

Негосударственное аккредитованное некоммерческое частное
образовательное учреждение высшего образования «Академия маркетинга и
социально-информационных технологий – ИМСИТ»
(г. Краснодар)

Академический колледж

Инженерно-информационное отделение

ДОКУМЕНТ ТЕСТИРОВАНИЯ
на разработку программного модуля
«База отдыха»

Разработчик: Таран И.А.

Заказчик: Учебное задание колледжа

Дата: 2025г.

Краснодар 2025

СОДЕРЖАНИЕ

1 Введение.....	3
2 Объект тестирования	3
3 Окружение тестирования	3
4 Типы тестирования	4
5 Критерии входа и выхода.....	4
6 План тестирования.....	4

1 Введение

Цель тестирования – проверить корректность работы веб-приложения базы отдыха в соответствии с техническим заданием.

Основные задачи:

- Проверка функциональных требований (регистрация, авторизация, просмотр номеров, бронирование, управление номерами администратором).
- Проверка нефункциональных требований (время отклика, производительность, безопасность).
- Обеспечение минимального покрытия кода $\geq 70\%$.

2 Объект тестирования

- web-приложение системы управления базой отдыха;
- сервер: Flask (Python);
- клиент: HTML/CSS (Jinja2);
- база данных: SQLite.

3 Окружение тестирования

- ОС: Windows 10 / Linux;
- Python 3.11;
- Flask 2.3.x;
- SQLite 3;
- браузеры: Chrome, Firefox;
- оборудование: стандартный ПК с 8 GB RAM;
- сеть: локальная / интернет (для проверки корректности работы в онлайн-режиме).

4 Типы тестирования

- unit-тестирование: проверка отдельных функций и модулей (обработчики Flask, модели БД);
- integration-тестирование: проверка взаимодействия модулей (бронирование + база данных + сервер);
- system-тестирование: проверка работы всей системы в целом;
- regression-testing: автоматизированный прогон тестов после каждого изменения кода;
- performance-testing: нагрузочные тесты (до 50 одновременных пользователей);
- security-testing: проверка аутентификации, авторизации, хэширования паролей и доступа к админ-панели.

5 Критерии входа и выхода

- Входные условия: система установлена и запущена, подготовлены тестовые данные, локальный сервер работает стабильно;
- Выходные условия: все критические функциональные тесты успешно пройдены, выявленные дефекты исправлены или задокументированы.

6 План тестирования

- составление набора тест-кейсов по всем функциям системы;
- ручное выполнение тестирования пользовательских сценариев;
- запуск автоматизированных unit- и integration-тестов;
- документирование фактических результатов и выявленных дефектов;
- оценка покрытия кода (statement/branch coverage);
- итоговый отчёт о тестировании.