

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”

Факультет компьютерных наук

Кафедра информационных систем

Сервис для создания и управления персонализированными новостными лентами,
с возможностью фильтрации контента по интересам

«Мои Новости»

Курсовой проект

по дисциплине

Технологии программирования

09.03.02 Информационные системы и технологии

Встраиваемые вычислительные системы и интернет вещей

6 семестр 2023/2024 учебного года

Зав. кафедрой _____ к. т. н., доцент Д.Н. Борисов

Обучающийся _____ ст. 3 курса оч. отд. А. О. Поляков

Обучающийся _____ ст. 3 курса оч. отд. Д. Д. Рындин

Обучающийся _____ ст. 3 курса оч. отд. В.Д. Михайлов

Руководитель _____ В.С. Тарасов, ст. преподаватель ___. ___.20__

Воронеж 2024

Содержание

| | |
|--|----|
| Введение | 3 |
| 1 Постановка задачи..... | 5 |
| 1.1 Цели создания системы..... | 5 |
| 1.2 Требования к разрабатываемой системе | 5 |
| 1.3 Задачи проекта..... | 5 |
| 2 Анализ предметной области | 6 |
| 2.1 Терминология | 6 |
| 2.2 Обзор аналогов | 7 |
| 2.2.1 Яндекс.Дзен..... | 8 |
| 2.2.2 Telegram..... | 9 |
| 2.2.3 Apple News | 9 |
| 2.3 Моделирование системы..... | 10 |
| 2.3.1 Диаграммы прецедентов..... | 10 |
| 2.3.2 Диаграмма последовательности..... | 13 |
| 2.3.3 Диаграмма развертывания..... | 14 |
| 2.3.4 Диаграмма состояний | 15 |
| 2.3.5 Диаграмма активности..... | 16 |
| 3 Реализация | 17 |
| 3.1 Средства реализации | 17 |
| Заключение | 19 |
| Список использованных источников | 20 |

Введение

В современном мире приложения с новостными лентами занимают важное место, обеспечивая доступ к множеству источников информации и событий со всего мира. Эти приложения позволяют пользователям создавать персонализированные новостные ленты, отражающие их уникальные интересы и предпочтения.

Одной из ключевых особенностей таких приложений является возможность настройки контента под индивидуальные потребности каждого пользователя. Благодаря функции фильтрации по интересам, пользователи могут выбирать категории новостей, конкретные темы или ключевые слова, которые им наиболее интересны. Это позволяет получать информацию о том, что действительно важно для них, и игнорировать неактуальный или малозначимый контент.

Кроме того, приложения с новостными лентами обеспечивают удобный доступ к актуальным событиям в любое время и в любом месте. Благодаря мобильной оптимизации и возможности синхронизации между устройствами, пользователи могут быть в курсе последних новостей даже в движении, не теряя связи с миром вокруг себя.

Они также предлагают различные функции для улучшения пользовательского опыта, такие как создание персонализированных лент, сохранение интересных статей для последующего прочтения, уведомления о важных событиях и многое другое. Эти возможности делают использование приложений с новостными лентами не только информативным, но и удобным и приятным.

В данной курсовой работе рассматривается процесс разработки веб-приложения "Мои Новости", предназначенного для создания и управления персонализированными новостными лентами с возможностью фильтрации контента по интересам.

В рамках исследования будут описаны различные аспекты разработки такого приложения, начиная с анализа предметной области, определения его

концепции и основных требований. Затем будет изучено проектирование пользовательского интерфейса и пользовательского опыта с учетом современных тенденций и передовых практик в этой области. Также, внимание будет уделено выбору и интеграции необходимых технологий и API с целью обеспечения функциональности приложения, включая возможность настройки новостных лент, поиск информации и другие необходимые функции.

1 Постановка задачи

1.1 Цели создания системы

Целью данной работы является Реализация сервиса, позволяющего просматривать новости с возможностью выбора тегов для пользователей данного сервиса для разных категорий пользователей, просмотр новостей по этим тегам, также с возможностями получения актуальной информации из журналов, написанных редакторами.

1.2 Требования к разрабатываемой системе

- использование протокола передачи данных HTTP;
- обеспечение защиты системы от SQL-инъекций, защиты конфиденциальных данных при помощи необходимых механизмов;
- создание возможности работы с аккаунтами (регистрация, авторизация, настройка);
- поддержка русского языка в приложении;
- приложение должно иметь архитектуру вида Клиент-Сервер.

1.3 Задачи проекта

- просмотр новостей или журналов;
- выбор тегов для настройки просмотра;
- редактирование информации аккаунта;
- просмотр новостей от издателя;
- создание, редактирование, удаление новостей или журналов в качестве редактора;
- создание, редактирование, удаление новостей или журналов в качестве администратора;
- возможность назначения редакторов администратором.

2 Анализ предметной области

2.1 Терминология

Таблица 1 - Глоссарий

| | |
|-----------------|---|
| Веб-приложение | Программное обеспечение, разновидность прикладного программного обеспечения, предназначенная для работы на смартфонах, планшетах и других мобильных (портативных, переносных, карманных) устройствах. |
| БД | Это организованная коллекция данных, хранящихся в централизованном месте и структурированные таким образом, чтобы обеспечивать эффективное добавление, доступ, управление и обновление информации. |
| Frontend | Презентационная часть информационной или программной системы, ее пользовательский интерфейс и связанные с ним компоненты |
| Backend | Логика работы сайта, внутренняя часть продукта, которая находится на сервере и скрыта от пользователя. |
| REST API (REST) | Стиль архитектуры программного обеспечения для построения масштабируемых веб-приложений. |
| Django | Высокоуровневый Python веб-фреймворк, который позволяет быстро создавать безопасные и поддерживаемые веб-сайты. |

| | |
|--------------|--|
| СУБД | Набор программ, которые управляют структурой БД и контролируют доступ к данным, хранящимся в БД. |
| JavaScript | Это язык программирования, предназначенный для front-end-разработки и использующийся для взаимодействия с пользователем. |
| CSS | Формальный язык описания внешнего вида веб-страницы, написанного с использованием языка разметки. |
| HTML | Стандартизированный язык разметки для просмотра веб-страниц в браузере. |
| Vue.js | JavaScript-фреймворк с открытым исходным кодом для создания пользовательских интерфейсов. |
| SQL-инъекция | Это один из видов атак, при которых в приложение внедряются SQL-запросы с целью получения несанкционированного доступа к базе данных или выполнения операций, неавторизованных для пользователя. |

2.2 Обзор аналогов

При разработке сервиса для создания и управления персонализированными новостными лентами с функцией фильтрации контента по интересам "Мои Новости", важно уделить внимание актуальности проекта. Это подразумевает не только создание функционального продукта, но и обеспечение его конкурентоспособности на рынке в виду наличия уникальных качеств. Для этого необходимо тщательно изучить аналогичные сервисы, выявив их преимущества и недостатки, чтобы учесть успешные практики и избежать ошибок.

При анализе аналогов следует особенно обращать внимание на способы персонализации контента, эффективность алгоритмов фильтрации, а также удобство использования интерфейса. Отдельное внимание стоит уделить тому, какие дополнительные функции предлагают конкуренты для разработки уникальных возможностей.

2.2.1 Яндекс.Дзен

Яндекс.Дзен - это платформа для чтения и публикации контента, созданная компанией Яндекс. Она предоставляет пользователям персонализированные новостные ленты, собирая материалы со множества источников и адаптируя их под интересы каждого конкретного пользователя.

Яндекс.Дзен представляет собой медиа-платформу, где пользователи обладают возможностями читать статьи, просматривать фото, видео и другой контент, а также создавать и публиковать свои материалы. Сервис объединяет разнообразные источники информации, включая блоги, издательства, СМИ, и даже личные блоги пользователей для предоставления широкого спектра интересных тем.

Яндекс.Дзен ориентирована на широкий круг пользователей, интересующихся получением информации на различные темы. Это могут быть люди всех возрастов и профессий.

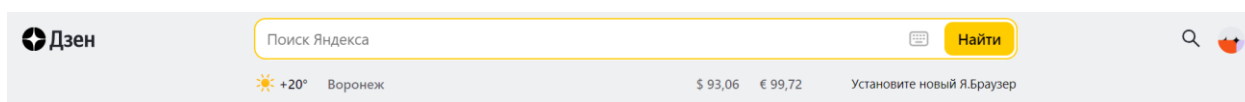


Рисунок 1 - Верхняя панель сервиса «Дзен»

Недостатки:

- из-за возможности публикации материалов пользователями, качество контента является неравномерным, ограничений по загрузке, кроме правил, нет;
- ограниченная настройка фильтрации контента;
- избыточное количество навязчивой рекламы, которую невозможно отключить в виду интеграции с другими сервисами;
- перегруженный интерфейс.

2.2.2 Telegram

Telegram – это мессенджер с возможностью обмена сообщениями, файлами, медиафайлами, аудио и видео вызовами. Он известен своей высокой степенью шифрования и возможностью создания защищенных чатов.

Telegram представляет собой популярный мессенджер, который позволяет пользователям обмениваться текстовыми сообщениями, мультимедийными файлами, аудио и видео вызовами. Он доступен на различных платформах, включая мобильные устройства и компьютеры, и предлагает синхронизацию между ними. Основной особенностью Telegram является его высокая степень шифрования, обеспечивающая безопасность переписки, а также возможность создания защищенных чатов с самоуничтожающимися сообщениями.

Сервис доступен на различных платформах, включая iOS, Android, Windows, macOS и Linux, что повышает удобство использования на различных устройствах. Также существует веб-версия, которая позволяет пользователям обмениваться сообщениями через браузер.

Telegram привлекает широкий круг пользователей. Он популярен среди тех, кто ценит конфиденциальность и безопасность в переписке, а также среди людей, которым важна возможность обмениваться файлами и медиафайлами в удобном интерфейсе. приложения.

Недостатки:

- ограничения в функциональности, связанной с тегами и таргетированными новостями (невозможно получить новости, если их нет в каналах или их не переслали в каналы, на которые оформлена подписка);
- отсутствие возможности писать большие статьи;
- наличие дополнительной функциональности, излишней для пользователей и невозможной для отключения.

2.2.3 Apple News

Apple News - это медиа-платформа и приложение, разработанные компанией Apple для пользователей устройств с операционной системой iOS и macOS. С его помощью пользователи могут получать доступ к новостям, статьям, журналам и другому контенту от различных изданий прямо на своих устройствах.

Apple News использует алгоритмы для адаптации контента под интересы каждого пользователя. Сервис анализирует предпочтения пользователя, включая темы, издания и авторов, чтобы создать индивидуализированные новостные ленты.

В дополнение к персонализированным лентам, Apple News предлагает кураторские подборки материалов от известных изданий, журналистов и блогеров. Это позволяет пользователям получать обзоры наиболее актуальных и интересных статей.

Платформа предлагает широкий спектр контента, включая новости, статьи, мультимедийные материалы, журналы и блоги по различным темам - от политики и экономики до развлечений и технологий.

Целевая аудитория Apple News в основном состоит из пользователей устройств Apple, таких как iPhone, iPad и Mac. Этот сервис привлекает тех, кто ценит удобство получения новостей и персонализированный подход к контенту, а также тех, кто предпочитает использовать продукты Apple в своей повседневной жизни.

Недостатки:

- ограничение использования сервиса для определенных стран;
- сервис доступен только для устройств с операционной системой IOS;
- ограниченный выбор контента.

2.3 Моделирование системы

2.3.1 Диаграммы прецедентов

Диаграмма прецедентов - это визуальное представление функциональных возможностей системы и взаимодействия между

пользователями (актерами) и системой. В ней выделены основные действия, которые пользователи могут выполнить в системе, и показывает, как эти действия связаны между собой. Основной целью диаграммы прецедентов является предоставление общее представление о функциональности системы и ее использовании.

Каждый прецедент представляет собой конкретное действие или операцию, которую пользователь может выполнить. Он описывается названием и может быть дополнен подробным описанием, объясняющим его функциональность.

На следующем рисунке представлена диаграмма прецедентов для сценариев для пользователя, описывающая основную функциональность сервиса:

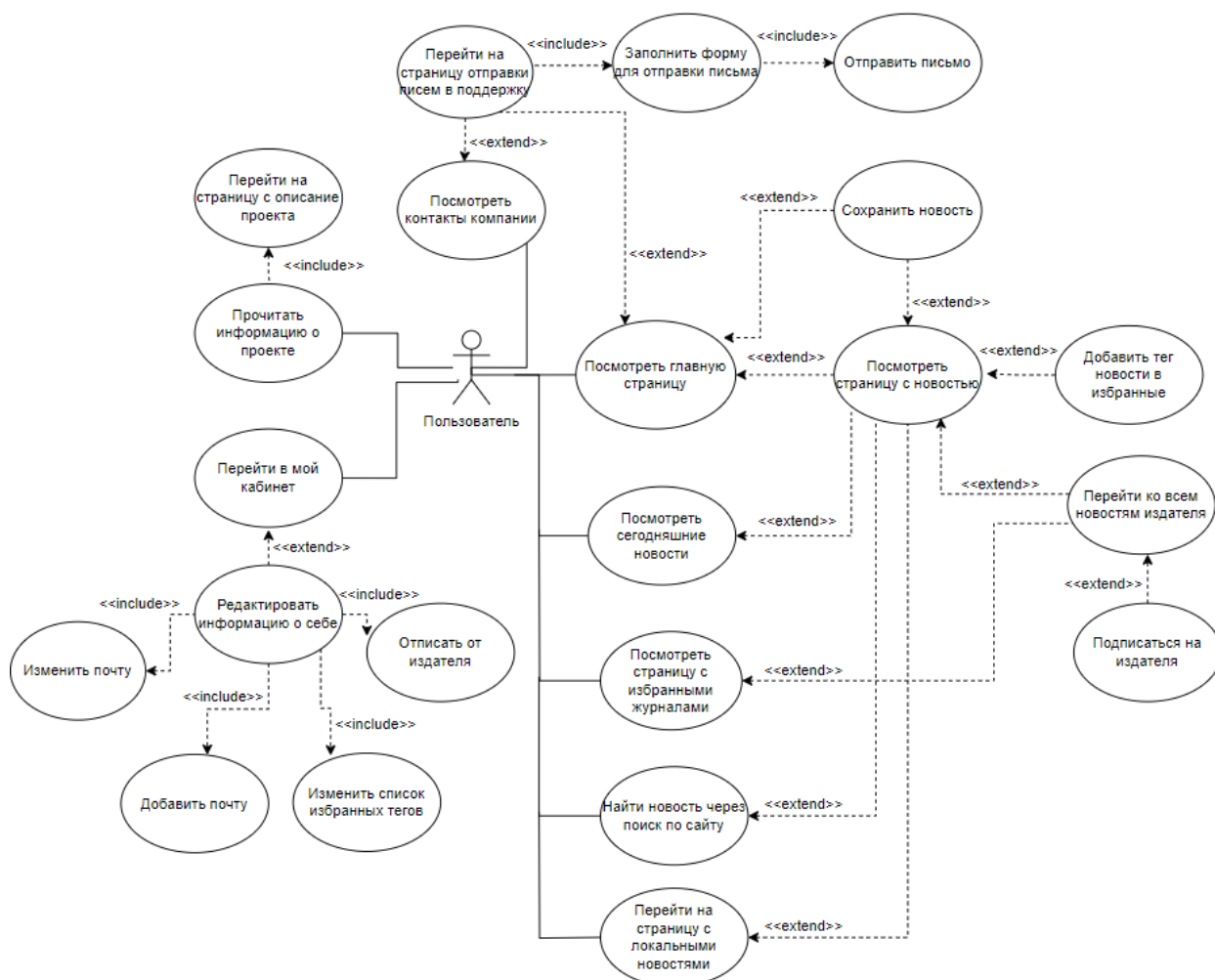


Рисунок 2 - Диаграмма прецедентов: Пользователь

Диаграмма прецедентов гостя описывает сценарии взаимодействия незарегистрированных пользователей, которые обладают ограниченными возможностями по использованию сервиса. Далее представлена диаграмма прецедентов для гостя:

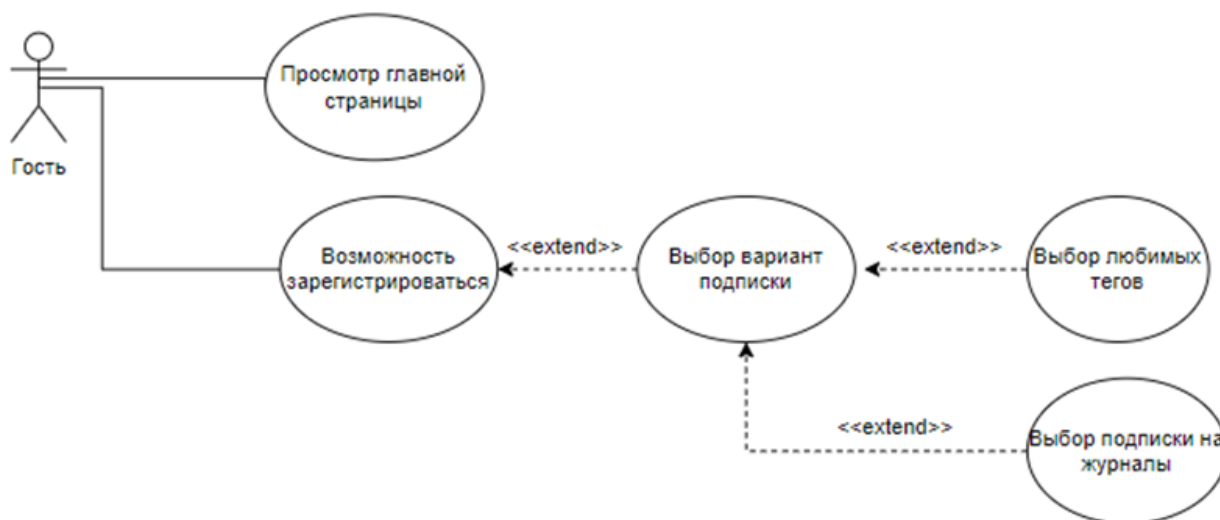


Рисунок 3 - Диаграмма прецедентов: Гость

Редактор является основным лицом, создающим контент для сервиса. В его возможности входит управление предоставляемым им контентом. Далее представлена диаграмма прецедентов для редактора:

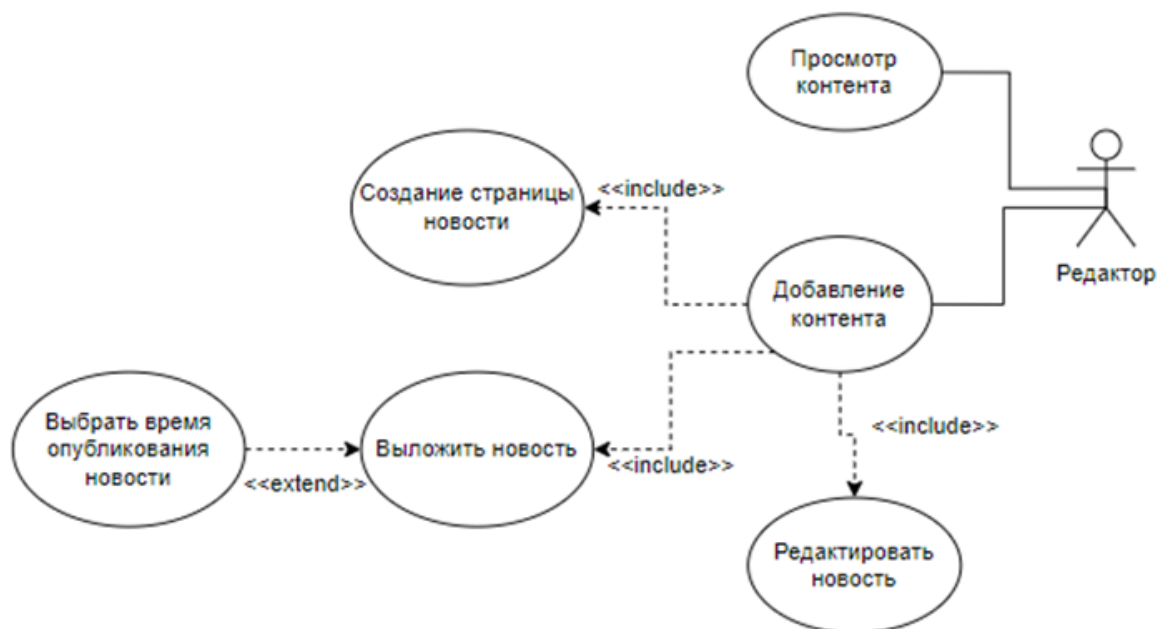


Рисунок 4 - Диаграмма прецедентов: Редактор

Администратор может управлять контентом. Также администратор управляет ролью редактора. Далее представлена диаграмма прецедентов для администратора:

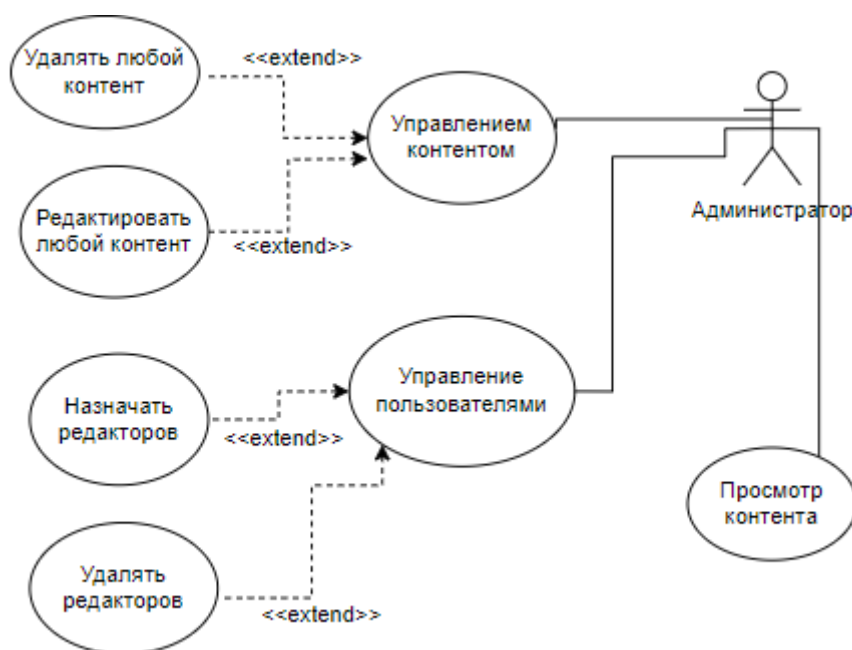


Рисунок 5 - Диаграмма прецедентов: Администратор

2.3.2 Диаграмма последовательности

Диаграмма последовательности взаимодействия пользователя с представляет собой отображение последовательности сообщений между клиентской и серверной частями при выполнении определенного сценария. Она демонстрирует, как клиентские объекты (например, пользовательский интерфейс) взаимодействуют с сервером для выполнения определенных операций или получения информации.

На диаграмме последовательности представлен каждый шаг взаимодействия для различных объекты (клиент, сервер. Каждое сообщение указывает на конкретное действие, которое выполняется, и может включать передачу данных или запрос на выполнение определенной операции.

Основной целью данной диаграммы последовательности является представление последовательность действий, необходимых для выполнения определенного сценария использования сервиса, иллюстрация взаимодействия между клиентом и сервером. Данная диаграмма нужна для отображения процесса взаимодействия.

Далее представлена соответствующая диаграмма последовательности:

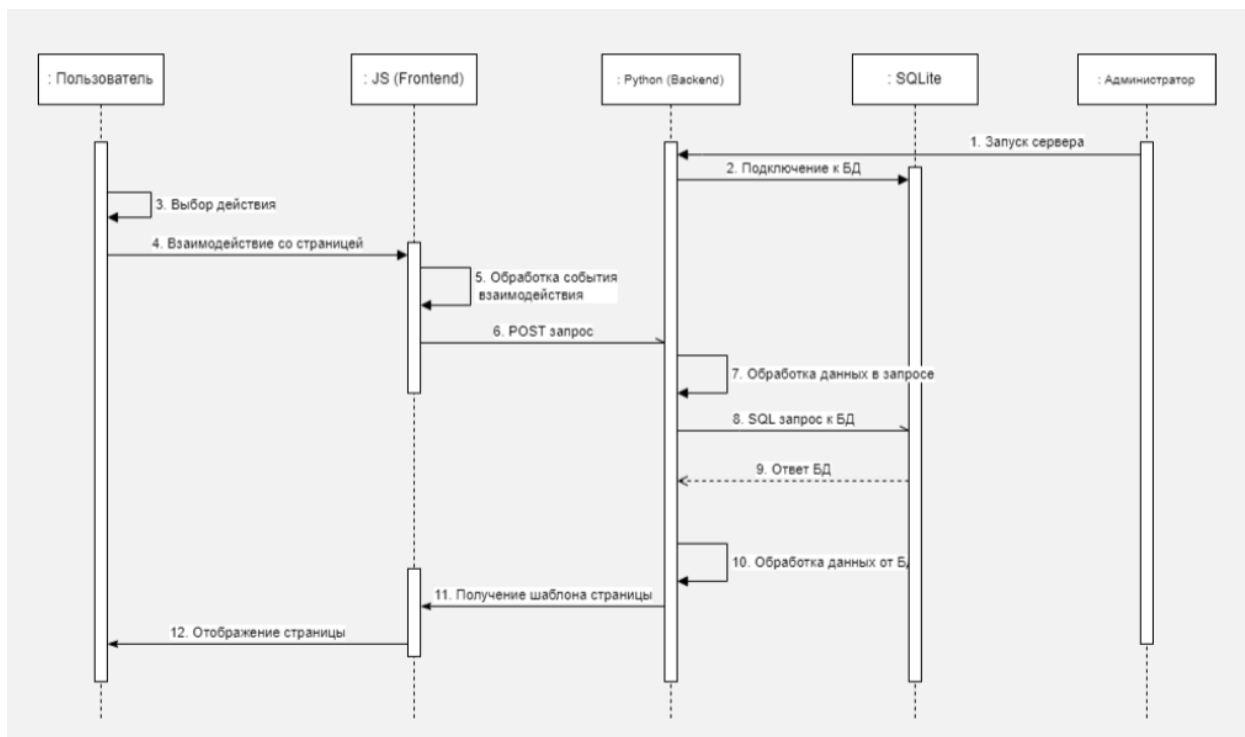


Рисунок 6 - Диаграмма последовательности взаимодействия пользователя и системы

2.3.3 Диаграмма развертывания

Диаграмма развертывания для сервиса представляет собой визуальное представление физической архитектуры системы, отображая размещение и взаимодействие различных компонентов приложения на различных узлах сети. Она позволяет понять, какие компоненты системы находятся на клиентской стороне, а какие на стороне сервера, а также как они связаны между собой.

Компоненты системы (например, клиентское приложение, серверное приложение, база данных) представлены в виде узлов.

Основная цель диаграммы развертывания - представить архитектуру системы и распределение ее компонентов. Это помогает разработчикам и системным администраторам понять, как организовано взаимодействие между клиентом и сервером, и обеспечить эффективную работу всей системы.

Далее представлена диаграмма развертывания для сервиса:

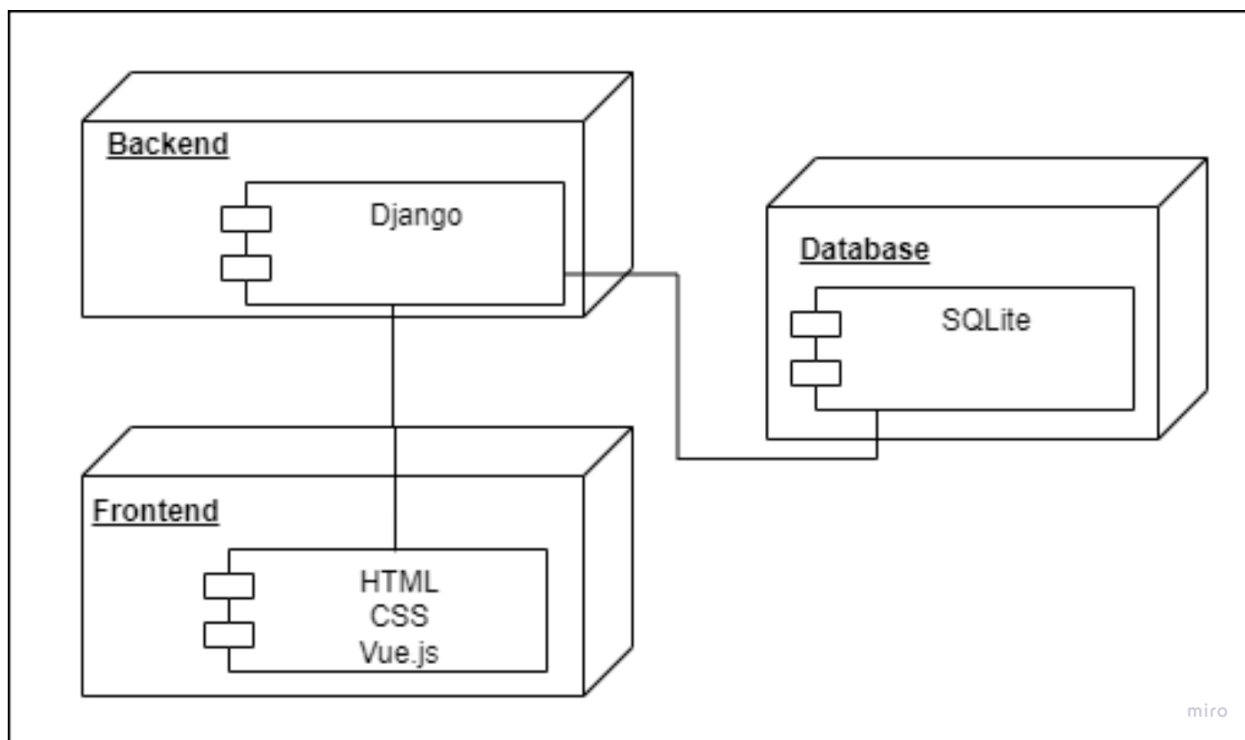


Рисунок 7 - Диаграмма развертывания

2.3.4 Диаграмма состояний

Диаграмма состояний для пользователя в сервисе является графическим представлением различных состояний, в которых может находиться пользователь при взаимодействии с системой, а также переходов между этими состояниями. Это позволяет визуальнo представить поведение пользователя в процессе использования сервиса и его реакции на различные события.

Диаграмма состояний помогает понять, как пользователь взаимодействует с системой в различных сценариях использования, а также как система реагирует на действия пользователя. Это обеспечивает реализацию взаимодействия между пользователем и сервисом. Также диаграмма описывает получение роли редактора пользователем.

Далее представлена диаграмма состояний пользователя:

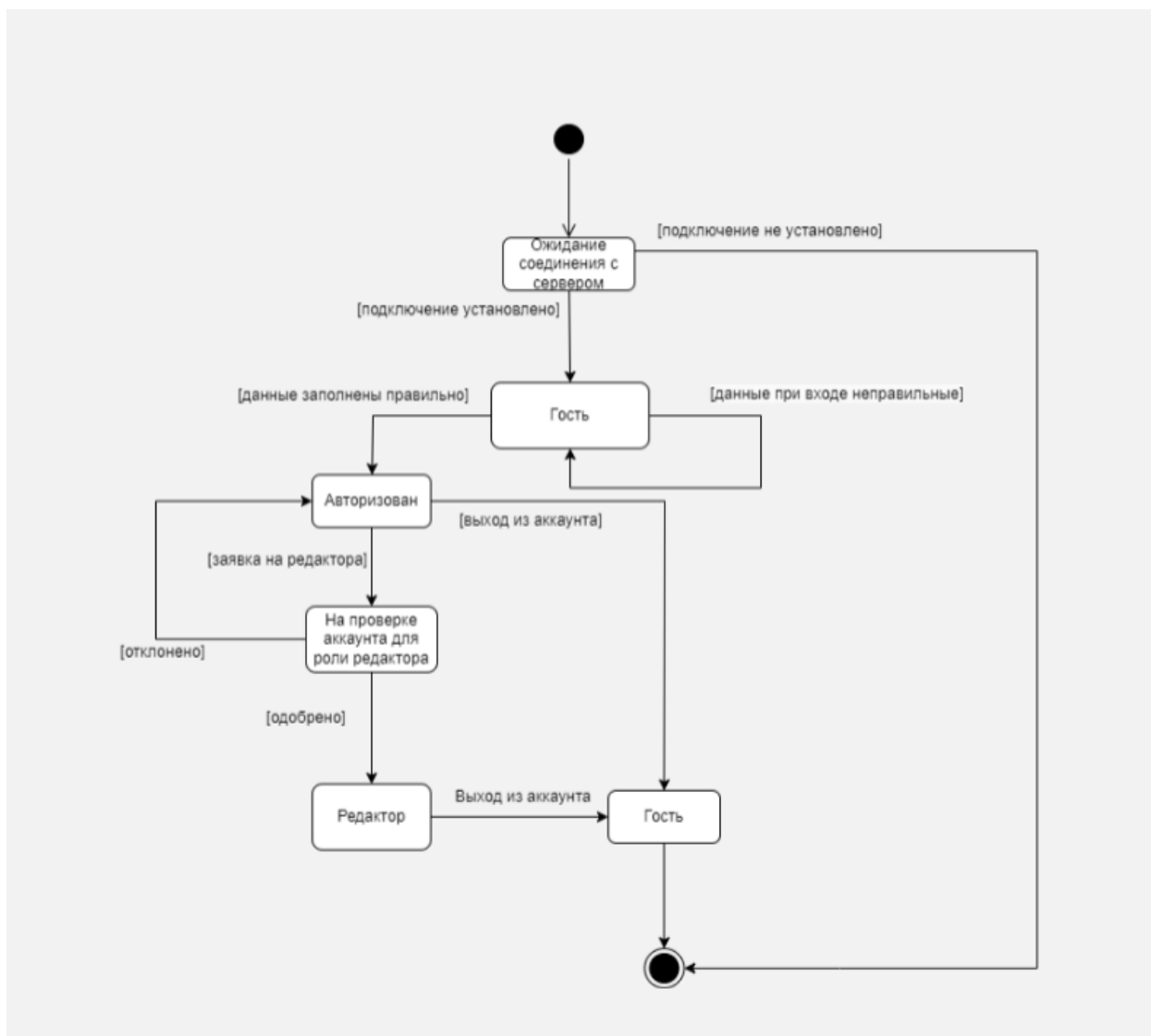


Рисунок 8 - Диаграмма состояний пользователя

2.3.5 Диаграмма активности

Диаграмма активностей для пользователя в данном сервисе представляет собой графическое отображение последовательности действий, которые пользователь выполняет при взаимодействии с системой. Она является визуальным представлением активностей пользователя в процессе использования сервиса и последовательности выполнения этих активностей.

На диаграмме активностей каждая активность описывает конкретное действие или операцию, выполняемую пользователем. Связи между активностями показывают порядок выполнения действий.

Диаграмма активностей отображает, как пользователь взаимодействует с системой в различных сценариях использования, и какие шаги он должен выполнить для достижения конкретной цели. Это показывает взаимодействие между пользователем и сервисом.

Далее представлена диаграмма активностей пользователя:

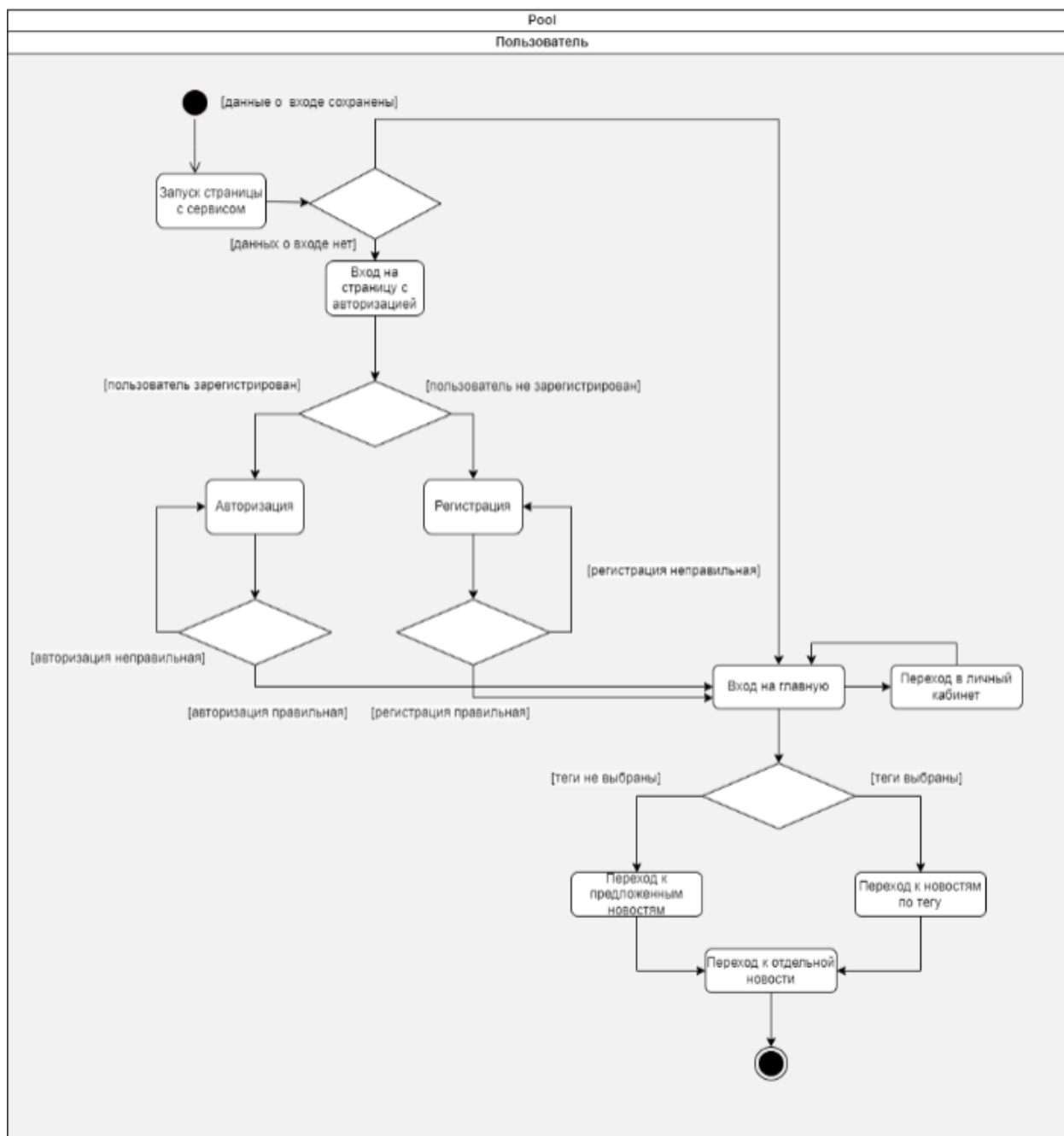


Рисунок 9 - Диаграмма активностей пользователя

3 Реализация

3.1 Средства реализации

Заключение

Список использованных источников

1.