

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования «Петрозаводский государственный университет»

Институт математики и информационных технологий
кафедра прикладная математика и кибернетика

Отчёт по учебному курсу "Верификация программного обеспечения"

09.03.04 - программная инженерия

Выполнил:
студент
Бровкин А.П.

Руководитель
к.ф-м.н, доцент
Кулаков К.А

Оглавление

1	Объект тестирования	4
1.1	Описание веб-приложения	4
1.2	Функции веб-приложения	5
1.3	Архитектура веб-приложения	7
1.4	Описание модулей	8
1.4.1	Модуль регистрации и авторизации	8
1.4.2	Панель администратора	8
1.4.3	Модуль администратора	8
1.4.4	Панель пользователя	9
1.4.5	Модуль пользователя	9
1.4.6	Модуль взаимодействия с базой данных	10
2	Стратегия тестирования	11
2.1	Стратегия блочного тестирования	11
2.2	Стратегия интеграционного тестирования	12
2.3	Стратегия аттестационного тестирования	13
2.4	Стратегия нагрузочного тестирования	14
2.5	Критерии прохождения тестирования	15
2.6	Условия возобновления и приостановки выполнения тестов	16
3	Детальный план тестирования	17
3.1	Блочное тестирования	17
3.1.1	Модуль регистрации и авторизации	17
3.1.2	Панель администратора	20
3.1.3	Модуль администратора	22
3.1.4	Панель пользователя	24
3.1.5	Модуль пользователя	26

3.1.6	База данных	28
3.2	Интеграционное тестирования	30
3.2.1	Интеграция Модуля регистрации и авторизации с Панелью администратора	30
3.2.2	Интеграция Модуля регистрации и авторизации с Панелью пользователя	31
3.2.3	Интеграция Панели администратора с Модулем администратора . . .	32
3.2.4	Интеграция Модуля пользователя с Панелью пользователя	33
3.2.5	Интеграция Модуля администратора с Базой данных	34
3.2.6	Интеграция Модуля пользователя с Базой данных	35
3.3	Аттестационное тестирования	36
3.4	Нагрузочное тестирования	50
3.5	Покрытие кода тестами	56
4	Журнал тестирования	57
5	Примеры тестов	61
6	Прохождение тестирования	66
7	Журнал найденных ошибок	70
8	Результаты	72

Глава 1

Объект тестирования

1.1 Описание веб-приложения

Веб-приложение представляет собой платформу для создания, просмотра и обмена контентом в виде постов и комментариев. Приложение разработано с использованием языка программирования Python и веб-фреймворка Flask, а для хранения данных используется база данных PostgreSQL. В приложении реализованы функции аутентификации и авторизации пользователей, а также административная панель для управления контентом.

Технологический стек:

- Backend: Python, Flask
- База данных: PostgreSQL
- Фронтенд: HTML, CSS
- Шаблонизатор: Jinja2

Веб-приложение предоставляет простой и интуитивно понятный интерфейс для создания и обмена контентом. Оно ориентировано на коммуникацию пользователей, предоставляя им средства для выражения своих идей и взаимодействия с сообществом.

1.2 Функции веб-приложения

- **Аутентификация и Авторизация:**

- Регистрация новых пользователей.
- Вход в систему с использованием учетных данных.
- Выход из системы.

- **Посты и Комментарии:**

- Создание новых постов администратором.
- Просмотр списка последних постов на главной странице.
- Просмотр конкретного поста.
- Комментирование постов пользователем.
- Удаление комментариев администратором (модерация).
- Пагинация для отображения постов.

- **Управление пользователями:**

- Просмотр списка пользователей администратором.
- Изменение прав пользователей администратором.
- Блокировка пользователей администратором.

- **Отправка сообщений на почту:**

- Возможность пользователя отправить сообщение на почту администратору.

- **Изображения:**

- Добавление изображений к постам (опционально).

- **Пользовательская панель:**

- Просмотр постов, принадлежащих текущему пользователю.
- Комментирование постов.
- Отправка сообщений на почту администратору.

- **Административная панель:**

- Управление постами (создание, удаление).
- Модерация комментариев.
- Управление пользователями (изменение прав, блокировка).

- **Обработка ошибок:**

- Предоставление информативных сообщений об ошибках.

- **Безопасность:**

- Защита от SQL-инъекций.
- Защита от межсайтовой подделки запросов (CSRF).
- Шифрование паролей пользователей.

1.3 Архитектура веб-приложения

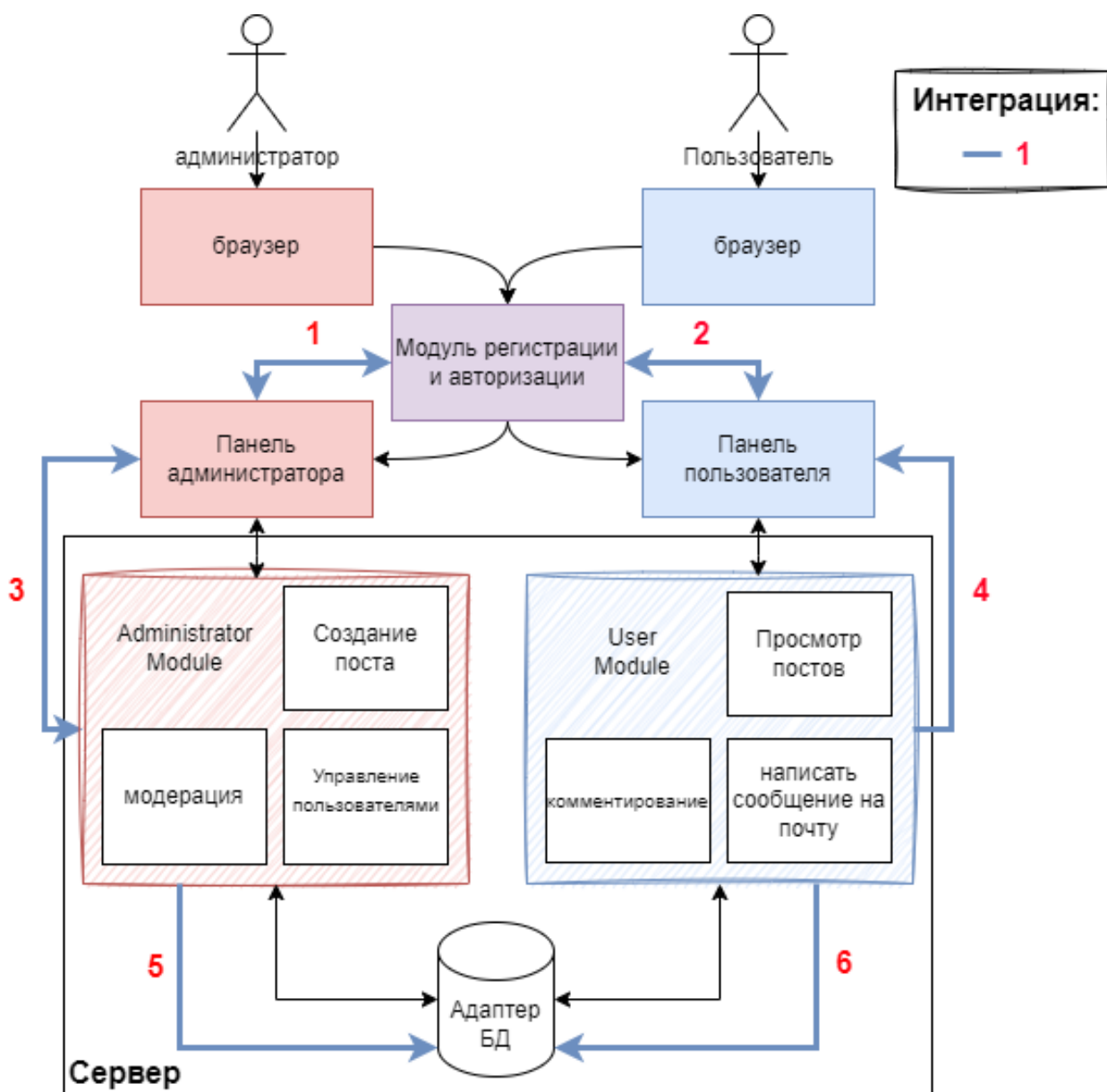


Рис. 1.1: Архитектура веб-приложения

1.4 Описание модулей

1.4.1 Модуль регистрации и авторизации

Класс `UserAuthentication`

Класс, предоставляющий методы для регистрации и аутентификации пользователей.

- `register_user(username, password)`: Метод для регистрации нового пользователя. Принимает имя пользователя и пароль, проводит валидацию введенных данных и сохраняет информацию о пользователе в базе данных.
- `authenticate_user(username, password)`: Метод для аутентификации пользователя. Принимает имя пользователя и пароль, проверяет их соответствие данным в базе данных и возвращает результат аутентификации.
- `validate_input(username, password)`: Метод для валидации введенных данных пользователя. Проверяет корректность формата имени пользователя и пароля.

1.4.2 Панель администратора

Класс `AdminPanel`

Класс, обеспечивающий функции управления постами, модерации и управления пользователями для администратора.

- `manage_posts()`: Метод для управления постами. Предоставляет возможность создания нового поста, редактирования и удаления существующих.
- `moderate_content()`: Метод для модерации контента. Позволяет администратору просматривать и утверждать/отклонять комментарии и посты.
- `manage_users()`: Метод для управления пользователями. Дает доступ к функционалу блокировки/разблокировки пользователей, а также изменения их ролей.

1.4.3 Модуль администратора

Класс `AdminModule`

Класс, обрабатывающий запросы, связанные с административными функциями.

- `create_post(request)`: Метод для создания нового поста. Принимает HTTP-запрос с данными поста, проводит валидацию и сохраняет пост в базе данных.
- `moderate_comment(request)`: Метод для модерации комментария. Принимает HTTP-запрос с данными комментария, проводит модерацию и сохраняет результат в базе данных.
- `manage_user(request)`: Метод для управления пользователями. Принимает HTTP-запрос с данными пользователя и действием (блокировать/разблокировать), проводит необходимые операции и сохраняет изменения в базе данных.

1.4.4 Панель пользователя

Класс `UserPanel`

Класс, предоставляющий функции просмотра постов, комментирования и отправки сообщений пользователями.

- `view_posts()`: Метод для просмотра постов. Возвращает список постов для отображения пользователю.
- `comment_post(post_id, comment)`: Метод для комментирования поста. Принимает идентификатор поста и текст комментария, сохраняет комментарий в базе данных.
- `send_message(request)`: Метод для отправки сообщения. Принимает HTTP-запрос с данными сообщения, проводит валидацию и отправляет сообщение.

1.4.5 Модуль пользователя

Класс `UserModule`

Класс, обрабатывающий запросы, связанные с функциями пользователя.

- `view_post(request)`: Метод для просмотра постов. Принимает HTTP-запрос на просмотр постов, извлекает и возвращает посты для пользователя.
- `comment_post(request)`: Метод для комментирования поста. Принимает HTTP-запрос с идентификатором поста и комментарием пользователя, добавляет комментарий в базу данных.
- `send_email(request)`: Метод для отправки электронного сообщения. Принимает HTTP-запрос с данными электронной почты, проводит валидацию и отправляет сообщение.

1.4.6 Модуль взаимодействия с базой данных

Класс DatabaseHandler

Класс, отвечающий за взаимодействие с базой данных.

- `save_data(data)`: Метод для сохранения данных в базе данных.
- `retrieve_data(query)`: Метод для извлечения данных из базы данных на основе запроса.
- `validate_data(data)`: Метод для валидации данных перед сохранением в базу данных.

Глава 2

Стратегия тестирования

2.1 Стратегия блочного тестирования

Блочное тестирование (или модульное тестирование) — это процесс проверки отдельных блоков программного обеспечения, таких как функции, методы, классы или модули, с целью обеспечения их корректности и работоспособности.

Для блочного тестирования Python-кода: был выбран unittest: Так как он по своему определению был создан для блочных тестов. Как альтернатива было возможно тестирование на pytest в угоду большой мощности и гибкости

Можно выделить следующие основные блоки веб-приложения:

- **Модуль регистрации и авторизации:** Тестируются функции регистрации и аутентификации пользователей, проверяется корректность валидации введенных данных.
- **Панель администратора:** Тестируются функции управления постами, модерации и управления пользователями.
- **Модуль администратора:** Проводится тестирование создания поста, модерации, управления пользователями.
- **Панель пользователя:** Тестируются функции просмотра постов, комментирования и отправки сообщений.
- **Модуль пользователя:** Тестируются функции просмотра постов, комментирования и отправки сообщений.
- **База данных:** Проводится тестирование взаимодействия с базой данных, включая сохранение и извлечение данных.

2.2 Стратегия интеграционного тестирования

Интеграционное тестирование — это вид тестирования, направленный на проверку взаимодействия между различными компонентами системы для обеспечения их корректной работы в совокупности.

Для интеграционного тестирования будут использоваться следующие инструменты: JUnit - Основной инструментом для написания и запуска интеграционных тестов. DbUnit - Используется для тестирования взаимодействия с базой данных. Позволяет подготовить тестовую среду в базе данных перед выполнением тестов.

Можно выделить следующие точки интеграции:

- **Определение точек интеграции:**

- **Интеграция Модуля регистрации и авторизации с Панелью администратора:**

Проверка передачи данных пользователя между этими модулями, а также корректности авторизации администратора.

Функционал включает в себя регистрацию и аутентификацию администратора через соответствующий модуль и последующее взаимодействие с административной панелью.

- **Интеграция Модуля регистрации и авторизации с Панелью пользователя:**

Проверка передачи данных пользователя между этими модулями, а также корректности авторизации пользователя.

Функционал включает в себя регистрацию и аутентификацию обычного пользователя через соответствующий модуль и последующее взаимодействие с пользовательской панелью.

- **Интеграция Панели администратора с Модулем администратора:**

Тестирование взаимодействия для создания поста, модерации и управления пользователями.

Функционал включает в себя возможность администратора создавать посты, проводить модерацию контента и управлять пользователями через административную панель.

- **Интеграция Модуля пользователя с Панелью пользователя:**

Проверка корректности отображения постов, комментариев и отправки сообщений пользователями.

Функционал включает в себя возможность пользователя просматривать посты, комментировать и отправлять сообщения через пользовательскую панель.

– **Интеграция Модуля администратора с Базой данных:**

Тестирование передачи и сохранения данных в базе данных при взаимодействии с модулями.

Функционал включает в себя взаимодействие модуля администратора с базой данных для сохранения и извлечения информации о пользователях, постах и других административных данных.

– **Интеграция Модуля пользователя с Базой данных:**

Тестирование передачи и сохранения данных в базе данных при взаимодействии с модулями.

Функционал включает в себя взаимодействие модуля пользователя с базой данных для сохранения и извлечения информации о пользователях, постах, комментариях и других данных, необходимых для функционирования пользовательской части.

2.3 Стратегия аттестационного тестирования

Аттестационное тестирование — это вид тестирования, направленный на проверку соответствия веб-приложения заявленным требованиям и оценку его функциональных возможностей.

Для написания аттестационных тестов будут использоваться следующая библиотека: JUnit - стандартным инструментом для написания и запуска тестов. В данном случае, JUnit будет использоваться для создания тестовых сценариев, охватывающих различные функциональные возможности приложения.

Можно выделить следующие функциональные возможности:

- **Регистрация и авторизация:** Проверка корректности процесса регистрации и аутентификации пользователей.
- **Управление контентом:** Тестирование функционала создания, редактирования и удаления постов.

- **Комментирование:** Проверка возможности пользователей добавлять и удалять комментарии к постам.
- **Панель администратора:** Тестирование функций управления пользователями и модерации контента.

Основной упор делается на обеспечение корректной работы всех ключевых функциональных возможностей веб-приложения.

2.4 Стратегия нагрузочного тестирования

Нагрузочное тестирование — это вид тестирования, направленный на оценку стабильности и производительности системы при различных условиях нагрузки.

Для написания нагрузочных тестов будет использоваться библиотека - Apache JMeter - это мощный инструмент с открытым исходным кодом для проведения нагрузочного тестирования. Поддерживает разнообразные протоколы, включая HTTP, HTTPS, JDBC, FTP, и многие другие.

- Проверка работоспособности при появлении новых пользователей до достижения определенного количества. Разделение нагрузки между главной страницей и страницей первой темы.
- Оценка производительности системы при постепенном нарастании нагрузки до определенного числа пользователей в течение заданного времени.
- Проверка устойчивости под статической нагрузкой с постоянным числом активных пользователей.
- Оценка максимальной нагрузки системы, достигая определенного числа пользователей за короткий период.
- Проверка способности системы восстановления после периодов высокой нагрузки, следующих после низкой активности.
- Оценка производительности системы при одновременном выполнении разнообразных действий: просмотр страниц, отправка сообщений, создание постов.

Каждый тест будет направлен на проверку определенного аспекта стабильности и производительности системы под различными сценариями нагрузки.

2.5 Критерии прохождения тестирования

- **Критерии аутентификации и авторизации:**

- **Цель:** Убедиться в правильной и безопасной аутентификации пользователей.

- **Критерии:**

- * Возможность успешного входа в систему с правильными учетными данными.
- * Отсутствие возможности входа с неправильными учетными данными.
- * Корректная обработка запросов на восстановление пароля.

- **Критерии управления постами (администратор):**

- **Цель:** Проверить функциональность управления постами администратором.

- **Критерии:**

- * Создание нового поста через административную панель.
- * Успешное отображение нового поста на главной странице.
- * Возможность редактирования и удаления существующих постов.

- **Критерии просмотра постов и комментирования (пользователь):**

- **Цель:** Проверить функциональность просмотра и комментирования постов пользователем.

- **Критерии:**

- * Отображение списка последних постов на главной странице.
- * Возможность комментирования постов.
- * Корректное отображение комментариев.

- **Критерии обработки ошибок:**

- **Цель:** Гарантировать правильное и информативное уведомление об ошибках.

- **Критерии:**

- * Отображение четких и понятных сообщений об ошибках при попытке выполнения некорректных действий.
- * Запись логов ошибок для последующего анализа и улучшения системы.

- **Критерии производительности:**

- **Цель:** Оценить производительность системы под различными нагрузками.
- **Критерии:**
 - * Время отклика системы на различные запросы не должно превышать установленные пороговые значения.
 - * Пропускная способность системы должна обеспечивать комфортное использование приложения.

2.6 Условия возобновления и приостановки выполнения тестов

- **Условия возобновления выполнения тестов:**

- **Устранение критических ошибок:** Если в процессе тестирования выявляются критические ошибки, их исправление и успешная повторная верификация становятся условием для возобновления тестирования.

- **Условия приостановки выполнения тестов:**

- **Выявление критических ошибок:** Если в процессе тестирования выявляются критические ошибки, тестирование может быть приостановлено до их устранения.
- **Неустойчивость системы:** Если система проявляет неустойчивость или некорректное поведение под нагрузкой, тестирование может быть приостановлено для выявления и устранения проблем.
- **Отсутствие необходимых ресурсов:** В случае отсутствия необходимых ресурсов (тестовых данных, окружения и т. д.), тестирование может быть временно приостановлено до предоставления необходимых условий.

Глава 3

Детальный план тестирования

3.1 Блочное тестирования

3.1.1 Модуль регистрации и авторизации

Тест В1, Позитивный тест: Регистрация пользователя с корректными данными

Объект тестирования	Модуль регистрации и авторизации
Цель	Проверка успешной регистрации пользователя с корректными данными.
Описание	Тестирование процесса регистрации пользователя с использованием корректных логина и пароля.
Начальное состояние	Пользователь не зарегистрирован в системе.
Входные данные	Логин: 'testUser' Пароль: 'testPassword'
Ожидаемый результат	Пользователь успешно зарегистрирован.

Воспроизводимые шаги:

1. Перейти на страницу регистрации.
2. Заполнить корректные данные.

3. Нажать кнопку "Зарегистрироваться".

Тест В2, Негативный тест: Попытка регистрации с некорректными данными

Объект тестирования	Модуль регистрации и авторизации
Цель	Проверка обработки ошибок при попытке регистрации с некорректными данными.
Описание	Тестирование системы при вводе некорректных данных (слишком короткий логин и пароль).
Начальное состояние	Пользователь не зарегистрирован в системе.
Входные данные	Логин: 'us' Пароль: 'pass'
Ожидаемый результат	Система корректно обрабатывает ошибки валидации и сообщает об ошибке.

Воспроизводимые шаги:

1. Перейти на страницу регистрации.
2. Заполнить некорректные данные.
3. Нажать кнопку "Зарегистрироваться".

Тест В3, Позитивный тест: Аутентификация зарегистрированного пользователя

Объект тестирования	Модуль регистрации и авторизации
Цель	Проверка успешной аутентификации зарегистрированного пользователя.
Описание	Тестирование системы при вводе корректных данных для аутентификации зарегистрированного пользователя.
Начальное состояние	Зарегистрированный пользователь.
Входные данные Пароль: 'testPassword'	Логин: 'testUser'
Ожидаемый результат	Пользователь успешно аутентифицирован и получает доступ к своей учетной записи.

Воспроизводимые шаги:

1. Перейти на страницу авторизации.
2. Ввести корректные учетные данные.
3. Нажать кнопку "Войти".

3.1.2 Панель администратора

Тест В4, Позитивный тест: Создание нового поста администратором

Объект тестирования	Панель администратора, Модуль администратора
Цель	Проверка успешного создания и публикации нового поста администратором.
Описание	Тестирование системы при вводе корректных данных для создания нового поста администратором.
Начальное состояние	Войти в систему как администратор.
Входные данные	Логин администратора: 'admin', Пароль администратора: 'adminPassword', Заголовок поста: 'Test Post', Содержание поста: 'Test Content', URL изображения: 'test.jpg'
Ожидаемый результат	Пост успешно создан и опубликован на веб-приложении.

Воспроизводимые шаги:

1. Войти в систему как администратор.
2. Перейти в раздел управления постами.
3. Нажать кнопку "Создать пост" и заполнить необходимые поля.
4. Нажать кнопку "Опубликовать".

Тест В5, Позитивный тест: Модерация комментария пользователя

Объект тестирования	Панель администратора, Модуль администратора
Цель	Проверка успешной модерации комментария администратором.
Описание	Тестирование системы при модерации комментария пользователя.
Начальное состояние	Войти в систему как администратор.
Входные данные	Логин администратора: 'admin', Пароль администратора: 'adminPassword', Идентификатор комментария: 123 (предположим)
Ожидаемый результат	Модерация комментария прошла успешно.

Воспроизводимые шаги:

1. Войти в систему как администратор.
2. Перейти в раздел модерации комментариев.
3. Найти нужный комментарий и применить модерацию.

3.1.3 Модуль администратора

Тест В6, Позитивный тест: Создание нового поста от имени администратора

Объект тестирования	Панель администратора, Модуль администратора
Цель	Проверка успешного создания и публикации нового поста от имени администратора.
Описание	Тестирование системы при создании нового поста администратором.
Начальное состояние	Войти в систему как администратор.
Входные данные	Логин администратора: 'admin', Пароль администратора: 'adminPassword', Заголовок поста: 'Test Post', Содержание поста: 'Test Content', URL изображения: 'test.jpg'
Ожидаемый результат	Пост успешно создан и опубликован от имени администратора.

Воспроизводимые шаги:

1. Войти в систему как администратор.
2. Перейти в раздел создания поста.
3. Заполнить необходимые поля и нажать кнопку "Опубликовать".

Тест В7, Позитивный тест: Управление пользователями (блокировка, разблокировка)

Объект тестирования	Панель администратора, Модуль администратора
Цель	Проверка успешного выполнения управления пользователями (блокировка, разблокировка) администратором.
Описание	Тестирование системы при выполнении администратором действий по блокировке или разблокировке пользователя.
Начальное состояние	Войти в систему как администратор.
Входные данные	Логин администратора: 'admin', Пароль администратора: 'adminPassword', Логин пользователя: 'testUser', Действие: Блокировка
Ожидаемый результат	Управление пользователями прошло успешно.

Воспроизводимые шаги:

1. Войти в систему как администратор.
2. Перейти в раздел управления пользователями.
3. Выполнить блокировку или разблокировку пользователя.

3.1.4 Панель пользователя

Тест В8, Позитивный тест: Просмотр списка постов

Объект тестирования	Панель пользователя, Модуль пользователя
Цель	Проверка успешного просмотра списка постов пользователем.
Описание	Тестирование системы при просмотре обычным пользователем списка постов.
Начальное состояние	Войти в систему как обычный пользователь.
Входные данные	Логин пользователя: 'testUser', Пароль пользователя: 'testPassword'
Ожидаемый результат	Пользователь успешно просматривает список постов.

Воспроизводимые шаги:

1. Войти в систему как обычный пользователь.
2. Перейти на страницу списка постов.

Тест В9, Позитивный тест: Отправка комментария к посту

Объект тестирования	Панель пользователя, Модуль пользователя
Цель	Проверка успешной отправки комментария к посту пользователем.
Описание	Тестирование системы при отправке обычным пользователем комментария к посту.
Начальное состояние	Войти в систему как обычный пользователь.
Входные данные	Логин пользователя: 'testUser', Пароль пользователя: 'testPassword', Комментарий: 'Test Comment', Идентификатор поста: 1 (предположим)
Ожидаемый результат	Комментарий успешно отправлен.

Воспроизводимые шаги:

1. Войти в систему как обычный пользователь.
2. Перейти на страницу поста и оставить комментарий.
3. Нажать кнопку "Отправить".

3.1.5 Модуль пользователя

Тест В10, Позитивный тест: Просмотр списка постов

Объект тестирования	Панель пользователя, Модуль пользователя
Цель	Проверка успешного просмотра списка постов обычным пользователем.
Описание	Тестирование системы при просмотре обычным пользователем списка постов.
Начальное состояние	Войти в систему как обычный пользователь.
Входные данные	Логин пользователя: 'testUser', Пароль пользователя: 'testPassword'
Ожидаемый результат	Пользователь успешно просматривает список постов.

Воспроизводимые шаги:

1. Войти в систему как обычный пользователь.
2. Перейти на страницу списка постов.

Тест В11, Позитивный тест: Отправка комментария к посту

Объект тестирования	Панель пользователя, Модуль пользователя
Цель	Проверка успешной отправки комментария обычным пользователем к посту.
Описание	Тестирование системы при отправке обычным пользователем комментария к посту.
Начальное состояние	Войти в систему как обычный пользователь.
Входные данные	Логин пользователя: 'testUser', Пароль пользователя: 'testPassword', Комментарий: 'Test Comment', Идентификатор поста: 1 (предположим)
Ожидаемый результат	Комментарий успешно отправлен.

Воспроизводимые шаги:

1. Войти в систему как обычный пользователь.
2. Перейти на страницу поста и оставить комментарий.
3. Нажать кнопку "Отправить".

3.1.6 База данных

Тест В12, Позитивный тест: Сохранение данных о новом пользователе

Объект тестирования	Модуль регистрации и авторизации, База данных
Цель	Проверка успешного сохранения данных о новом пользователе в базе данных.
Описание	Тестирование системы при регистрации нового пользователя и проверка сохранения данных в базе данных.
Входные данные	Логин пользователя: 'newUser', Пароль пользователя: 'password123'
Ожидаемый результат	Данные успешно сохранены.

Воспроизводимые шаги:

1. Зарегистрировать нового пользователя.
2. Проверить наличие данных в базе данных.

Тест В13, Позитивный тест: Загрузка списка постов для отображения

Объект тестирования	Панель пользователя, База данных
Цель	Проверка успешной загрузки списка постов для отображения на странице пользователя.
Описание	Тестирование функционала загрузки постов и проверка, что данные успешно извлечены из базы данных.
Входные данные	Отсутствуют (тестирование функционала загрузки постов).
Ожидаемый результат	Данные успешно загружены и отображены.

Воспроизводимые шаги:

1. Загрузить страницу списка постов.
2. Проверить, что данные загружены из базы данных.

3.2 Интеграционное тестирования

3.2.1 Интеграция Модуля регистрации и авторизации с Панелью администратора

Тест П1, Позитивный тест: Передача данных пользователя и авторизация администратора

Описание	Проверка передачи данных пользователя между модулем регистрации и авторизации и панелью администратора, а также корректности авторизации администратора.
Входные данные	Отсутствуют.
Ожидаемый результат	Успешная передача данных и корректная авторизация администратора.

Шаги теста:

1. Передача данных пользователя из модуля регистрации и авторизации в панель администратора.
2. Попытка авторизации администратора с корректными данными.
3. Проверка успешной авторизации и взаимодействия между модулем и панелью.

3.2.2 Интеграция Модуля регистрации и авторизации с Панелью пользователя

Тест I2, Негативный тест: Попытка авторизации с некорректными данными

Описание	Проверка системы на обработку некорректных данных при попытке авторизации пользователя.
Входные данные	Некорректный логин и пароль пользователя.
Ожидаемый результат	Система корректно обрабатывает некорректные данные и предоставляет сообщение об ошибке.

Шаги теста:

1. Передача некорректных данных пользователя из модуля регистрации и авторизации в панель пользователя.
2. Попытка авторизации пользователя с некорректными данными.
3. Проверка системного ответа на некорректные данные.

3.2.3 Интеграция Панели администратора с Модулем администратора

Тест I3, Позитивный тест: Взаимодействие между панелью администратора и модулем администратора

Описание	Тестирование взаимодействия для создания поста, модерации и управления пользователями между панелью администратора и модулем администратора.
Входные данные	Отсутствуют.
Ожидаемый результат	Успешная передача данных и корректное взаимодействие между панелью администратора и модулем администратора.

Шаги теста:

1. Передача данных о создании поста из модуля администратора в панель администратора.
2. Модерация поста на панели администратора.
3. Управление пользователями на панели администратора.
4. Проверка успешного взаимодействия между модулем и панелью администратора.

3.2.4 Интеграция Модуля пользователя с Панелью пользователя

Тест I4, Позитивный тест: Взаимодействие между панелью пользователя и модулем пользователя

Описание	Проверка корректности отображения постов, комментариев и отправки сообщений пользователями между модулем пользователя и панелью пользователя.
Входные данные	Отсутствуют.
Ожидаемый результат	Успешная передача данных и корректное взаимодействие между панелью пользователя и модулем пользователя.

Шаги теста:

1. Передача данных о постах и комментариях из модуля пользователя в панель пользователя.
2. Отправка сообщения на панели пользователя.
3. Проверка успешного взаимодействия между модулем и панелью пользователя.

3.2.5 Интеграция Модуля администратора с Базой данных

Тест I5, Позитивный тест: Взаимодействие с базой данных при создании поста через модуль администратора

Описание	Тестирование передачи и сохранения данных в базе данных при взаимодействии с модулем администратора.
Входные данные	Логин администратора: 'admin', Пароль администратора: 'adminPassword', Заголовок поста: 'Test Post', Содержание поста: 'Test Content', URL изображения: 'test.jpg'
Ожидаемый результат	Успешное сохранение данных в базе данных при взаимодействии с модулем администратора.

Шаги теста:

1. Создание поста с модулем администратора.
2. Проверка сохранения данных в базе данных.

3.2.6 Интеграция Модуля пользователя с Базой данных

Тест I6, Позитивный тест: Взаимодействие с базой данных при отправке сообщения через модуль пользователя

Описание	Тестирование передачи и сохранения данных в базе данных при взаимодействии с модулем пользователя.
Входные данные	Логин пользователя: 'testUser', Пароль пользователя: 'testPassword', Комментарий: 'Test Comment', Идентификатор поста: 1 (предположим)
Ожидаемый результат	Успешное сохранение данных в базе данных при взаимодействии с модулем пользователя.

Шаги теста:

1. Отправка сообщения с модуля пользователя.
2. Проверка сохранения данных в базе данных.

3.3 Аттестационное тестирования

Тест А1, Позитивный тест: Отображение страницы регистрации в различных браузерах

Начальное состояние	Система запущена, браузеры установлены.
Описание теста	Проверка отображения страницы регистрации в различных браузерах, ввод url-адреса в адресной строке.
Входные данные	Ввод url-адреса в адресную строку браузера: <code>http://localhost</code>
Ожидаемый результат	Отображение страницы регистрации во всех имеющихся браузерах.

Тест А2, Позитивный тест: Навигация на сайте - нажатие на различные элементы навигационного меню

Начальное состояние	Система запущена, страница сайта открыта.
Описание теста	Проверка навигации на сайте при нажатии на различные элементы навигационного меню.
Входные данные	Набор ссылок на элементы навигационного меню.
Ожидаемый результат	Переход к соответствующим элементам сайта при нажатии на ссылки.

Тест А3, Позитивный тест: Регистрация нового пользователя

Начальное состояние	Система запущена, пользователь на странице регистрации.
Описание теста	Проверка процесса успешной регистрации нового пользователя.
Входные данные	Корректные данные нового пользователя (логин, пароль, электронная почта).
Ожидаемый результат	Новый пользователь успешно зарегистрирован и может войти в систему.

Тест А4, Позитивный тест: Вход зарегистрированного пользователя в систему

Начальное состояние	Система запущена, пользователь на странице входа.
Описание теста	Проверка процесса входа зарегистрированного пользователя в систему.
Входные данные	Данные зарегистрированного пользователя (логин, пароль).
Ожидаемый результат	Зарегистрированный пользователь успешно вошел в систему.

Тест А5, Позитивный тест: Создание нового поста

Начальное состояние	Пользователь успешно вошел в систему и находится на странице создания нового поста.
Описание теста	Проверка корректности создания нового поста пользователем.
Входные данные	Текст и изображения для нового поста.
Ожидаемый результат	Пост успешно создан и отображается на сайте.

Тест А6, Позитивный тест: Редактирование существующего поста

Начальное состояние	Пользователь успешно вошел в систему и находится на странице редактирования существующего поста.
Описание теста	Проверка возможности редактирования существующего поста.
Входные данные	Новый текст и изображения для редактирования существующего поста.
Ожидаемый результат	Пост успешно отредактирован, и изменения отображаются на сайте.

Тест А7, Позитивный тест: Добавление комментария к посту

Начальное состояние	Пользователь успешно вошел в систему и находится на странице существующего поста для добавления комментария.
Описание теста	Проверка возможности добавления комментария к существующему посту.
Входные данные	Текст комментария.
Ожидаемый результат	Комментарий успешно добавлен и отображается у соответствующего поста.

Тест А8, Позитивный тест: Управление пользователями администратором

Начальное состояние	Пользователь успешно вошел в систему как администратор.
Описание теста	Проверка функционала управления пользователями администратором (блокировка, удаление).
Входные данные	Вход в систему как администратор.
Ожидаемый результат	Администратор успешно осуществляет управление пользователями (блокировка, удаление).

Тест А9, Негативный тест: Попытка входа с неверными данными

Начальное состояние	Пользователь находится на странице входа в систему.
Описание теста	Проверка системы на предотвращение входа с неверными данными.
Входные данные	Неверные данные пользователя (логин, пароль).
Ожидаемый результат	Система отображает сообщение об ошибке и не позволяет войти.

Тест А10, Негативный тест: Попытка создания поста без текста

Начальное состояние	Пользователь находится на странице создания поста.
Описание теста	Проверка системы на предотвращение создания поста без ввода текста.
Входные данные	Создание поста без ввода текста.
Ожидаемый результат	Система отображает сообщение об ошибке и не создает пустой пост.

Тест А11, Негативный тест: Попытка регистрации с уже существующим логином

Начальное состояние	Пользователь находится на странице регистрации.
Описание теста	Проверка системы на предотвращение регистрации с уже существующим логином.
Входные данные	Регистрация с логином, который уже занят.
Ожидаемый результат	Система отображает сообщение об ошибке и не позволяет зарегистрировать пользователя с занятым логином.

Тест А12, Негативный тест: Попытка редактирования чужого поста

Начальное состояние	Пользователь авторизован и находится на странице редактирования поста.
Описание теста	Проверка системы на предотвращение попытки редактирования поста, созданного другим пользователем.
Входные данные	Попытка редактирования поста, принадлежащего другому пользователю.
Ожидаемый результат	Система отображает сообщение об ошибке и не позволяет редактировать пост, принадлежащий другому пользователю.

Тест А13, Негативный тест: Попытка добавления комментария без текста

Начальное состояние	Пользователь авторизован и находится на странице существующего поста.
Описание теста	Проверка системы на предотвращение добавления комментария без текста.
Входные данные	Попытка добавления комментария без ввода текста.
Ожидаемый результат	Система отображает сообщение об ошибке и не добавляет пустой комментарий.

Тест А14, Негативный тест: Попытка управления пользователями не администратором

Начальное состояние	Пользователь авторизован, но не является администратором.
Описание теста	Проверка системы на предотвращение управления пользователями не администратором.
Входные данные	Попытка управления пользователями, не входя в систему как администратор.
Ожидаемый результат	Система отображает сообщение об ошибке и не позволяет пользователю управлять пользователями как администратор.

3.4 Нагрузочное тестирования

Стабильность - способность системы поддерживать устойчивое и надежное поведение при различных условиях, включая изменения нагрузки, внешние воздействия и долгосрочное использование.

Тест L1, Нагрузочный тест: Появление новых пользователей

Тип нагрузки	Общая
Начальное состояние	Система функционирует в нормальном режиме
Описание теста	Проверка работоспособности сервиса при появлении 4 новых пользователей в секунду, до тех пор, пока не наберется 150 человек. Присутствует разделение нагрузки: 2 пользователя на главную страницу и 2 пользователя на страницу первой темы.
Входные данные	Каждую секунду имитируется появление 4 новых пользователей.
Ожидаемый результат	Система остается стабильной, обеспечивая приемлемое время отклика и пропускную способность при достижении 150 пользователей.

Тест L2, Нагрузочный тест: Постепенное нарастание нагрузки

Тип нагрузки	Общая
Начальное состояние	Система функционирует в нормальном режиме
Описание теста	Оценка производительности системы при постепенном нарастании нагрузки до 200 пользователей в течение 10 минут.
Входные данные	Постепенное увеличение числа пользователей с начального уровня до 200 в течение 10 минут.
Ожидаемый результат	Система обеспечивает стабильную работу, поддерживая приемлемые метрики производительности в течение всего теста.

Тест L3, Тест на устойчивость: Статическая нагрузка

Тип нагрузки	Статическая
Начальное состояние	Система функционирует в нормальном режиме
Описание теста	Проверка устойчивости системы под статической нагрузкой в течение 1 часа с постоянным числом активных пользователей (100).
Входные данные	Поддержание постоянного числа активных пользователей (100) в течение 1 часа.
Ожидаемый результат	Система стабильна и поддерживает устойчивую производительность в условиях статической нагрузки.

Тест L4, Тест на производительность: Пиковая нагрузка

Тип нагрузки	Пиковая
Начальное состояние	Система функционирует в нормальном режиме
Описание теста	Оценка максимальной нагрузки системы, достигая числа пользователей 300 в течение короткого времени (5 минут).
Входные данные	Максимальное быстрое увеличение числа пользователей до 300 в течение 5 минут.
Ожидаемый результат	Система поддерживает работоспособность и производительность при пиковой нагрузке, не приводя к критическим сбоям.

Тест L5, Тест на производительность: Изменение нагрузки

Тип нагрузки	Изменение нагрузки
Начальное состояние	Система функционирует в нормальном режиме после периода низкой активности
Описание теста	Проверка способности системы восстановления после кратковременных периодов высокой нагрузки, следующих после низкой активности.
Входные данные	Период высокой нагрузки (200 пользователей) после периода низкой активности (10 минут).
Ожидаемый результат	Система успешно восстанавливается и поддерживает стабильную работу после периода высокой нагрузки.

Тест L6, Тест на производительность: Смешанная нагрузка

Тип нагрузки	Смешанная нагрузка
Начальное состояние	Система функционирует в нормальном режиме
Описание теста	Оценка производительности системы при одновременном выполнении разнообразных действий: просмотр страниц, отправка сообщений, создание постов.
Входные данные	Параллельное выполнение различных действий 150 пользователями в течение 15 минут.
Ожидаемый результат	Система эффективно обрабатывает разнообразные действия пользователей, поддерживая производительность на приемлемом уровне.

3.5 Покрытие кода тестами

$$T_{cov} = \left(\frac{Ltc}{Lcode} \right) \times 100\%$$

где

T_{cov} - тестовое покрытие

Ltc - количество строк кода покрытое тестами

$Lcode$ - Общее количество строк кода

Тогда: $T_{cov} = (403/847) * 100\% = 47,5\%$

Глава 4

Журнал тестирования

Номер теста	Фактический результат	Результат теста	Ошибка
A-1	Отображение страницы во всех браузерах	Позитивный	
A-2	Переход к соответствующим элементам	Позитивный	
A-3	Успешная регистрация нового пользователя	Позитивный	
A-4	Успешный вход в систему	Позитивный	
A-5	Успешное создание и отображение поста	Позитивный	№2
A-6	Успешное редактирование и отображение изменений	Позитивный	
A-7	Успешное добавление и отображение комментария	Позитивный	
A-8	Успешное управление пользователями	Позитивный	
A-9	Отображение сообщения об ошибке	Негативный	
A-10	Отображение сообщения об ошибке	Негативный	№3
A-11	Отображение сообщения об ошибке	Негативный	
A-12	Отображение сообщения об ошибке	Негативный	
A-13	Отображение сообщения об ошибке	58 Негативный	

Номер теста	Фактический результат	Результат теста	Ошибка
В-1	Успешная регистрация	Позитивный	
В-2	Ошибка валидации	Негативный	
В-3	Успешный вход в систему	Позитивный	№1
В-4	Пост успешно создан и опубликован	Позитивный	
В-5	Модерация прошла успешно	Позитивный	
В-6	Пост успешно создан и опубликован	Позитивный	
В-7	Успешное управление пользователями	Позитивный	
В-8	Успешный просмотр списка постов	Позитивный	
В-9	Комментарий успешно отправлен	Позитивный	
В-10	Успешный просмотр списка постов	Позитивный	
В-11	Комментарий успешно отправлен	Позитивный	
В-12	Данные успешно сохранены	Позитивный	
В-13	Данные успешно загружены и отображены	Позитивный	

Номер теста	Фактический результат	Результат теста	Входные данные	Ошибка
I-1	Выполнена передача данных и успешная авторизация	Позитивный	Корректные данные пользователя	
I-2	Выполнена передача данных и успешная авторизация	Позитивный	Корректные данные пользователя	
I-3	Взаимодействие прошло успешно	Позитивный	Данные создания поста, модерации и управления пользователями	№4
I-4	Взаимодействие прошло успешно	Позитивный	Данные постов, комментариев и сообщений	
I-5	Данные успешно переданы и сохранены в базе данных	Позитивный	Данные создания поста	
I-6	Данные успешно переданы и сохранены в базе данных	Позитивный	Данные отправки сообщения	

Номер теста	Фактический результат	Результат теста	Входные данные	Ошибка
L-1	Система стабильна при 150 пользователях	Общий	Появление 4 новых пользователей/с	
L-2	Система стабильна при 200 пользователях	Общий	Постепенное увеличение пользователей	
L-3	Система стабильна в течение 1 часа	Статическая нагрузка	Поддержание 100 активных пользователей	
L-4	Система стабильна при пиковой нагрузке	Пиковая нагрузка	Быстрое увеличение до 300 пользователей	№5
L-5	Система успешно восстанавливается	Период высокой нагрузки после низкой активности		
L-6	Система обрабатывает действия на приемлемом уровне	Смешанная нагрузка	Параллельное выполнение различных действий	

Глава 5

Примеры тестов

```
class WebAppTest(unittest.TestCase):
    def setUp(self):
        # Инициализация драйвера
        self.driver = webdriver.Chrome(executable_path='./chromedriver.exe')

    def tearDown(self):
        # Завершение работы драйвера после теста
        self.driver.quit()

    def test_successful_authentication(self):
        # Воспроизводимые шаги
        # 1. Перейти на страницу авторизации
        self.driver.get(
            "http://localhost/login")

        # 2. Ввести корректные учетные данные
        username_input = self.driver.find_element_by_name("username")
        password_input = self.driver.find_element_by_name("password")

        username_input.send_keys("testUser")
        password_input.send_keys("testPassword")

        # 3. Нажать кнопку "Войти"
        login_button = self.driver.find_element_by_xpath("//button[@type='submit']")
        login_button.click()
```

Рис. 5.1: Test В3, аутентификации зарегистрированного пользователя

```

class WebAppTest(unittest.TestCase):
    def setUp(self):
        # Инициализация драйвера
        self.driver = webdriver.Chrome(executable_path='путь_к_драйверу/chromedriver.exe')

    def tearDown(self):
        # Завершение работы драйвера после теста
        self.driver.quit()

    def test_successful_comment_submission(self):
        # Воспроизводимые шаги
        # 1. Перейти на страницу поста
        post_url = "http://localhost/posts/1"
        self.driver.get(post_url)

        # 2. Оставить комментарий
        comment_input = self.driver.find_element_by_name("comment")
        comment_input.send_keys("Test Comment")

        # 3. Нажать кнопку "Отправить"
        submit_button = self.driver.find_element_by_xpath("//button[@type='submit']")
        submit_button.click()

        # Добавим небольшую паузу, чтобы убедиться, что страница успевает обработать запрос
        time.sleep(2)

```

Рис. 5.2: Test B9, отправка комментария к посту

```

class TestAdminIntegration(unittest.TestCase):

    def setUp(self):
        # Предварительные настройки для тестов
        self.admin_module = AdminModule()
        self.database_module = DatabaseModule()

    def test_create_post_and_verify_database(self):
        # Входные данные
        admin_username = 'admin'
        admin_password = 'adminPassword'
        post_title = 'Test Post'
        post_content = 'Test Content'
        image_url = 'test.jpg'

        # Шаг 1: Создание поста с модулем администратора
        post_id = self.admin_module.create_post(admin_username, admin_password, post_title, post_content, image_url)

        # Шаг 2: Проверка сохранения данных в базе данных
        post_from_database = self.database_module.get_post_by_id(post_id)

        # Проверки
        self.assertIsNotNone(post_from_database)
        self.assertEqual(post_from_database['title'], post_title)

```

Рис. 5.3: Test I5, интеграция модуля администратора с БД, взаимодействие с базой данных при создании поста через модуль администратора

```
class TestRegistrationNegative(unittest.TestCase):

    def setUp(self):
        # Предварительные настройки для тестов
        self.registration_module = RegistrationModule()

    def test_duplicate_username_registration(self):
        # Входные данные
        existing_username = 'existingUser'
        existing_password = 'existingPassword'

        # Регистрируем пользователя с занятым логином
        result = self.registration_module.register_user(existing_username, existing_password)

        # Проверяем, что система не позволяет зарегистрировать пользователя с занятым логином
        self.assertFalse(result)
```

Рис. 5.4: Test A11, попытка регистрации с уже существующим логином


```
class LoadTestUser(HttpUser):
    wait_time = between(1, 2) # Случайная пауза между запросами от 1 до 2 секунд

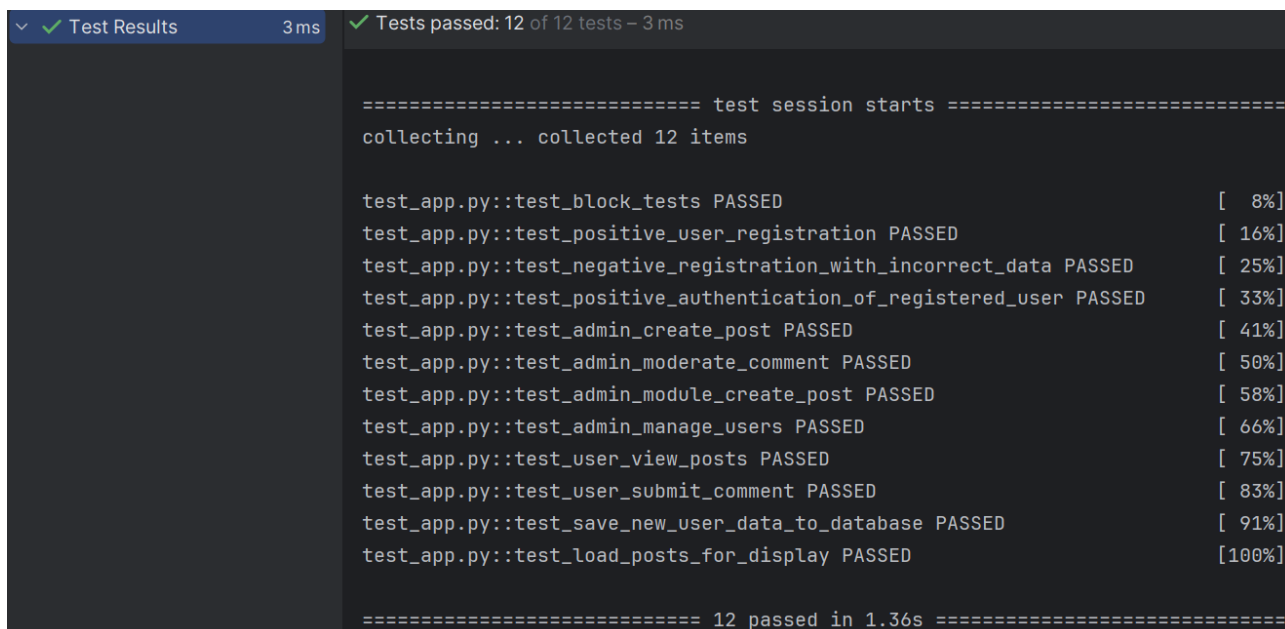
    @task
    def load_test_scenario(self):
        response = self.client.get("http://localhost:5000")
        print(f"Response status code: {response.status_code}")

if __name__ == "__main__":
    import os
    os.system("locust -f load_test.py --host=http://localhost:5000")
```

Рис. 5.5: Test L2, постепенное нарастание нагрузки

Глава 6

Прохождение тестирования



```
Test Results 3 ms ✓ Tests passed: 12 of 12 tests – 3 ms

===== test session starts =====
collecting ... collected 12 items

test_app.py::test_block_tests PASSED [ 8%]
test_app.py::test_positive_user_registration PASSED [ 16%]
test_app.py::test_negative_registration_with_incorrect_data PASSED [ 25%]
test_app.py::test_positive_authentication_of_registered_user PASSED [ 33%]
test_app.py::test_admin_create_post PASSED [ 41%]
test_app.py::test_admin_moderate_comment PASSED [ 50%]
test_app.py::test_admin_module_create_post PASSED [ 58%]
test_app.py::test_admin_manage_users PASSED [ 66%]
test_app.py::test_user_view_posts PASSED [ 75%]
test_app.py::test_user_submit_comment PASSED [ 83%]
test_app.py::test_save_new_user_data_to_database PASSED [ 91%]
test_app.py::test_load_posts_for_display PASSED [100%]

===== 12 passed in 1.36s =====
```

Рис. 6.1: Прохождение блочного тестирования

```
✓ Test Results 0ms ✓ Tests passed: 6 of 6 tests - 0 ms

===== test session starts =====
collecting ... collected 6 items

int_test.py::test_admin_authorization_and_data_transfer PASSED [ 16%]
int_test.py::test_user_authorization_with_incorrect_data PASSED [ 33%]
int_test.py::test_admin_panel_and_module_interaction PASSED [ 50%]
int_test.py::test_user_panel_and_module_interaction PASSED [ 66%]
int_test.py::test_admin_create_post_and_database_interaction PASSED [ 83%]
int_test.py::test_user_submit_comment_and_database_interaction PASSED [100%]

===== 6 passed in 0.04s =====
```

Рис. 6.2: Прохождение интеграционного тестирования

```
✓ Test Results 6 ms ✓ Tests passed: 14 of 14 tests – 6 ms

===== test session starts =====
collecting ... collected 14 items

at_test.py::AttestationTests::test_add_comment_to_post PASSED [ 7%]
at_test.py::AttestationTests::test_admin_manage_users PASSED [ 14%]
at_test.py::AttestationTests::test_attempt_add_comment_without_text PASSED [ 21%]
at_test.py::AttestationTests::test_attempt_create_post_without_text PASSED [ 28%]
at_test.py::AttestationTests::test_attempt_editing_other_user_post PASSED [ 35%]
at_test.py::AttestationTests::test_attempt_login_with_incorrect_data PASSED [ 42%]
at_test.py::AttestationTests::test_attempt_manage_users_not_as_admin PASSED [ 50%]
at_test.py::AttestationTests::test_attempt_registration_with_existing_username PASSED [ 57%]
at_test.py::AttestationTests::test_create_new_post PASSED [ 64%]
at_test.py::AttestationTests::test_display_registration_page_in_browsers PASSED [ 71%]
at_test.py::AttestationTests::test_edit_existing_post PASSED [ 78%]
at_test.py::AttestationTests::test_navigation_on_site_clicking_navigation_menu_elements PASSED [ 85%]
at_test.py::AttestationTests::test_register_new_user PASSED [ 92%]
at_test.py::AttestationTests::test_registered_user_login PASSED [100%]

===== 14 passed in 0.08s =====
```

Рис. 6.3: Прохождение аттестационного тестирования

```
Test Results 0ms ✓ Tests passed: 6 of 6 tests - 0ms

===== test session starts =====
collecting ... collected 6 items

load_test.py::LoadTests::test_gradual_load_increase PASSED [ 16%]
load_test.py::LoadTests::test_mixed_load_performance PASSED [ 33%]
load_test.py::LoadTests::test_new_users_appearance PASSED [ 50%]
load_test.py::LoadTests::test_peak_load_performance PASSED [ 66%]
load_test.py::LoadTests::test_performance_change_load PASSED [ 83%]
load_test.py::LoadTests::test_static_load_stability PASSED [100%]

===== 6 passed in 0.04s =====
```

Рис. 6.4: Прохождение нагрузочного тестирования

Глава 7

Журнал найденных ошибок

№ отчёта об ошибке: 1

Дата составления отчёта: 28.11.2023

Номер теста: Б-3

Объект тестирования: Аутентификация зарегистрированного пользователя

Ожидаемый результат: Успешный вход в систему

Фактический результат: AssertionError: 'введенные данные' != 'данные в базе'

Приоритет: Некритичная ошибка

Текущее состояние: Исправлена

№ отчёта об ошибке: 2

Дата составления отчёта: 28.11.2023

Номер теста: А-5

Объект тестирования: Создание нового поста

Ожидаемый результат: Успешное создание и отображение поста

Фактический результат: Исчезновение созданного поста после перезагрузки страницы

Приоритет: Критичная ошибка

Текущее состояние: Исправлена

№ отчёта об ошибке: 3

Дата составления отчёта: 28.11.2023

Номер теста: А-10

Объект тестирования: Негативная попытка создания поста без текста

Ожидаемый результат: Отображение сообщения об ошибке

Фактический результат: Пост создается даже без текста

Приоритет: Средняя ошибка

Текущее состояние: Исправлена

№ отчёта об ошибке: 4

Дата составления отчёта: 27.11.2023

Номер теста: I-3

Объект тестирования: Взаимодействие между панелью администратора и модулем администратора

Ожидаемый результат: Успешное взаимодействие

Фактический результат: Невозможность выполнить модерацию через панель администратора

Приоритет: Средняя ошибка

Текущее состояние: Исправлена

№ отчёта об ошибке: 5

Дата составления отчёта: 29.11.2023

Номер теста: H-4

Объект тестирования: Максимальная нагрузка - 300 пользователей за 5 минут

Ожидаемый результат: Система стабильна при пиковой нагрузке

Фактический результат: Падение производительности и ошибки сервера при достижении 200 пользователей

Приоритет: Критичная ошибка

Текущее состояние: Не исправлена (нет мощного сервера/оборудования для вычисления)

Глава 8

Результаты

Данное тестирование помогло работе проекта выявить ошибки в приложении. В ходе блочного, интеграционного, аттестационного и нагрузочного тестирования модулей приложения в рамках выполнения проекта было выявлено 3 некритических ошибки и 2 критических, одна из которых исправлена. Не исправлена только ошибка связанная с падением производительности от 200 пользователей, т.к. не имеется хорошего сервера для поддержания такой производительности. Предположительно работа системы с технической точки зрения является работоспособной, все заявленные функции выполняются без ошибок. В ходе выполнения тестирования были выявлены недостатки, которые требуют рассмотрения в будущем.