

**ПРОЕКТНАЯ  
ДОКУМЕНТАЦИЯ**

web-сервиса для хостинга изображений

«HosteyPic»

*Версия 2.5*

## **1. ОПИСАНИЕ ПРОЕКТА**

### **1.1. Полное наименование системы и ее условное обозначение**

Социальный Интернет-сервис для хостинга изображений «HosteyPic».  
Сокращенное обозначение – фотохостинг «HosteyPic».

### **1.2. Наименования организации-заказчика и организаций-участников работ**

Филиал Национального исследовательского университета «НИУ «МЭИ» в г. Смоленске.

### **1.3. Исполнитель работ**

Исполнителем работ является команда «Nice Solutions».

### **1.4. Исходные репозитории**

Репозиторий проекта: <https://gitlab.com/OnBell/hosteypic>

### **1.5. Дизайн проекта**

Дизайн: <https://www.figma.com/design/DSyShlF0uJwIiiJcHUFgNH/HosteyPIC>

## **2. ЦЕЛИ И ОБЪЕМЫ ПРОЕКТА**

### **2.1. Цели и тип проекта**

Тип ПО: веб-приложение.

Цели проекта:

- предоставление аналога сервисов формата «фотохостинг»;
- создание Интернет-ресурса для публикации и сохранения изображений;
- обеспечение возможности установки категории (далее тегирования) опубликованных изображений, добавления подписей к ним;
- обеспечение возможности модерации публикуемого контента;
- обеспечение возможности публикации информации о себе авторами сервиса.

### **2.2. Проблематика проекта**

В ходе опроса потенциальных пользователей была определена основная проблема, которую должен решать проект, а именно отсутствие достойных аналогов

иностранных фотохостингов на локальном рынке. Заданные вопросы в ходе интервьюирования отражены в приложении А. Проанализированные требования отражены в п. 2.3 настоящего документа.

### **2.3. Функциональные требования**

Сервис должен обладать следующими функциональными требованиями:

- пользователи должны иметь возможность быстро и легко загружать изображения;
- пользователи должны иметь возможность ставить реакции на контент других пользователей;
- пользователи должны иметь возможность тегирования своего контента и добавления текстовых подписей;
- пользователи должны иметь возможность удаления всего своего контента;
- пользователи должны иметь возможность поиска контента по текстовому запросу;
- пользователи должны иметь возможность заполнять свой профиль следующими данными: псевдоним, фото профиля, графа «О себе», ссылки на соц. сети (ВКонтакте, Одноклассники, GitHub, GitLab);
- пользователи должны иметь возможность подписываться на других пользователей и отписываться от них;
- пользователи должны иметь возможность просмотр чужого контента;
- пользователи должны иметь возможность просмотр контента по подписке на пользователей;
- пользователи должны иметь возможность оставлять жалобы на неподобающий контент.
- в системе должны быть предусмотрены роли: пользователь, заблокированный пользователь, неавторизованный пользователь, модератор и администратор;

- модераторы должны иметь возможность удаления чужого контента в рамках соглашения;
- модераторы должны иметь возможность блокировки других пользователей в рамках соглашения;
- администратор должен иметь возможность загрузки тегов в систему и их удаление;
- администратор должен иметь возможность удаления профиля другого пользователя в рамках соглашения или по просьбе;
- администратор должен иметь возможность добавлять адреса электронной почты в «белый список» на время демонстрации работы.

## **2.4. Нефункциональные требования**

Сервис должен обладать следующими нефункциональными требованиями:

- интуитивно понятный интерфейс с поддержкой устройств планшетного типа;
- работа в системе должна быть интуитивно понятна неопытному пользователю;
- сервис должен обеспечивать защиту данных пользователей в рамках сервиса и на весь период нахождения данных в базе данных сервиса;
- сервис должен обеспечивать защиту пользователей от неправомерного и шокирующего контента в рамках законов стран, где сервис официально функционирует;
- сервис должен иметь возможность масштабируемости в рамках дальнейшего его развития;
- система должна обеспечивать одновременную работу не менее 100 пользователей;
- надежность системы должна быть не менее 98% (суммарное допустимое время простоя в работе системы не более 9 часов в течение месяца).

### 3. АРХИТЕКТУРА И ИНТЕРФЕЙС

Архитектура веб-приложения представлена на рисунке 3.1.

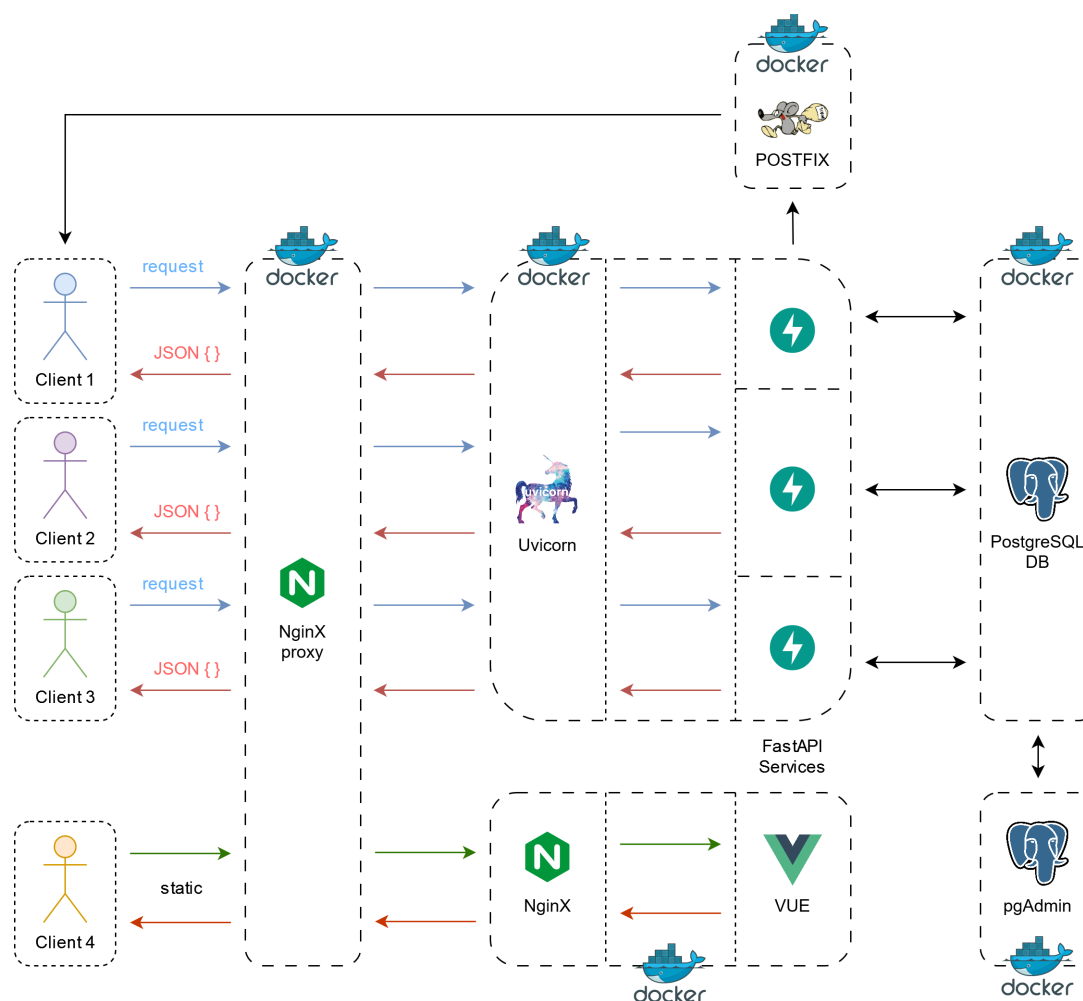


Рисунок 3.1 – Архитектура веб-приложения

Описание задействованных инструментов:

**Nginx (1.27.2)** – веб-сервер, осуществляющий функции обратного прокси-сервера.

**Uvicorn (0.30.6)** – это веб-сервер поддерживающий протокол ASGI для Python.

**FastAPI (0.115.0)** – веб-фреймворк на Python, является основой back-end составляющей веб-сервиса.

**Vue.js (3.5.11)** – JavaScript-фреймворк для создания пользовательских интерфейсов, является основой front-end составляющей веб-сервиса.

**PostgreSQL (17.0)** – объектно-реляционная система управления базами данных.

**PgAdmin (8.3 и выше)** – средство администрирования для PostgreSQL.

**Docker (27.3.1 и выше)** – программное обеспечение для автоматизации развёртывания и управления приложениями в средах с поддержкой контейнеризации, контейнеризатор приложений.

**Postfix (3.9.0 и выше)** – агент передачи почты (MTA – mail transfer agent).

ER-диаграмма базы данных веб-приложения представлена на рисунке 3.2.

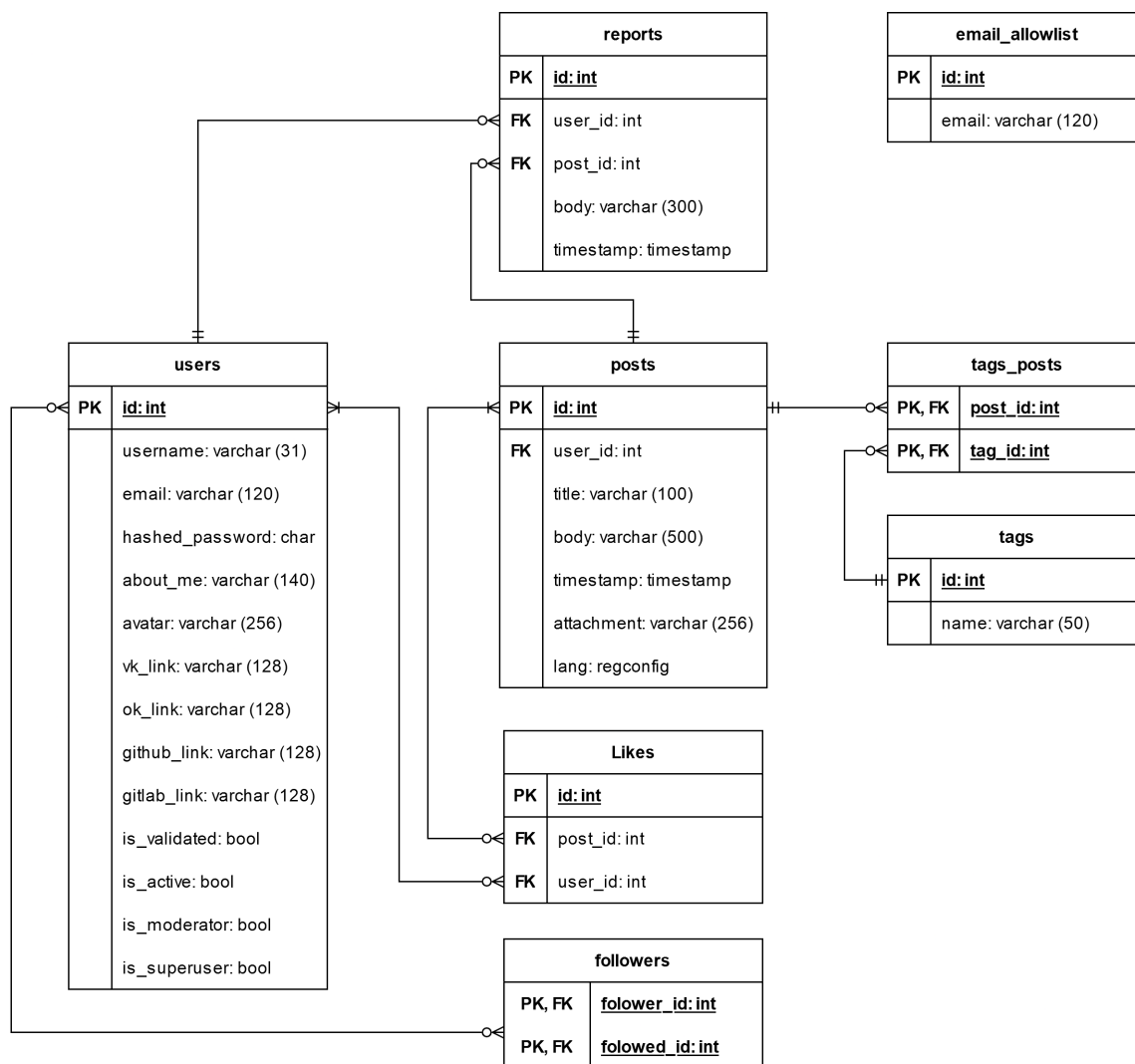


Рисунок 3.2 – ER-диаграмма базы данных

Основные диаграммы бизнес-процессов представлены в репозитории.

Макеты пользовательского интерфейса представлены в документе «Проект дизайна системы» и в п 1.4. настоящего документа. Поля и формы представлены в приложении Б.

Диаграммы вариантов использования представлены в приложении В.

## **4. ДОРОЖНАЯ КАРТА**

### **Фаза 1: Планирование**

- Задача 1: Сбор и анализ требований (13 – 15 сентября 2024)
- Задача 2: Разработка проектного плана (16 – 19 сентября 2024)

### **Фаза 2: Проектирование**

- Задача 1: Создание прототипов и макетов (20 – 25 сентября 2024).
- Задача 2: Утверждение дизайна (26 – 30 сентября 2024).

### **Фаза 3: Разработка**

- Задача 1: Настройка среды исполнения (1 – 5 октября 2024)
- Задача 2: Разработка бэкенда и фронтенда (25 октября – 25 ноября 2024)

### **Фаза 4: Тестирование**

- Задача 1: Функциональное тестирования (26 – 29 ноября 2024)
- Задача 2: Проведение нагрузочного тестирования (30 ноября 2024)

### **Фаза 5: Внедрение**

- Задача 1: Подготовка к запуску (1 – 2 декабря 2024)
- Задача 2: Запуск веб-приложения (3 декабря 2024)

### **Фаза 6: Поддержка**

- Задача 1: Поддержка и исправление ошибок (4 – 5 декабря 2024)
- Задача 2: Выпуск обновлений и улучшений (с 6 декабря 2024)

## **6. РИСКИ И ПЛАН УПРАВЛЕНИЯ РИСКАМИ**

Риски, которые могут возникать во время работы над проектами: задержка в разработке, проблемы с интеграцией, недостаточный уровень опыта.

План управления рисками: обеспечение команды требуемыми источниками знаний, коммуникация с преподавателями образовательного учреждения с целью обеспечения достаточного времени для разработки, резервация времени для решения проблем.

## **7. БЮДЖЕТ И РЕСУРСЫ**

Затраты, возникающие во время работы над проектами: хостинг VPS/VDS сервера, домен и SSL-сертификация.

### **7.1. Оценка стоимости разработки (3 месяца):**

- Стоимость дизайна: 5 тыс. рублей.
- Стоимость разработки (за 3 месяца): 80 тыс. рублей.
- Стоимость тестирования: 5 тыс. рублей.
- Стоимость внедрения: 10 тыс. рублей.

Итого: 100 тыс. рублей.

### **7.2. Дополнительные расходы:**

- Аренда хостинга (на 3 календарных месяца): 1500 рублей.
- Аренда доменного имени (на 12 календарных месяцев): 200 рублей.
- Сертификаты на доменное имя (на 6 календарных месяцев): 300 рублей.

Итого: 2 тыс. рублей.

Управление бюджетом: мониторинг затрат ежемесячно, отчетность.



## **Приложение А. Сбор и анализ требований**

В ходе сбора требований были заданы следующие вопросы:

- **Введение и общие вопросы**

- 1) Опишите идею веб-приложения.
- 2) Какие основные цели создания веб-приложения?
- 3) Кто является целевой аудиторией?
- 4) Какие проблемы или потребности приложение должно решать для пользователей?

- **Функциональные требования**

- 1) Какие ключевые функции должны быть реализованы в приложении?
- 2) Какие функции будут доступны зарегистрированным пользователям?
- 3) Планируется ли функциональность поиска? Если да, то какие параметры поиска являются важными? (требование не критическое)
- 4) Поддерживаемый формат изображений? Какие дополнения к изображениям будут доступны пользователям (хештеги, описания и т.п.)?
- 5) Какие возможности социального взаимодействия вы хотите реализовать (лайки, комментарии, подписки на пользователей и т.д.)?
- 6) Как вы планируете управлять контентом на платформе? (модерация/удаление запрещенного контента)
- 7) Кто будет отвечать за модерацию контента, и какие правила или политики будут использоваться?
- 8) Планируете ли вы интеграцию с внешними базами данных или API для загрузки контента?

- **Нефункциональные требования**

- 1) Есть ли у вас предпочтения для интерфейса?
- 2) Какие у вас требования по производительности?
- 3) Какие у вас требования по безопасности?
- 4) Какие у вас планы по поддержке приложения?
- 5) Какую документацию необходимо составить?

## Приложение Б. Поля и формы

Таблица Б.1 – Обязательные поля регистрации и профиля пользователя

Псевдоним	Псевдоним длиной от 5 до 30 символов, содержащий хотя бы одну латинскую букву и цифры.
Электронная почта	Электронная почта по стандарту RFC5322.
Пароль	Пароль длиной от 6 до 50 символов, содержащий хотя бы одну прописную и строчную латинские буквы, один спец. символ (@\$!%*?&#) и хотя бы одну арабскую цифру.
Повтор пароля	Повторение пароля.

Таблица Б.2 – Поля профиля пользователя

Аватар	Графическое изображение разрешением минимум 256x256 пикселей и максимум 4000x4000 пикселей с соотношением сторон 1:1.
Обо мне	Текстовое поле максимальной длиной в 140 символов
Ссылка на сервис VK	Ссылка вида <a href="https://vk.com/{*}">https://vk.com/{*}</a>
Ссылка на сервис ОК	Ссылка вида <a href="https://ok.ru/profile/{*}">https://ok.ru/profile/{*}</a>
Ссылка на сервис GitHub	Ссылка вида <a href="https://github.com/{*}">https://github.com/{*}</a>
Ссылка на сервис GitLab	Ссылка вида <a href="https://gitlab.com/{*}">https://gitlab.com/{*}</a>

\* – соответствующий правилам сервиса идентификатор пользователя в системе.

Таблица Б.3 – Обязательные поля загрузки и изменения изображения

Изображение	Графическое изображение разрешением минимум 256x256 пикселей и максимум 4000x4000 пикселей с соотношением сторон не более 2 и не менее 0,45.
Заголовок	Текстовое поле длиной до 140 символов.

Таблица Б.4 – Поля загрузки и изменения изображения

Описание	Текстовое поле длиной до 500 символов.
Список тегов	Текстовое поле поиска тегов с системе с функционалом добавления тегов.

Таблица Б.5 – Обязательные поля панели управления

Пользователь	Никнейм пользователя, с возможностью поиска
Тег	Тег, с возможностью поиска

Таблица Б.6 – Обязательные поля окна жалобы

Описание	Текст жалобы длиной до 300 символов.
----------	--------------------------------------

## Приложение В. Диаграммы вариантов использования

Диаграммы вариантов использования представлены на рисунках В.1 – В.4.

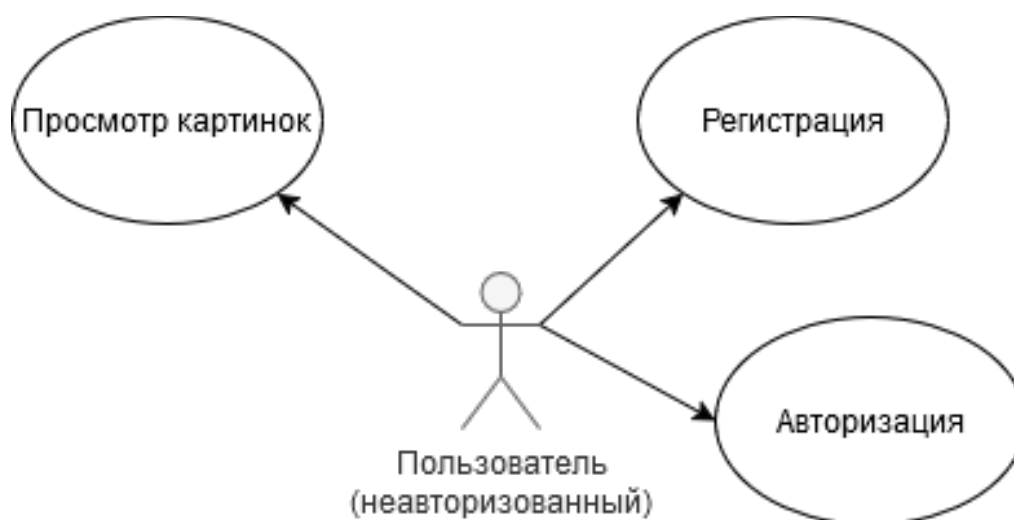


Рисунок В.1 – Диаграмма вариантов использования для неавторизованного пользователя

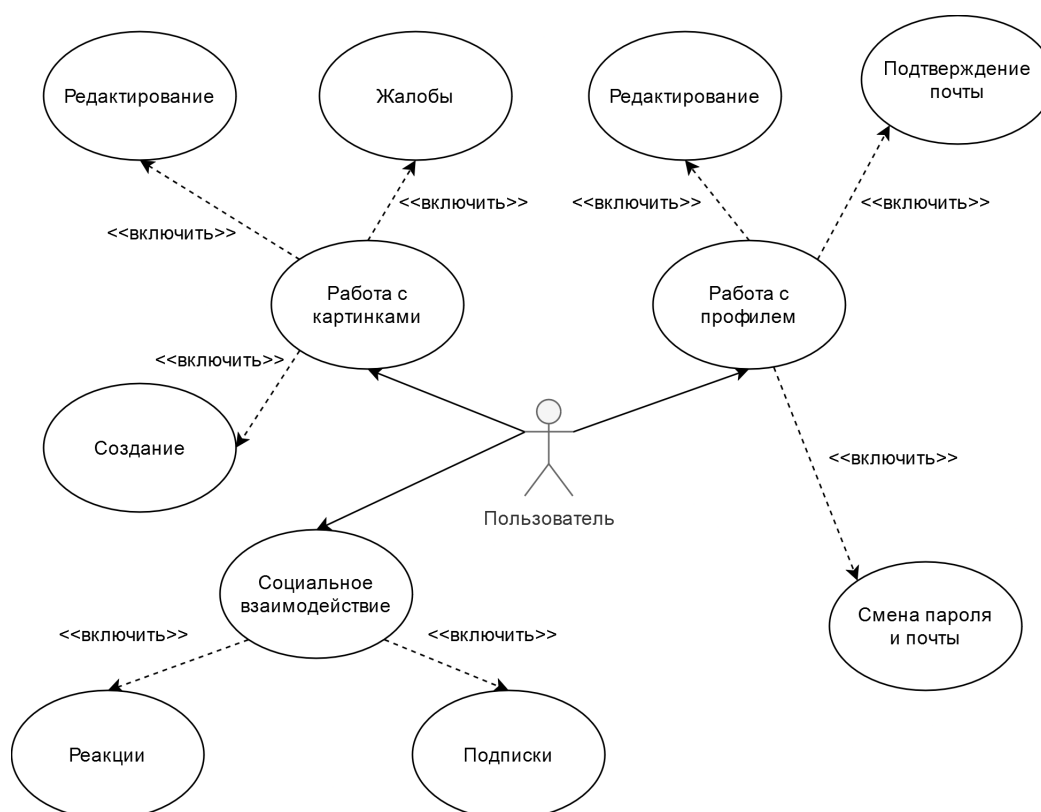


Рисунок В.2 – Диаграмма вариантов использования для авторизованного пользователя

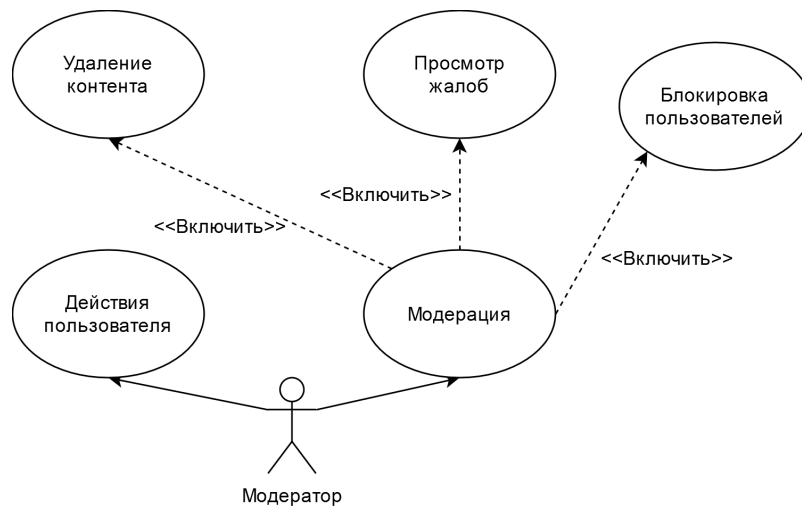


Рисунок В.3 – Диаграмма вариантов использования для модератора

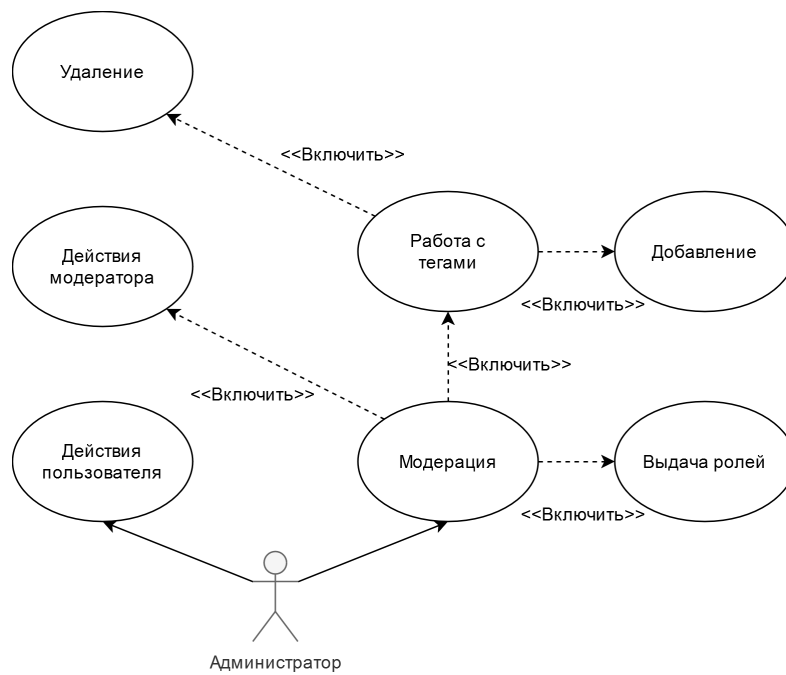


Рисунок В.4 – Диаграмма вариантов использования для администратора