Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение   
высшего образования

**Иркутский национальный исследовательский технический университет**

|  |
| --- |
| Институт информационных технологий и анализа данных |
| наименование института |

Допускаю к защите

|  |  |
| --- | --- |
| Руководитель |  |
|  | подпись |
|  | М.Д. Каташевцев | |
|  | И.О. Фамилия |

Разработка фуллстек-cервиса с использованием Django и vue

|  |
| --- |
| наименование темы |

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе по дисциплине

|  |
| --- |
| Web-программирование |

|  |
| --- |
| 1.004.00.00 - ПЗ |
| обозначение документа |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Выполнил студент |  | ИСТб-22-2 |  |  |  | | Дробит А.А. |
|  |  | шифр |  | подпись |  | | И.О. Фамилия |
| Нормоконтроль |  |  |  |  |  | | М.Д. Каташевцев |
|  |  |  |  | подпись |  | | И.О. Фамилия |
| Курсовой проект защищен с оценкой | | | |  | |  | |

Иркутск 2024 г.

**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**ЗАДАНИЕ**

**НА КУРСОВУЮ РАБОТУ**

|  |  |
| --- | --- |
| По курсу | Web-программирование |
| Студенту | Дробит А.А. |
|  | (фамилия, инициалы) |
| Тема работы: | Разработка фуллстек-cервиса с использованием Django и vue | |
|  | | |
| Рекомендуемая литература: | | |
| 1. Чаинка [Электронный ресурс] – Режим доступа: <https://django.tealeaf.su/> 2. Vue.js 3 By Example: Blueprints to Learn Vue Web Development, Full-stack Development, and Cross-platform Development Quickly / John Au-Yeung, 2021. 399 с. | | |

|  |  |  |
| --- | --- | --- |
| Дата выдачи задания |  | 2 октября 2024 г. |
| Задание получил студент | |  |  | Дробит А.А. |
|  | | подпись |  | Фамилия И.О. |

|  |  |  |
| --- | --- | --- |
| Дата представления проекта руководителю | 12 декабря 2024 г. | |
| Руководитель курсового проекта | |  |  | Каташевцев М.Д. |
|  | | подпись | Фамилия И.О. |

**Содержание**

[Введение 4](#_Toc185192585)

[1 Функциональные возможности 5](#_Toc185192586)

[2 Инфологическая модель данных 6](#_Toc185192587)

[3 Физическая модель данных 7](#_Toc185192588)

[4 API описание 8](#_Toc185192589)

[5 Иерархия компонентов 13](#_Toc185192590)

[6 Интерфейс 14](#_Toc185192591)

[7 Отличительная особенность проекта 18](#_Toc185192592)

[Список используемых источников 19](#_Toc185192593)

Введение

В условиях современной цифровизации и активного использования социальных медиа возрастает потребность в создании инновационных платформ для общения и самовыражения. Одной из актуальных тенденций является стремление пользователей к конфиденциальности, что порождает интерес к анонимным платформам, где можно свободно делиться мыслями и обсуждать различные темы без привязки к личности.

Целью данной курсовой работы является разработка веб-сайта «Анонимная социальная сеть», предназначенного для создания безопасной и открытой среды, где пользователи могут публиковать посты, комментировать их и участвовать в обсуждениях. Особенностью системы является обеспечение анонимности, что способствует более честному и открытомy общению, исключая предвзятость, основанную на личности автора.

Разрабатываемая платформа включает в себя следующие основные функции: публикация текстовых записей, возможность комментирования, редактирования и добавления постов и авторов. Интерфейс создается с акцентом на простоту и удобство, чтобы обеспечить комфортное взаимодействие пользователей с системой.

Таким образом, разработка анонимной социальной сети направлена на решение актуальных задач создания пространства для свободного общения и обмена идеями. Это делает проект востребованным и значимым, отвечающим современным запросам пользователей цифровых платформ.

1 Функциональные возможности

Анонимная социальная сеть обладает широким набором функциональных возможностей, обеспечивающих удобство и эффективность взаимодействия пользователей. В системе реализованы все базовые операции для работы с контентом: создание, просмотр, редактирование и удаление анонимных постов, их параметров, авторов и комментариев. Это позволяет пользователям свободно выражать свои мысли и участвовать в обсуждениях, сохраняя при этом полную конфиденциальность.

Платформа включает функционал для отображения статистики, предоставляющий данные о количестве опубликованных постов и комментариев, а также минимальные и максимальные значения идентификаторов записей. Этот инструмент помогает анализировать активность пользователей и структуру контента.

Также, реализована возможность экспорта данных, что может быть полезно для аналитики и внешней обработки информации.

Система поддерживает управление изображениями, что дает пользователям возможность добавлять к своим постам иллюстрации, делая их более выразительными и информативными.

Дополнительно платформа оснащена функцией аутентификации, которая позволяет пользователям создавать и защищать свои анонимные профили. Безопасный вход обеспечивает конфиденциальность и защищает данные учетной записи.

2 Инфологическая модель данных

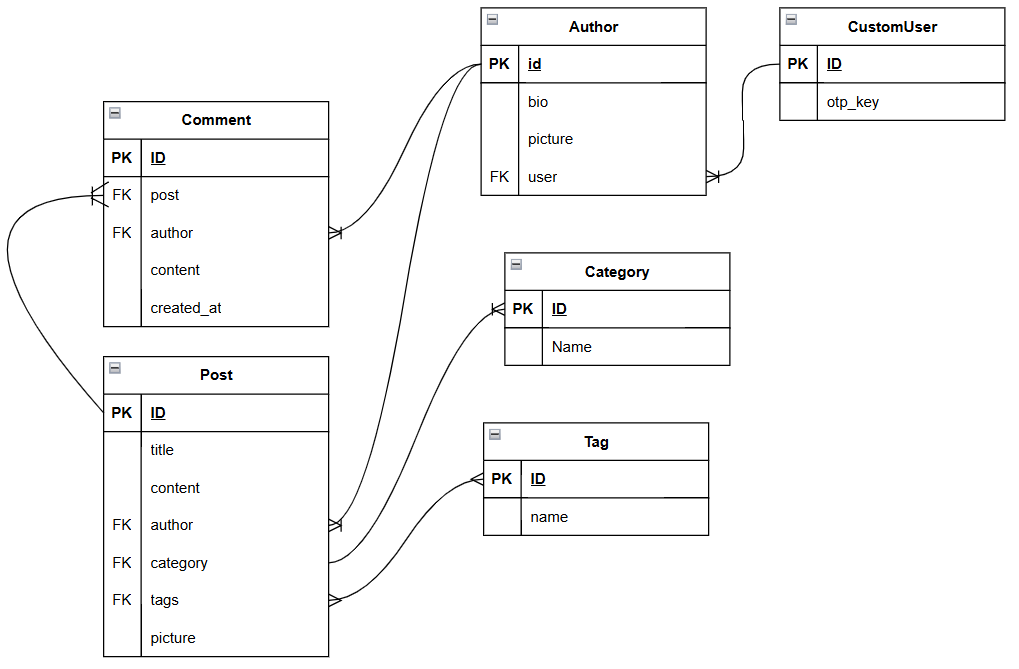


Рисунок 2.1 – Инфологическая модель

На рисунке 2.1 приведена инфологическая модель данных. Она включает пять взаимосвязанных сущностей: "Post", "Comment", "Author", "Category", "Tag" и "CustomUser". Эти сущности отражают основные данные для публикации на сайте "Постов".

3 Физическая модель данных

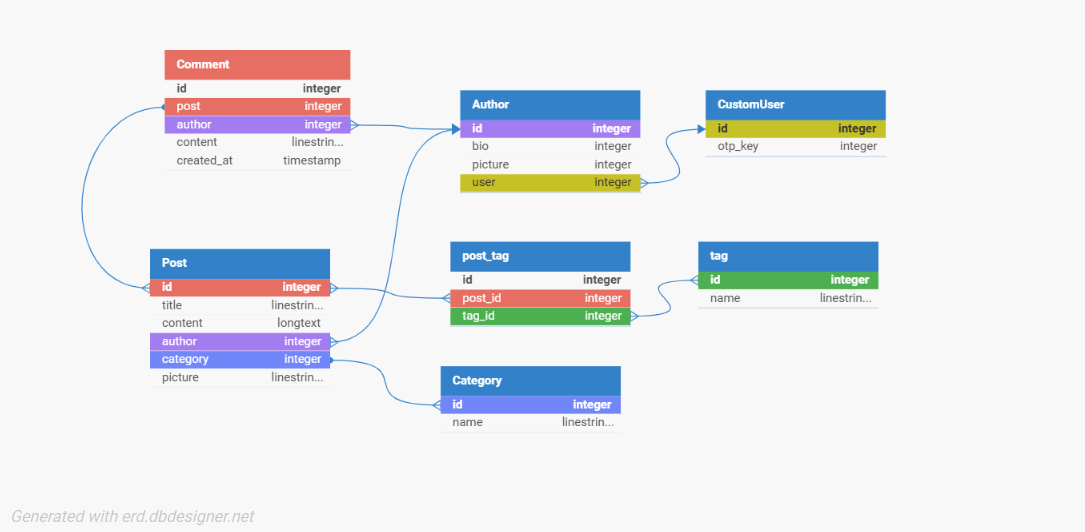


Рисунок 3.1 – Физическая модель

На рисунке 3.1 приведена физическая модель данных, которая включает в себя все необходимые таблицы, столбцы, связи, свойства базы данных для физической реализации баз данных.

4 API описание

Таблица 4.1 – класс TagViewset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **URL** | **Метод** | **Назначение** | **JSON вход** | **JSON выход** |
| /api/tags / | GET | Получение списка тэгов | - | [{"id": int, "name": str}] |
| /api/tags / | POST | Создание нового тэга | {"name": str} | {"id": int, "name": str, "group": str, "room": {"number": int}} |
| /api/tags /{id}/ | PUT | Редактирование данных тэга | {name: str} | {"id": int, "name": str, } |
| /api/student/{id}/ | DELETE | Удаление тэга |  | - |
| /api/tags/stats/ | GET | Получение статистики тэгов | - | {  "count": int,  "avg": int,  "max": int,  "min": int  } |

Таблица 4.2 – класс CategoryViewset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **URL** | **Метод** | **Назначение** | **JSON вход** | **JSON выход** |
| /api/categorys/ | GET | Список всех категорий | - | Список категорий (array) |

Таблица 4.3 – класс PostViewset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **URL** | **Метод** | **Назначение** | **JSON вход** | **JSON выход** |
| /api/ posts / | GET | Получение всех постов | - | Список постов  [{author: int},  {category: int},  {content: text },  {id: int},  {picture: String},  {tags: Array},  {title: string}] |
| /api/ posts / | POST | Создание нового поста | [{author: int},  {category: int},  {content: text },  {id: int},  {picture: String},  {tags: Array},  {title: string}] | [{author: int},  {category: int},  {content: text },  {id: int},  {picture: String},  {tags: Array},  {title: string}] |
| /api/posts/{id}/ | PUT | Обновление данных поста | [{author: int},  {category: int},  {content: text },  {id: int},  {picture: String},  {tags: Array},  {title: string}] | [{author: int},  {category: int},  {content: text },  {id: int},  {picture: String},  {tags: Array},  {title: string}] |
| /api/posts/{id}/ | DELETE | Удаление поста | - | - |
| /api/posts/stats/ | GET | Получение статистики по постам | - | {"count": int, "avg": float, "max": int, "min": int} |

Таблица 4.4 – класс CommentViewset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **URL** | **Метод** | **Назначение** | **JSON вход** | **JSON выход** |
| /api/comments / | GET | Список всех комментариев | - | [{"id": int, "content": str,  “author”: int, "post": str}] |
| /api/ comments / | POST | Создание нового комментария | [{"id": int, "content": str,  “author”: int, "post": str}] | [{"id": int, "content": str,  “author”: int, "post": str}] |
| /api/comments/{id}/ | GET | Получение информации о комментарии | - | [{"id": int, "content": str,  “author”: int, "post": str}] |

Таблица 4.5 – класс AuthorViewset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **URL** | **Метод** | **Назначение** | **JSON вход** | **JSON выход** |
| /api/authors / | GET | Получение списка авторов | - | [{"id": int, "bio": str, "picture": str,  "user": int }] |
| /api/ authors/ | POST | Создание нового автора | [{"id": int, "bio": str, "picture": str,  "user": int }] | [{"id": int, "bio": str, "picture": str,  "user": int }] |
| /api/authors /{id}/ | GET | Получение информации об авторе | - | [{"id": int, "bio": str, "picture": str,  "user": int }] |
| /api/authors /{id}/ | PUT | Обновление информации об авторе | [{"id": int, "bio": str, "picture": str,  "user": int }] | [{"id": int, "bio": str, "picture": str,  "user": int }] |
| /api/authors /{id}/ | DELETE | Удаление автора | - | - |
| /api/authors /stats/ | GET | Получение статистики авторов | - | {"count": int, "avg": float, "max": int, "min": int} |
| /api/authors/reset\_otp/ | POST | Сброс OTP ключа | - | {"success": true, "new\_otp\_key": "key" } |

Таблица 4.6 – класс UserViewSet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **URL** | **Метод** | **Назначение** | **JSON вход** | **JSON выход** |
| /api/users/info/ | GET | Получение информации об авторизованном пользователе | - | {"is\_authenticated": bool, "username": str, "user\_id": int, "is\_superuser": bool,  "otpVerified":bool,  "otpRequired": bool} |
| /api/users/login / | POST | Авторизация пользователя | {"username": str, "password": str } | [{  "otp\_required":bool, "otp\_key": str }] |
| /api/users/otp-login/ | POST | Аутентификация пользователя | {“otp\_code”: int} | {"success": bool} |
| /api/user/otp-status/ | GET | Проверка статуса OTP | - | {"otp\_verified ": bool} |

5 Иерархия компонентов

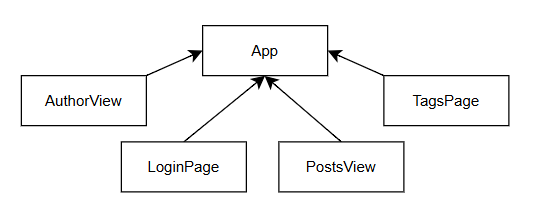


Рисунок 5.1 – Иерархия компонентов

На рисунке 5.1 представлена иерархия компонентов, используемых в проекте. Главным компонентом является App.vue, который служит контейнером для всех остальных элементов.

6 Интерфейс

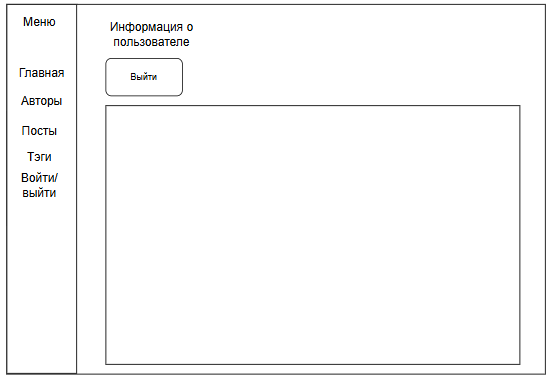


Рисунок 6.1 – Макет интерфейса App

На рисунке 6.1 представлен основной макет приложения, включающий навигационное меню в боковой части экрана и область для отображения выбранного содержимого. Навигация позволяет переключаться между различными разделами.

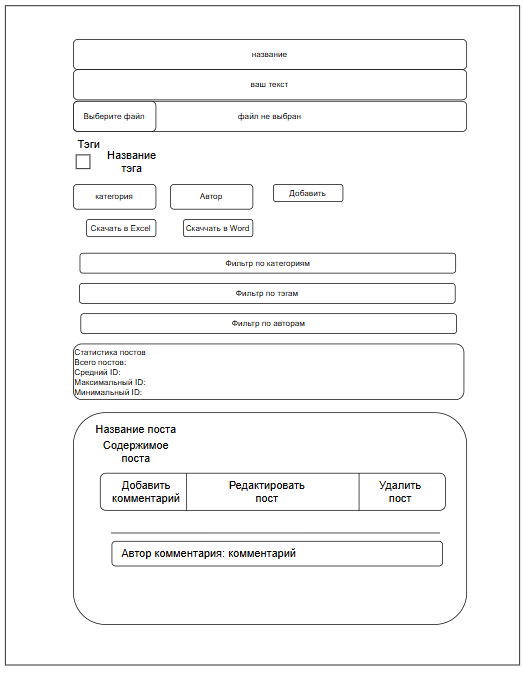


Рисунок 6.2 – Макет интерфейса PostsView

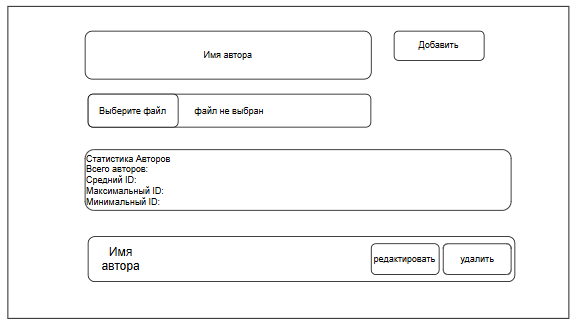


Рисунок 6.3 – Макет интерфейса AuthorsView

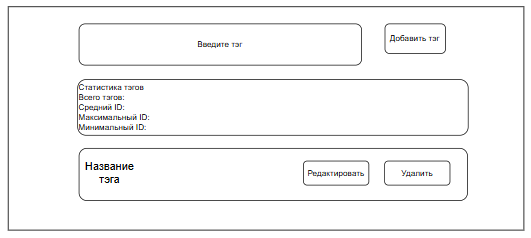


Рисунок 6.4 – Макет интерфейса TagsViewset

На рисунках 6.2-6.4 представлены макеты разделов: Главное, Посты, Автор и Тэги. Интерфейс для управления данными включает: добавление, редактирование, удаление, различные фильтры, вывод статистики. Также на макете раздела «Посты» есть кнопки экспорта данных в Excel и Word.

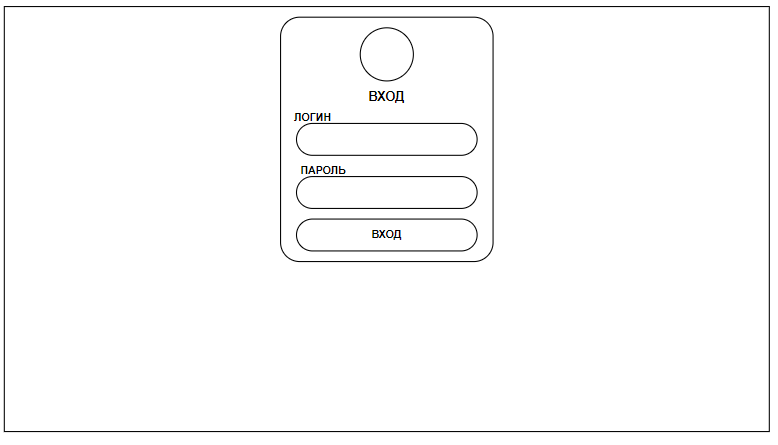


Рисунок 6.7 – Макет интерфейса LoginView

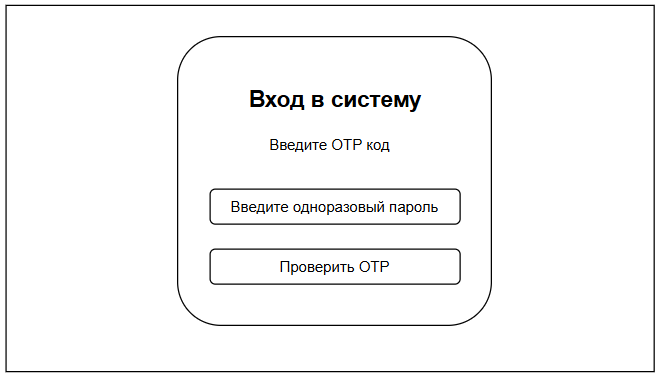


Рисунок 6.8 – Макет интерфейса LoginView

На рисунке 6.7 и 6.8 представлено окно авторизации пользователей

7 Отличительная особенность проекта

Уникальная система комментирования: новые грани взаимодействия с контентом

Система комментирования в проекте реализована с учетом акцента на прозрачность, постоянство и уникальность пользовательского опыта. Комментарии, оставленные под постами, нельзя удалить или изменить, что способствует сохранению истории обсуждений и укреплению доверия среди пользователей.

**Особенности:**

* **Неизменяемость комментариев:** Все комментарии остаются в исходном виде, обеспечивая прозрачность и честность общения.
* **Многопользовательские авторы:** Один аккаунт может включать несколько авторов, которых пользователь создает самостоятельно. Это позволяет гибко распределять роли и взаимодействовать с контентом от имени разных авторов.
* **Простота и удобство:** Функционал комментирования разработан так, чтобы быть максимально понятным и доступным для всех пользователей.

**Реализация:**  
Система комментирования интегрирована в интерфейс приложения, предоставляя пользователям возможность оставлять текстовые комментарии к постам. Неизменяемость обеспечивается отсутствием функций редактирования и удаления комментариев после их публикации. Для реализации многопользовательских авторов предусмотрен механизм управления подаккаунтами, где главный пользователь создает и настраивает авторов с уникальными именами или аватарами.

Эта система делает процесс взаимодействия с контентом более честным, открытым и увлекательным, подчеркивая индивидуальность каждого автора и создавая живую атмосферу общения.

Список используемых источников

1. TheCatAPI Documentation [Электронный ресурс] – Режим доступа: <https://developers.thecatapi.com/view-account/ylX4blBYT9FaoVd6OhvR?report=bOoHBz-8t>
2. Документация vue 3 [Электронный ресурс] – Режим доступа: <https://vuejs.org/guide/introduction.html>
3. Чаинка [Электронный ресурс] – Режим доступа: <https://django.tealeaf.su/>
4. Использование Python и Excel для обработки и анализа данных [Электронный ресурс] – Режим доступа: <https://habr.com/ru/companies/otus/articles/331998/>
5. Объект Document модуля python-docx в Python [Электронный ресурс] – Режим доступа: <https://docs-python.ru/packages/modul-python-docx-python/klass-document/>

Ссылка на GitHub:

<https://github.com/drars38/WEBB>

