**Test Plan Restful-booker**

# 1. Introduction

Этот тест-план описывает процесс тестирования RESTful-сервиса "restful-booker" с использованием инструмента Postman. "restful-booker" предоставляет API для управления бронированием номеров в отеле.

**2. In Scope**

- Регистрация новых пользователей.

- Получение списка доступных номеров.

- Бронирование номеров.

- Получение информации о бронировании.

- Обновление информации о бронировании.

- Удаление бронирования.

**3. Out of Scope**

- Аутентификация пользователей.

- Подтверждение оплаты бронирования.

- Управление пользователями (изменение паролей, удаление аккаунтов и т. д.).

**4. Quality Objective**

Цель тестирования - обеспечить высокое качество API "restful-booker", убедившись в правильной функциональности и отсутствии ошибок.

**5. Roles and Responsibilities**

- QA-инженер: Создание тест-плана, выполнение тестирования, отслеживание ошибок.

- Разработчики: Исправление выявленных ошибок.

**6. Test Methodology**

Тестирование будет проводиться вручную с использованием Postman. Для каждого эндпоинта будут написаны тесты с учетом позитивных и негативных сценариев.

**Test levels:**  
  
Unit Testing  
Проверка корректности парсинга и валидации данных запроса.  
Тестирование функций обработки данных (например, шифрование паролей, генерация токенов).  
  
Integration Testing  
Проверка правильности обработки запросов к базе данных для получения информации о бронированиях.  
Тестирование взаимодействия между компонентами, ответственными за авторизацию и доступ к ресурсам.

System Testing  
Проверка общей функциональности сервиса, включая его способность обрабатывать запросы и выдавать правильные ответы.  
Тестирование производительности сервиса при различных нагрузках.  
Проверка соответствия API документации и реальной функциональности.  
  
  
**Test Environment:**Windows 10 и выше

**7. Bug Triage**

1. Функциональное тестирование:

Критические ошибки (высокий приоритет):

- Ошибка при регистрации пользователя: Невозможность создать аккаунт. Пользователь не получает токен доступа после успешной регистрации.

- Ошибка при бронировании номера: Система не создает бронь в базе данных, что приводит к недоступности номеров.

Серьезные ошибки (средний приоритет):

- Ошибка при получении списка номеров: Система возвращает неполный список или неактуальные данные.

- Ошибка при обновлении информации о бронировании: Изменения не сохраняются корректно в базе данных.

Минорные ошибки (низкий приоритет):

- Ошибка при удалении бронирования: Система не удаляет бронь или удаляет неправильную бронь.

2. Тестирование производительности:

- Критические ошибки (высокий приоритет):

- Медленное выполнение запросов: Запросы на бронирование или получение информации о бронировании выполняются слишком медленно.

- Серьезные ошибки (средний приоритет):

- Низкая стойкость к высокой нагрузке: Система не выдерживает большой нагрузки при одновременном бронировании нескольких номеров.

- Минорные ошибки (низкий приоритет):

- Медленная загрузка при малой нагрузке: Загрузка страницы с информацией о номере занимает слишком много времени.

3. Тестирование контента:

- Критические ошибки (высокий приоритет):

- Отсутствие данных в ответе: Нет данных о бронировании в ответе при успешном запросе.

- Серьезные ошибки (средний приоритет):

- Неполная информация: Отображение неполной информации о номерах при запросе на получение списка доступных номеров.

- Минорные ошибки (низкий приоритет):

- Некорректные данные в ответе: Отображение некорректных данных в ответе на запрос о бронировании.

4. Визуальное тестирование:

- Критические ошибки (высокий приоритет):

- Некорректное отображение интерфейса: Отображение элементов пользовательского интерфейса с ошибками при различных разрешениях экрана.

- Серьезные ошибки (средний приоритет):

- Неправильное выравнивание: Неправильное выравнивание текста или элементов интерфейса.

- Минорные ошибки (низкий приоритет):

- Стилистические ошибки: Незначительные стилистические ошибки, не влияющие на функциональность.

5. Тестирование на краш:

- Критические ошибки (высокий приоритет):

- Краш при попытке доступа к несуществующему или поврежденному эндпоинту: Система не должна крашиться, предоставляя четкие сообщения об ошибке.

- Краш при попытке доступа к сервису во время недоступности API: Обработка отсутствия доступа к API.

- Серьезные ошибки (средний приоритет):

- Краш при передаче некорректных данных в запросе: Не должно происходить крашей даже при передаче неверных данных.

- Минорные ошибки (низкий приоритет):

- Изредка возникающие краши: Изредка возникающие краши, не влияющие на работоспособность сервиса в целом.

.

**8. Suspension Criteria and Resumption Requirements**

**Критерии приостановки:**

- Сбой критических функций :

- Если процесс регистрации пользователя не работает на Windows.

- Если получение информации о бронировании возвращает неправильные или неполные данные.

- Если обновление данных о бронировании не отражает изменений в системе.

- Если удаление бронирования не удаляет его из системы.

- Если любые критические функции, перечисленные выше, не работают должным образом, что приводит к значительному ухудшению пользовательского опыта.

**Требования для возобновления:**

Функциональная стабильность :

- Все функциональные возможности, указанные в рамках, должны работать должным образом без критических сбоев.

Пользовательский опыт :

- Пользовательский опыт должен оставаться постоянным на различных устройствах и для разных типов пользователей.

Производительность :

- Приложение должно поддерживать оптимальную производительность, с плавным взаимодействием и без значительных задержек.

Совместимость :

- Приложение должно быть совместимо с различными браузерами и устройствами, обеспечивая бесперебойную работу всех функций на различных платформах.  
  
 Безопасность и управление доступом :

- Должны быть предусмотрены правильные механизмы контроля доступа, чтобы обеспечить доступ только авторизованным пользователям к определенным функциям.

- Процессы аутентификации и авторизации должны быть надежными и безопасными.

Видимость данных :

- Пользователи должны иметь доступ только к данным, к которым они имеют право просмотра на основе своей роли и разрешений.

- Представление данных должно быть точным и последовательным на различных экранах и представлениях.

**9. Test Deliverables**

- Отчет о тестировании.  
- Отчет о найденных ошибках.  
- Чек-лист

**10. Resource & Environment Needs**

- Postman для выполнения тестов.

- Тестовое окружение "restful-booker", доступное для тестирования.

- https://pairwise.teremokgames.com/ для попарного тестирования

- Jira для заведения баг-репортов