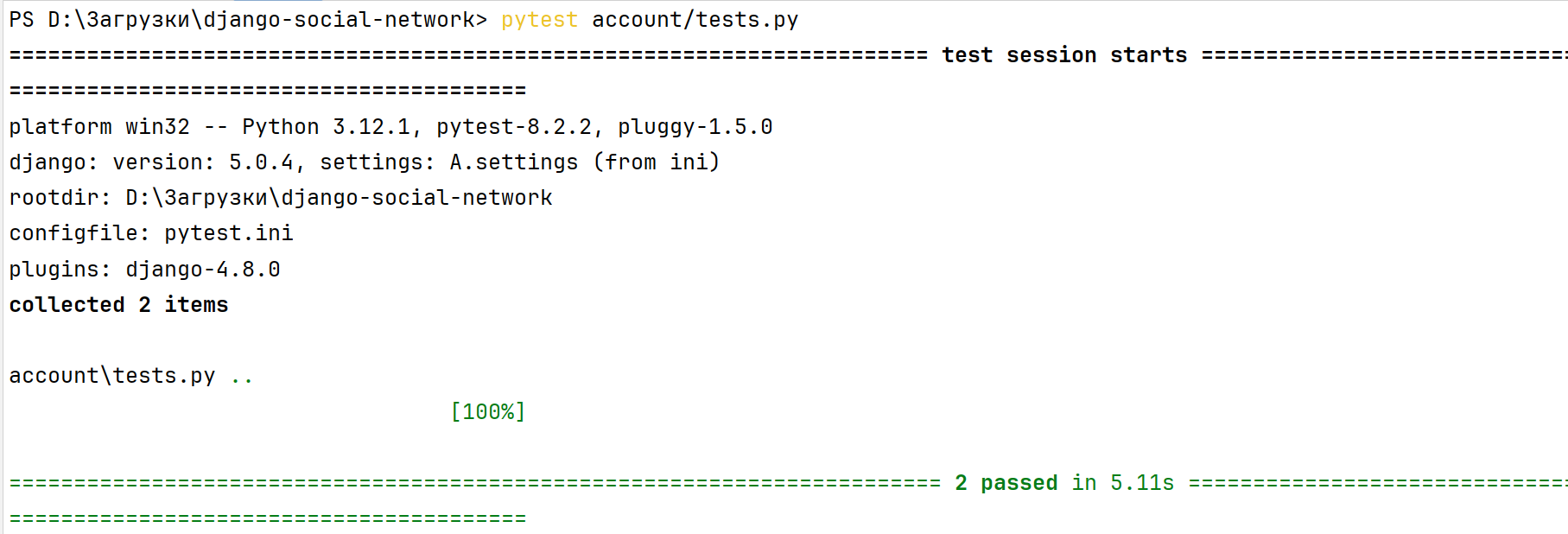


Рисунок 46 – Успешное прохождение тестов

Класс тестирования PostCreationTest

Этот класс тестирует процесс создания нового поста. В методе setUp создается тестовый пользователь, и выполняется его вход в систему, что необходимо для проверки создания поста. Рисунок 47.

Рисунок 47 – Класс тестирования PostCreationTest

Этот тест проверяет:

—Успешность выполнения POST-запроса на создание поста (ожидаемый статус-код 302, указывающий на перенаправление).

—Создание нового поста в базе данных (путем проверки наличия поста с заданным содержанием).

Класс тестирования CommentCreationTest

Этот класс тестирует процесс создания комментария к посту. В методе setUp создается тестовый пользователь, пост и начальный комментарий, что необходимо для проверки функциональности добавления комментариев. Рисунок 48.

Рисунок 48 – Класс тестирования CommentCreationTest

Этот тест проверяет:

—Успешность выполнения POST-запроса на создание комментария (ожидаемый статус-код 302, указывающий на перенаправление).

—Создание нового комментария в базе данных (путем проверки наличия комментария с заданным содержанием).

Успешное выполнение всех проверок подтверждает, что пользователи могут корректно создавать посты и добавлять комментарии к постам, что является важной функцией для взаимодействия и обмена мнениями на платформе. Рисунок 49-50.

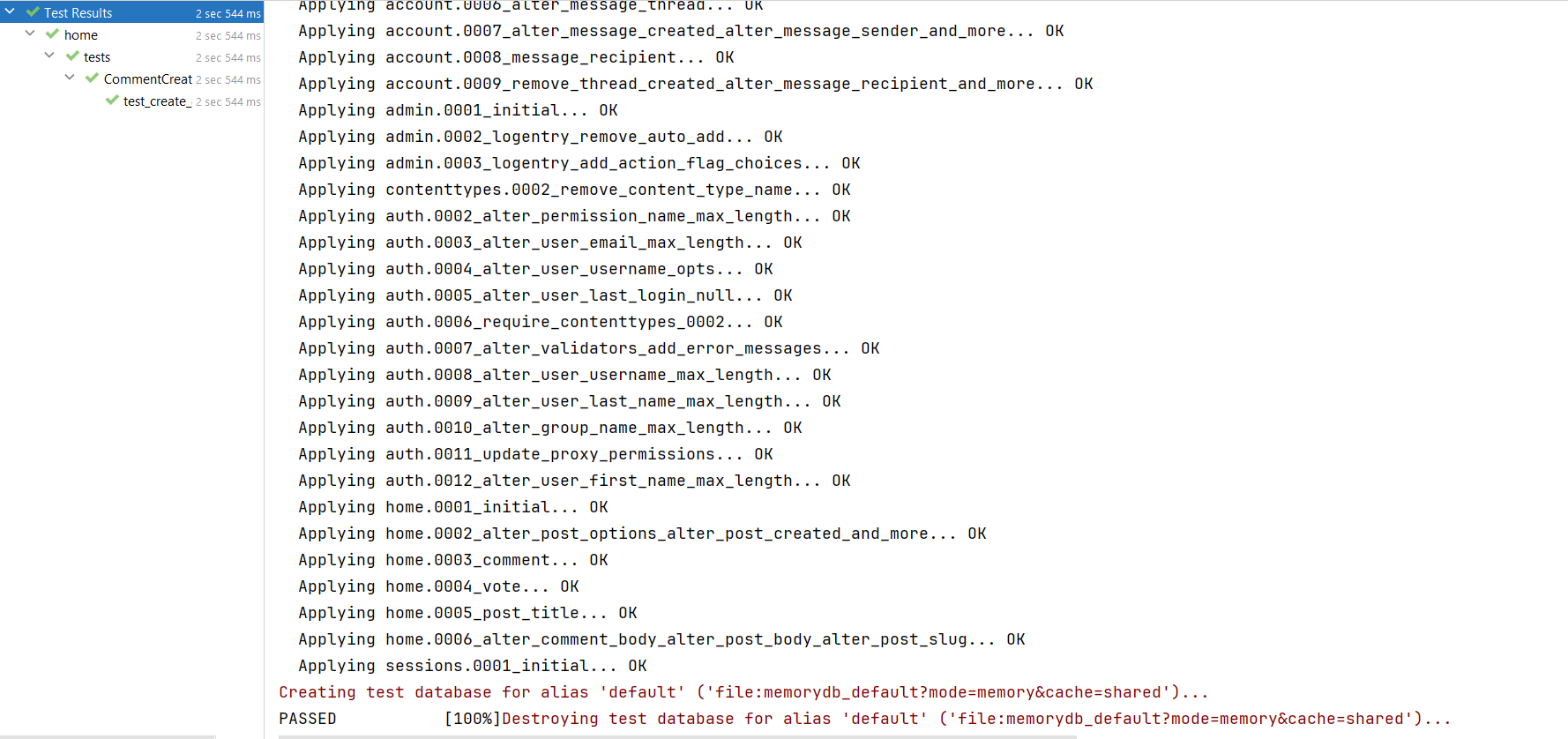


Рисунок 49 – Успешное прохождение тестов

  
Рисунок 50 – Успешное прохождение тестов

Таким образом, проведенное функциональное тестирование обеспечивает уверенность в том, что основные функции приложения работают корректно и соответствуют требованиям. Эти тесты помогают выявлять и устранять ошибки на ранних этапах разработки, что существенно улучшает качество и надежность конечного продукта.

## **4.2 Отладка ошибок и недочетов**

Отладка является неотъемлемой частью процесса разработки программного обеспечения. Она включает в себя выявление, анализ и устранение ошибок, а также оптимизацию кода для повышения его производительности и надежности. В этом разделе рассмотрим основные шаги и методы, которые были использованы для отладки приложения.

Методика отладки

Для отладки использовались следующие методики и инструменты:

—Логирование: Использование встроенной библиотеки Python для логирования (logging) для отслеживания состояния приложения и выявления ошибок в различных частях кода.

—Интерактивная отладка: Использование встроенного отладчика PyCharm, который позволяет пошагово выполнять код, устанавливать точки останова и инспектировать состояние переменных.

—Тестирование кода: Регулярное выполнение тестов, описанных в разделе 4.1, для проверки работоспособности отдельных компонентов и всего приложения в целом.

—Просмотр журналов ошибок сервера: Анализ логов веб-сервера для выявления ошибок, происходящих на стороне сервера.

Примеры отладки:

Ошибка при регистрации пользователя

Одной из первых обнаруженных ошибок была проблема с регистрацией пользователей. Тесты показали, что некоторые пользователи не могли зарегистрироваться из-за несоответствия паролей. Для решения этой проблемы был добавлен дополнительный шаг проверки паролей перед сохранением пользователя в базу данных. Рисунок 51.



Рисунок 51 – Код проверки паролей

Ошибка аутентификации

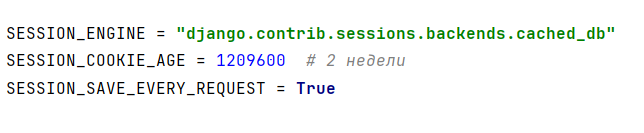
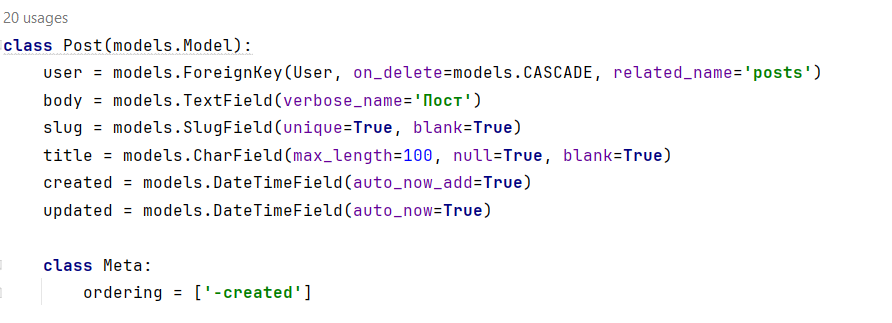
В процессе тестирования логина пользователей было обнаружено, что некоторые пользователи не могут войти в систему даже при вводе правильных учетных данных. Эта ошибка была связана с неверным конфигурированием сессий. После детального анализа и проверки логов, проблема была решена путем обновления настроек сессий в файле конфигурации Django. Рисунок 52.

Рисунок 52 – Код настройки сессии

Проблемы с созданием постов

При тестировании создания постов было выявлено, что некоторые посты не отображались на главной странице. Ошибка была связана с некорректной сортировкой постов в базе данных. Для решения этой проблемы был изменен запрос на получение постов, чтобы они сортировались по дате создания. Рисунок 53

Рисунок 53 – Запрос по дате

Ошибка создания комментариев

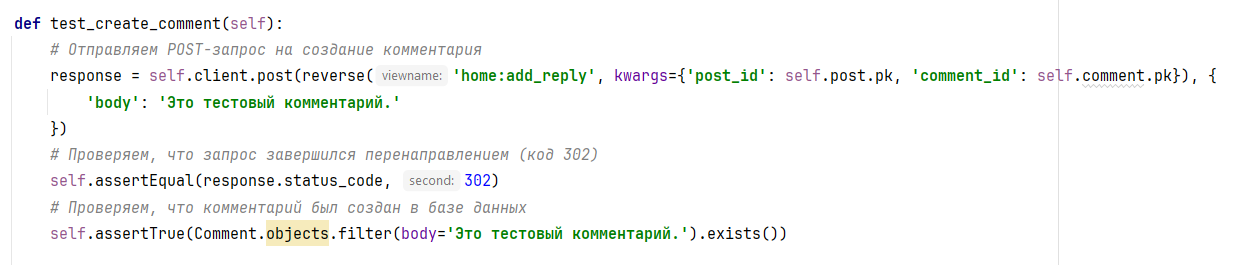
Во время тестирования функции добавления комментариев была обнаружена ошибка, приводящая к сбоям при попытке ответить на комментарий. Проблема была связана с неправильной передачей данных о родительском комментарии. Ошибка была исправлена путем добавления правильных параметров в запрос. Рисунок 54.

Рисунок 54 – Исправление параметров для запроса

Отладка является важнейшим этапом в процессе разработки, который позволяет обеспечить высокое качество и надежность программного обеспечения. Благодаря тщательному тестированию и систематическому подходу к выявлению и исправлению ошибок, удалось создать стабильное и функционально полное приложение.

**5. ТЕХНИКО-ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ**

1. Определение времени на разработку и комплексные испытания программного обеспечения:

1.1. Время на разработку программного обеспечения

где КДп – длительность разработки, согласованная с заказчиком в днях,

Вр – длительность рабочего дня в часах (Вр = 8 часов);

1.2. Время на комплексные испытания программного обеспечения.

Срок опытной эксплуатации (КДоэ) устанавливается в размере 50 % от длительности разработки, согласованной с заказчиком);

где КДоэ – длительность опытной эксплуатации в днях,

Н – норматив трудоемкости при проведении опытной эксплуатации, в часах (Н = 0,0004 часа);

1.3. Общее время на разработку и комплексные испытания программного обеспечения:

где Тп - время на разработку программного обеспечения в часах;

Тоэ - время на комплексные испытания программного обеспечения

2. Расчет фонда оплаты труда за разработку и комплексные испытания программного обеспечения.

2.1. Расчет оплаты труда за разработку программного обеспечения.

где Тп – время на разработку программного обеспечения, час.

СЧп – стоимость часа работы программиста, руб. (Чп: от 500 руб. до 800 руб. за один час)

2.2.Расчет оплаты труда за проведение опытной эксплуатации программной системы.

где Тоэ – время на комплексные испытания программного обеспечения, в

часах

СЧоэ – стоимость часа работы тестировщика, руб. (Чоэ: от 350 руб. до 600 руб. за один час)

2.3. Оплата труда за разработку и комплексные испытания программного обеспечения

где ЗПп – оплаты труда за разработку программного обеспечения, руб.

ЗПоэ – оплаты труда за проведение опытной эксплуатации программной системы.

2.4. Расчет страховых взносов от оплаты труда за разработку и комплексные испытания программного обеспечения

где СВзп – сумма страховых взносов от заработной платы, в рублях.

%Озп - процент отчислений от заработной платы (%Озп = 30 %).

2.5. Фонд оплаты труда за разработку и комплексные испытания программного обеспечения.

где ЗПобщ - Оплата труда за разработку и комплексные испытания программного обеспечения

СВзп – сумма страховых взносов от заработной платы, в рублях.

3. Расчет накладных и прочих расходов при разработке и комплексных испытаниях программного обеспечения.

3.1. Расчет накладных расходов.

К накладным расходам относятся расходы на управление подразделением, содержание и ремонт зданий, оборудования, коммунальные услуги и т. п.

Накладные расходы (%Нр) принимаются в размере 40% от оплаты труда за разработку и комплексные испытания программного обеспечения (ЗПобщ).

3.2. Расчет прочих затрат организации.

Прочие затраты (%Пз) составляют 25% от оплаты труда за разработку и комплексные испытания программного обеспечения (ЗПобщ).

4. Составление калькуляции разработки и комплексных испытаний программного обеспечения.

Таблица 2- Калькуляция себестоимости разработки и комплексных испытаний программного обеспечения.

|  |  |  |
| --- | --- | --- |
| №  п/п | Наименование статьи расходов | Сумма, руб. |
| 1 | Фонд оплаты труда за разработку и комплексные испытания программного обеспечения () | 156.003,9 |
| 2 | Накладные расходы () | 48.001,2 |
| 3 | Прочие затраты () | 30.000,75 |
| 4 | Затраты на проведение работ (Затр) | ∑ 234.005,85 |

5. Расчет финансовых показателей разработки и комплексного испытания программного обеспечения.

5.1. Расчет прибыли за выполнение работ:

Пр = 234.005,85 × 30% = 70.201,75 руб.

Процент прибыли (%Пр.) составляет от 20 до 30% от затрат на проведение работ (Затр.)

5.2. Налог на прибыль

Нпр = 70.201,75 × 20% = 14.040,35 руб.

где: Снп – ставка налога на прибыль (Снп = 20%).

5.3. Чистая прибыль

Пч = 70.201,75 – 14.040,35= 56.161,4 руб.

где: Пр. – сумма прибыли за выполнение работ, руб.

Нпр. – сумма налога на прибыль, руб.

5.4. Цена выполненной работы:

Цр = 234.005,85 + 70.201,75 = 304.207,6 руб.

где: Затр. – затраты на проведение работ, руб.

Пр – сумма прибыли за выполнение работ, руб.

6. Финансово-экономические показатели выполнения работ

Таблица 3 - Финансово-экономические показатели выполненных работ

|  |  |  |
| --- | --- | --- |
| Наименование | Единица измерения | Значения показателя |
| Общее время на разработку и комплексные испытания программного обеспечения (Т) | час. | 240,006 |
| Затраты на проведение работ | руб. | 234.005,85 |
| Процент прибыли при реализации проекта | % | 30 |
| Чистая прибыль | руб. | 56.161,4 |
| Цена выполненной работы | руб. | 304.207,6 |

# **ЗАКЛЮЧЕНИЕ**

Инициирование проекта по разработке социальной сети с использованием фреймворка Django представляет собой заметный шаг в области усовершенствования интерактивности между пользователями; дополнительно, это способствует повышению уровня их онлайн-опыта. В процессе реализации проекта осуществлена всесторонняя разработка системы, адаптированной под функции регистрации, аутентификации участников, администрирования профилей, создания и комментирования постов, обмена сообщениями.

Освещение потенциалов Django, как высокоэффективного инструмента для создания веб-приложений, стало одним из результатов проекта. Применение данного фреймворка оказало существенную поддержку в организации структуры базы данных, обеспечении конфиденциальности пользовательских данных, реализации надежной системы управления аккаунтами. Совместно с этим, использование передовых технологий вроде HTML, CSS, JavaScript, Bootstrap способствовало созданию интуитивно понятного и удобного пользовательского интерфейса.

В ходе разработки уделено пристальное внимание детальному тестированию, дебаггингу системы, что позволило выявить и устранить множество ошибок, тем самым повысив уровень надежности и стабильности приложения. Внедрение системы логирования и интерактивной отладки способствовало более детальному выявлению и решению возникающих проблем.

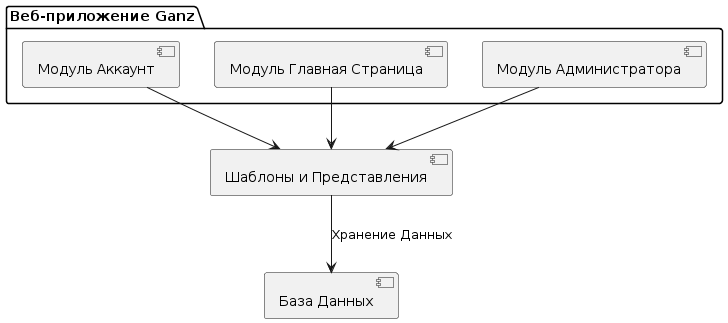
Разработка данной социальной сети на базе Django открывает перед проектом множество возможностей для будущего развития. Среди возможных направлений развития выделяются: расширение функциональности, оптимизация производительности и масштабирование приложения для обслуживания большего числа пользователей. Такие действия способствуют созданию более совершенной и многофункциональной платформы для общения пользователей.

Таким образом, создание социальной сети на базе Django обозначает важный прогресс в сфере веб-разработок, демонстрирует применение современных технологий и открывает потенциал для создания гибких, функциональных веб-приложений, отвечающих запросам пользователей. Данный проект способствует не только улучшению пользовательского взаимодействия, но и развитию профессиональных навыков команды разработчиков в области проектирования, программирования и тестирования веб-приложений, что становится значительным этапом в их профессиональном становлении.

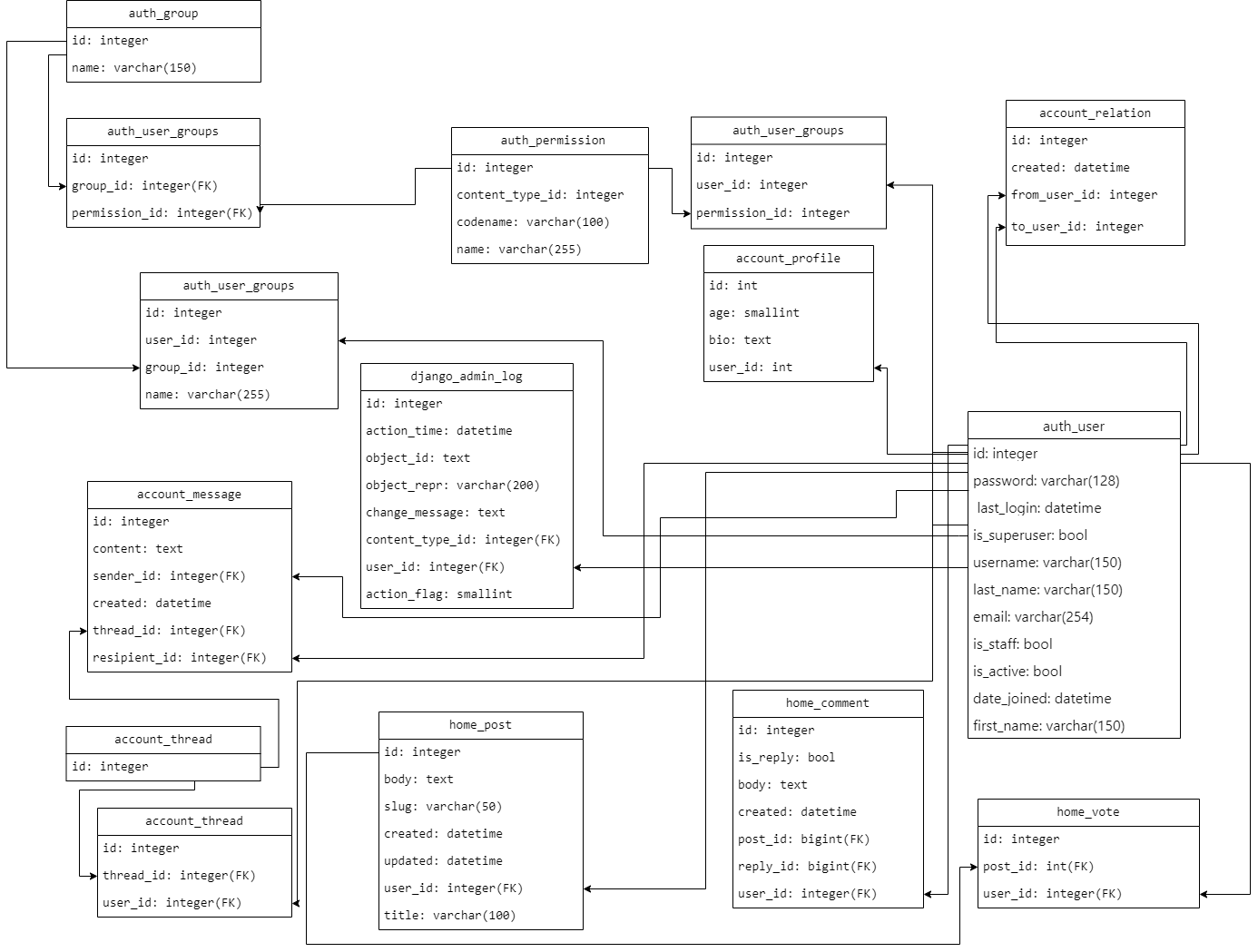
**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. ГОСТ 7.1-2003. СИБИД. Библиографическая запись. Библиографическое описание. Общие требования и правила составления.
2. ГОСТ 7.0.5-2008. СИБИД. Библиографическая запись. Аббревиатуры слов и словосочетаний на русском языке. Общие требования и правила.
3. ГОСТ 34.601-90. Автоматизированные системы. Стадии создания.
4. Лайтфут, Дж. Проектирование баз данных с примерами на SQL / Дж. Лайтфут и Э. Морисси. — М.: Диалектика, 2020. — 500 с.
5. Мюн, Эндрю К. Изучаем SQL. Самоучитель / Эндрю К. Мюн, Фортинбоус Чамберлин. — СПб.: Питер, 2021. — 400 с.
6. Браунли, Джейсон. Python для сложных задач: наука о данных, машинное обучение и сетевые приложения / Джейсон Браунли. — М.: Вильямс, 2019. — 550 с.
7. Дюбуа, Пол. MySQL. Оптимизация производительности / Пол Дюбуа, Стивен Фейерстейн. — СПб.: БХВ-Петербург, 2018. — 450 с.
8. Лутц, Марк. Программирование на Python / Марк Лутц. — М.: Эксмо, 2019. — 600 с.
9. Python: Официальный сайт Python: https://www.python.org/. Документация Python: https://docs.python.org/.
10. Django: Официальная документация: https://docs.djangoproject.com/en/stable/.
11. Bootstrap: Официальная документация: https://getbootstrap.com/docs/.
12. JavaScript: MDN Web Docs: https://developer.mozilla.org/ru/docs/Web/JavaScript.
13. jQuery: Официальная документация: https://api.jquery.com/.
14. Реализация REST API с использованием Django Rest Framework: https://www.django-rest-framework.org/.
15. SQLite: Официальная документация: https://www.sqlite.org/docs.html.
16. Учебный курс "Веб-разработка с Django" на платформе Coursera: https://www.coursera.org/learn/django-web-development.
17. Учебный курс "Введение в JavaScript" на платформе Codecademy: https://www.codecademy.com/learn/introduction-to-javascript.
18. Учебный курс "Основы HTML и CSS" на платформе FreeCodeCamp: https://www.freecodecamp.org/learn.
19. Python и Django: руководство по созданию веб-приложений: https://realpython.com/tutorials/django/.
20. Статья "Создание пользовательских форм в Django" на портале Habr: https://habr.com/ru/post/469011/.
21. Документация по Git: https://git-scm.com/doc.
22. Учебный курс "Основы работы с Bootstrap" на платформе Udemy: https://www.udemy.com/course/bootstrap-basics/.
23. Сайт "MDN Web Docs" - ресурсы для веб-разработки: https://developer.mozilla.org/.
24. Книга "Django for Beginners: Build websites with Python and Django" - William S. Vincent, 2018.

# **ПРИЛОЖЕНИЕ А**

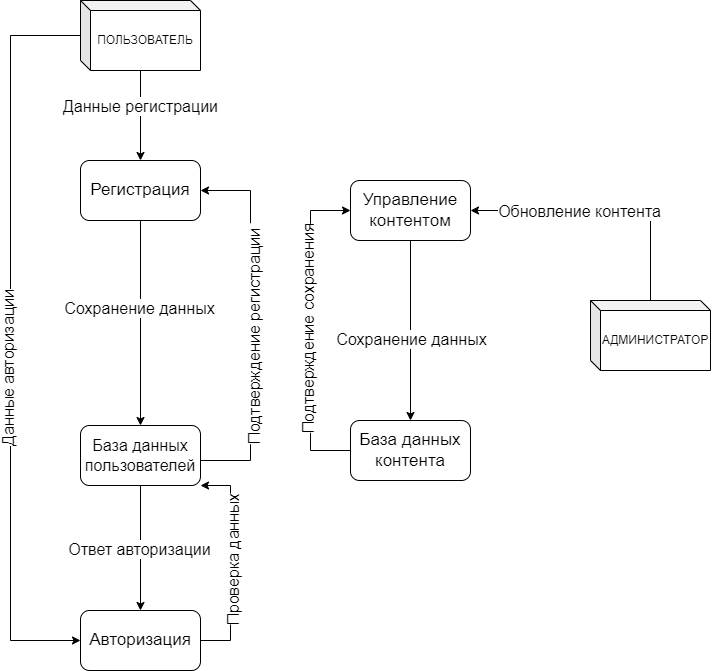
**Архитектура социальной сети "Ganz**

# **ПРИЛОЖЕНИЕ Б**

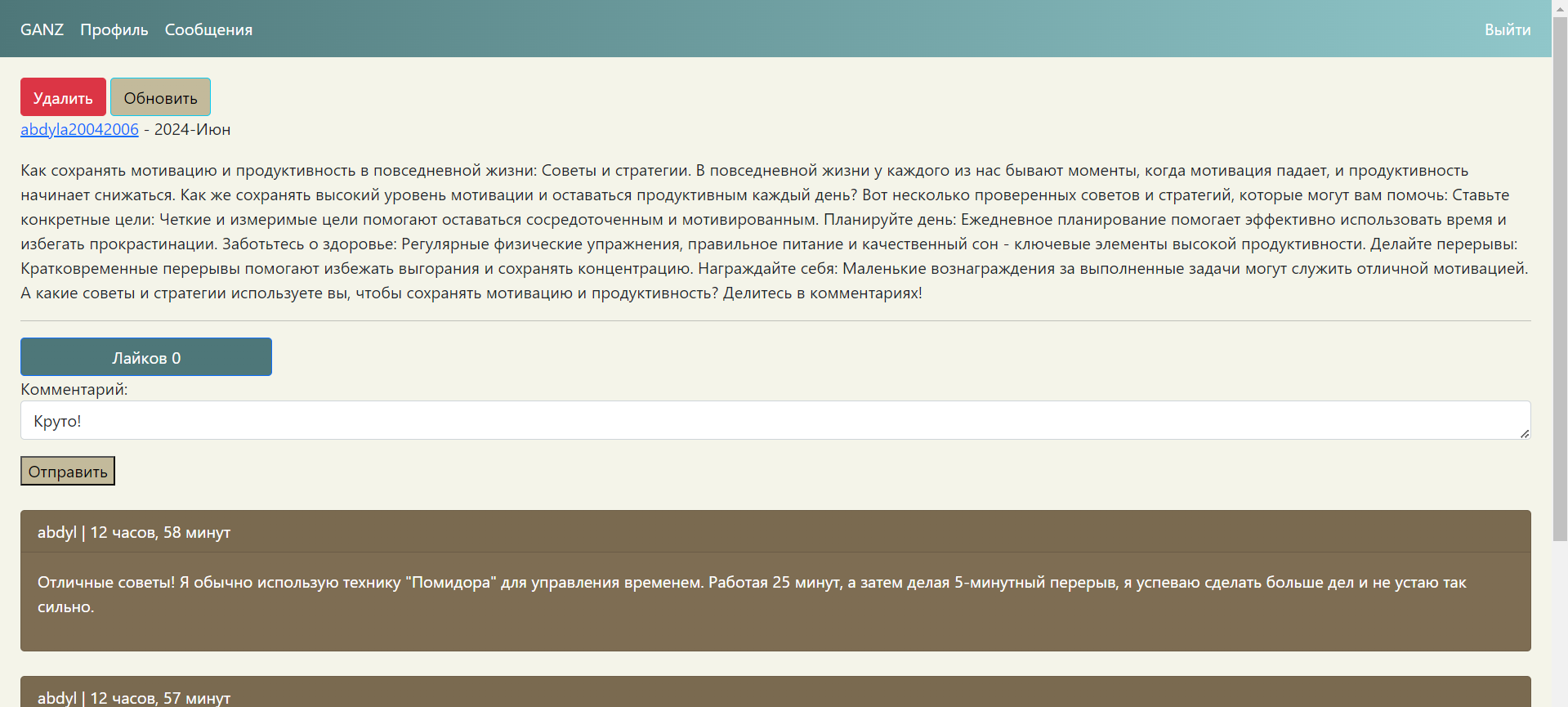
**Схема базы данных "Ganz"**

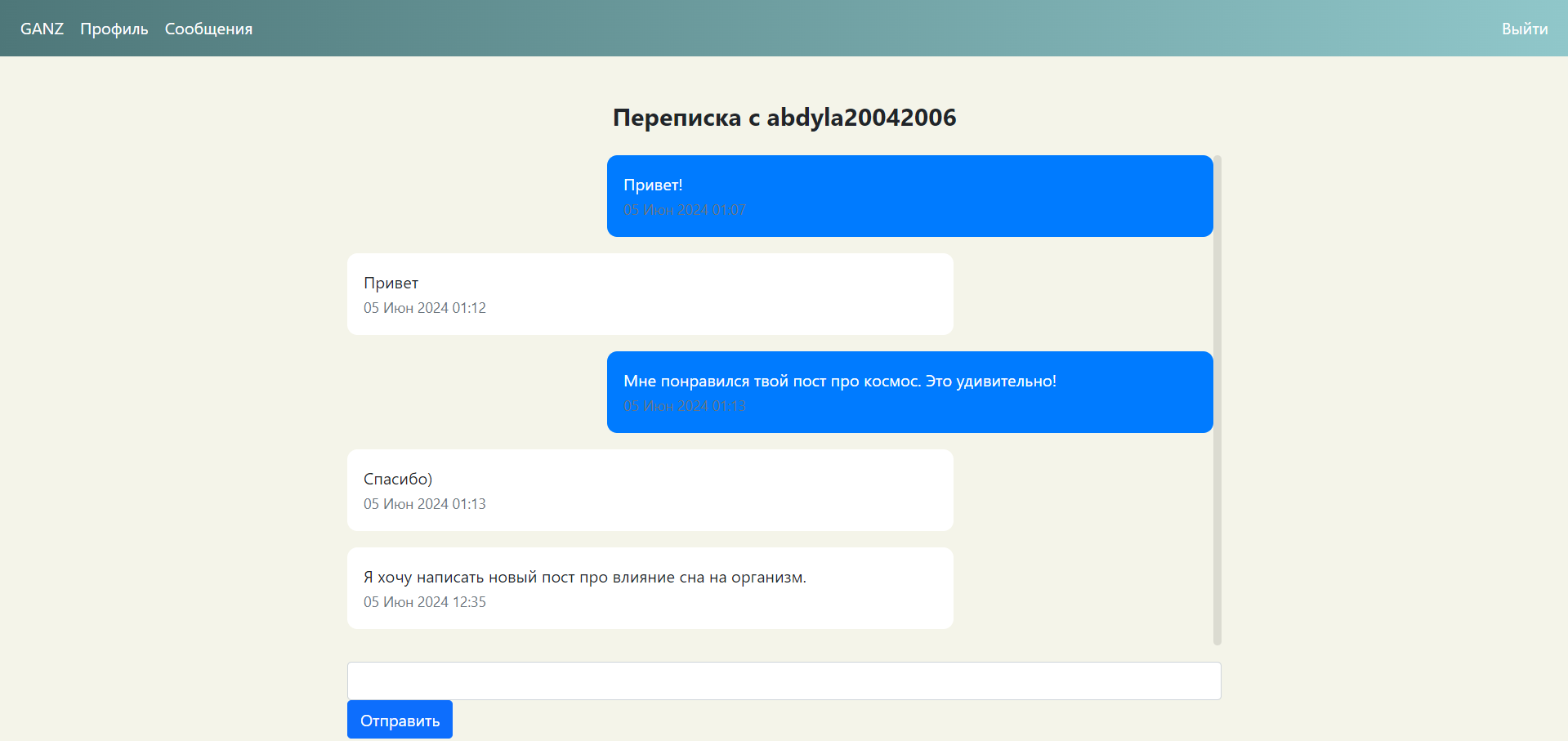
# **ПРИЛОЖЕНИЕ В**

**Диаграмма потоков данных в социальной сети "Ganz"**



# **ПРИЛОЖЕНИЕ Г**

**Экранные формы интерфейса пользовательской части и административной панели "Ganz"**

****