

Федеральное государственное автономное образовательное учреждение
высшего образования
«Омский государственный технический университет»

Кафедра «Автоматизированные системы обработки информации и
управления»

ОТЧЕТ

по лабораторной работе №2

на тему

«Создание плана тестирования»

Выполнил: ст. гр. ПИН-202

Горшенин Л.И.

Проверила: ст. преподаватель

Фатеева А.С.

Омск 2023

Введение

Назначение тестового плана

Данный тестовый план разработан для обеспечения качества функциональности классического CRUD приложения, а именно создание, удаление, редактирование и просмотр данных. Его целью является осуществление проверки на соответствие заявленным требованиям, выявление потенциальных дефектов и обеспечение стабильной работы бота.

Уровень предоставляемого плана

План охватывает функциональное тестирование данного приложения. Он не включает в себя тестирование производительности, безопасности или другие аспекты, отличные от функциональности.

Обзорный план относительно плана программного проекта

Данный тестовый план нацелен на проверку функциональности приложения и сверку с требованиями, изложенными в плане программного проекта. Он определяет процедуры тестирования, используемые методы, ресурсы и составляющие процесса тестирования.

1. Сценарий тестирования

Сценарий 1: Создание пользователя

Описание: Создание нового пользователя в приложении.

Шаги тестирования:

- Ввести имя, возраст и пол пользователя
- Отправить введенные данные при помощи кнопки «Присоединиться»
- Получить ответ от сервера
- Ожидаемый результат: В случае передачи корректных данных, данные о пользователе сохраняются в БД, сервер генерирует и отправляет клиенту. Иначе отправляет диалоговое окно о некорректно введенных данных.

Сценарий 2: Создание нового поста

Описание: Создание нового поста от имени пользователя.

Шаги тестирования:

- Выбрать пользователя
- Выбрать пользователя
- Ввести заголовок вопроса
- Отправить введенные данные при помощи кнопки «Задать вопрос»
- Получить ответ от сервера
- Ожидаемый результат: Данные о посте сохраняются в БД, сервер генерирует и отправляет клиенту. Появляется окно поста, в котором хранится история комментариев.

Сценарий 3: Написание комментария

Описание: Создание нового комментария в определенном посте.

Шаги тестирования:

- Выбрать пользователя и пост.
- Ввести текст комментария.
- Отправить введенные данные при помощи кнопки «Ответить».
- Получить ответ от сервера.
- Ожидаемый результат: Данные о комментарии сохраняются в БД, сервер генерирует и отправляет клиенту.

Сценарий 4: Редактирование комментария

Описание: Проверка возможности редактирования комментария.

Шаги тестирования:

- Создание и выбор комментария.
- Ввести новый текст комментария.
- Получить ответ от сервера.
- Ожидаемый результат: Измененные данные о комментарии сохраняются в БД, сервер генерирует и отправляет клиенту.

Сценарий 5: Удаления комментария

Описание: Проверка возможности удаления комментария.

Шаги тестирования:

- Выбор комментария.
- Нажать на кнопку «Удалить комментарий».
- Получить ответ от сервера.
- Ожидаемый результат: Данные об изменении сохраняются в БД, сервер генерирует и отправляет клиенту.

Сценарий 6: Переключение между постами

Описание: Проверка возможности переключения между разными постами и проверка корректного отображения представления.

Шаги тестирования:

- Создание нескольких постов.
- Нажать на кнопку «Следующий».
- Получить ответ от сервера.
- Ожидаемый результат: Сервер сгенерирует список данных указанного поста и отправит их клиенту, клиент в свою очередь отобразит данные.

2. Тестовые данные

Для проведения тестирования необходимы следующие тестовые данные:

Примеры задач разного типа (важные, срочные, обычные) с различными данными.

3. Тестовый скрипт

Для автоматизации тестирования созданы тестовые скрипты, использующие инструменты тестирования, такие как xUnit. Эти скрипты включают в себя автоматизированные шаги, которые выполняются для проведения тестов, включая создание, удаления, редактирования данных.

4. Набор тестов

Набор тестов включает в себя разнообразные сценарии, которые покрывают функциональность программы:

- Создание пользователя;
- Создание поста;
- Создание комментария;
- Редактирование комментария;
- Удаление комментария
- Переключение между постами.

5. Список идей тестов

В список идей для тестов включены возможные сценарии и тест-кейсы, которые могут потребоваться при дополнительном тестировании.

Тестирование на старых версиях: Убедитесь, что ваше ПО работает корректно на разных версиях Windows.

Тестирование на больших объемах данных: Убедитесь, что ваше ПО может обрабатывать большие объемы данных без сбоев или задержек.

Тестирование отказоустойчивости: Проведите тестирование для определения того, как ваше ПО ведет себя в случае сбоев серверов или других критических

ошибок.

6. Результаты тестирования

После выполнения каждого теста, результаты фиксируются в специальных отчетах. Эти отчеты содержат информацию о прохождении теста, включая успешное завершение или обнаруженные дефекты.

7. Дефекты

Если в процессе тестирования были обнаружены дефекты, они фиксируются в системе управления дефектами. Каждый дефект снабжается описанием, шагами воспроизведения и приоритетом для последующего устранения разработчиками.