***План тестирования***

**OpenCart Demo Testing**

**Product version v 1.0.0**

| **Предоставлено для:** | **Заказчик** |
| --- | --- |
| **Предоставил:** | **Компания …** |
| **Дата:** | **20 марта 2025 года** |
| **Версия:** | **V 1.0.0** |
| **Исполнитель:** | **Осипенко А.С.** |

### Оглавление

[**1. Введение 3**](#_obuwkxn3g7je)

[1.1 О чем этот документ? 3](#_svpbkwzezev9)

[1.2 Цель документа 4](#_6y2a3jgmuocz)

[1.3 Целевая аудитория 4](#_80rak0rnf5si)

[1.4 Ссылки на проектную документацию 4](#_xqzza3e8go2i)

[1.5 Обзор программного продукта 4](#_295qj1yej0p7)

[1.6 Цели тестирования 4](#_alp6ala17iqg)

[**2. Область тестирования 5**](#_vx0k65i8u6h7)

[2.1 Область применения 5](#_3g0n6wc01thy)

[2.2 Тестируемые модули: 5](#_e3vpxvnhqz5d)

[**3. Подходы к тестированию 5**](#_uo58d1lsyn0y)

[**4. Инструменты для тестирования 6**](#_4cre95w5nv58)

[**5. Среда тестирования 6**](#_sfkql579jt5e)

[5.1 Устройства и ОС: 6](#_eh0y69s1if2q)

[5.2 Браузеры: 6](#_fi5lq8a53q6k)

[**6. Тестовая документация 6**](#_443rf6khb3vq)

[6. 1 Чек-листы 6](#_53yan9u73lbq)

[6.2 Тест-кейсы 7](#_odejstapdrf1)

[6.3 Баг-репорты 7](#_vffl6jtqoihg)

[6.4 Тестовые отчеты 7](#_vbqaire1ol1f)

[**7. Критерии начала и окончания тестирования 8**](#_d4uhqc5x22zi)

[7.1 Критерии начала тестирования: 8](#_mifrw9lgn5kg)

[7.2 Критерии окончания тестирования: 8](#_5d5b61gfg7kk)

[**8. Возможные риски 8**](#_ppuaf2h4hqfc)

[**9. График тестирования 8**](#_gup16ooxj6hx)

[**10. Отчет о зарегистрированных дефектах 8**](#_1i1mf3u403jk)

### 1. Введение

#### 1.1 О чем этот документ?

Данный документ подробно описывает общий процесс тестирования, включая все его этапы — от подготовки и планирования до завершения.

**Этапы тестирования**

1. **Подготовка тестовой среды**

* Настройка оборудования и программного обеспечения
* Определение конфигурации тестовых устройств и браузеров

1. **Разработка тестовой документации**

* Создание чек-листов
* Написание тест-кейсов
* Формирование баг-репортов

1. **Выбор подходов для различных типов тестирования**

* Функциональное тестирование
* Тестирование производительности
* UI/UX тестирование

1. **Исполнение тестов и анализ результатов**

* Проведение тестирования в соответствии с тест-кейсами
* Фиксация результатов тестов
* Анализ выявленных дефектов

1. **Регистрация дефектов и их дальнейшее отслеживание**

* Заведение багов в систему учета дефектов
* Контроль исправления ошибок

**Ожидаемые результаты**

Определение критериев оценки тестирования и метрик успешности:

* Процент успешно пройденных тестов
* Общее количество найденных дефектов
* Количество критических и блокирующих багов

**Подходы к тестированию**

Описание используемых методов и стратегий:

* Исследовательское тестирование
* Разработка тест-кейсов и чек-листов
* Автоматизация тестирования (при необходимости)

**Применение метрик**

Количественные показатели, используемые для оценки качества тестирования:

* Общее количество тестов - общее число запланированных тест-кейсов/чек-листов
* Пройдено тестов - Количество успешно выполненных тестов
* Провалено тестов - Количество тестов, которые завершились с ошибкой
* Заблокировано - Тесты, которые не удалось выполнить из-за блокирующих факторов
* В ожидании - Тесты, которые еще не были запущены
* Процент успешного прохождения - процент пройденных тестов от общего количества
* Процент ошибок - Процент завершенных с ошибкой тестов от общего количества
* Количество заведенных багов - Общее количество найденных дефектов
* Критические баги - Баги, нарушающие ключевой функционал или блокирующие работу системы
* Высокий приоритет багов - Серьезные ошибки, влияющие на пользовательский опыт
* Средний приоритет багов - Ошибки, влияющие на функционал, но имеющие обходные пути
* Низкий приоритет багов - незначительные дефекты, косметические недочеты

**График тестирования**

Определение сроков и этапов тестирования с учетом различных его видов (функциональное, нагрузочное, интеграционное тестирование).

#### 1.2 Цель документа

* Формирование документа согласно стандартам
* Оценка рисков и способы их устранения
* Организация контроля процесса тестирования, определение методов регистрации дефектов, необходимого программного обеспечения, видов предоставления отчетов
* Определить график работ, описать используемую методологию и стратегию тестирования
* Определение критериев качества
* Определение ресурсов, необходимых для реализации последующих проектов
* Описание документации, необходимой для обеспечения тестирования по проекту

#### 1.3 Целевая аудитория

Данный документ предназначен для дальнейшего использования и служит для понимания общей картины тест-плана. Он создан для того, чтобы понять ключевые аспекты тестовых артефактов и провести тестирование в полном его объеме (От формирования мастер тест-плана до конечного тестового отчета)

#### 1.4 Ссылки на проектную документацию

План тестирования основан и имеет ссылки на следующую проектную документацию:

scheme\_Opencart.xmind (Схема вариантов сценариев);

Check-Lists and Test-Cases.xlsx (Временное хранилище для чек-листов и тест-кейсов)

TR-XXX.doc (Тестовые отчеты);

ХХХХ.doc (Руководство пользователей);

Техническое тестирование.doc (Сценарии технического тестирования);

Журнал тестирования.xls.

Сценарии тестирования.xls

#### 1.5 Обзор программного продукта

**URL для тестирования**:<https://demo-opencart.ru/index.php?route=common/home>

Данный сайт является демонстрационной версией интернет-магазина, реализованного на платформе OpenCart. Наша цель — провести полный цикл тестирования веб-приложения, чтобы выявить дефекты в функциональности, производительности, UI/UX и других аспектах сайта. В процессе тестирования будет осуществляться проверка работы всех основных модулей, взаимодействия между компонентами, а также проверка откликов API и взаимодействия с базой данных.

Результатом тестирования станет отчет с описанием найденных дефектов, рекомендациями по исправлению ошибок и итогами тестирования. Все дефекты будут регистрироваться в системе Jira для дальнейшего отслеживания.

#### 1.6 Цели тестирования

Оценка качества работы сайта на основе следующих аспектов:

* **Функциональное тестирование**: тестирование основных сценариев использования для выявления ошибок в логике и функциональности.
* **Производительность**: проверка работы сайта под условной нагрузкой, его откликов и стабильности при большом числе пользователей.
* **Удобство использования (UI/UX)**: оценка удобства взаимодействия пользователя с интерфейсами, а также анализ удобства навигации и восприятия.
* **Smoke тестирование**: предварительная проверка основных функций для удостоверения в работоспособности системы перед основным тестированием.
* **API тестирование (интеграционное тестирование)**: проверка взаимодействия между различными модулями сайта через API.
* **Исследовательское тестирование**: ручная проверка нестандартных сценариев и поиск возможных скрытых дефектов.

### 2. Область тестирования

#### 2.1 Область применения

Тестирование проводится на демонстрационной версии интернет-магазина на платформе OpenCart, расположенной по указанному URL. Все модули, указанные в разделе “Тестируемые модули”, будут подвергнуты различным типам тестирования, включая функциональное, производительное и API. Поскольку это демонстрационная версия, будет также осуществляться проверка стабильности сайта при изменениях в структуре и контенте.

#### 2.2 Тестируемые модули:

* **Модуль регистрации**: проверка правильности работы процесса регистрации пользователя на сайте.
* **Модуль авторизации**: тестирование корректности ввода данных пользователя и авторизации.
* **Каталог товаров и поиск товаров**: проверка функциональности фильтров, поиска, отображения товаров.
* **Личный кабинет пользователя**: проверка наличия функций управления данными пользователя, таких как редактирование профиля и просмотр истории заказов.
* **Сортировка товаров**: проверка работы различных сортировок товаров по цене, рейтингу, новизне и т. д.
* **Добавление товаров в “Закладки”**: тестирование возможности добавления товаров в список желаемого и его сохранение.
* **Работа со скидками**: проверка применения скидок, промокодов и их правильности.
* **Проверка карточек товаров**: проверка отображения информации о товаре, доступности, изображений и цен.
* **Оформление заказа**: тестирование процесса оформления заказа, проверки корзины и отправки данных на сервер

### **Оплата заказа**: тестирование взаимодействия с платежной системой. **Управление товарами**: проверка функционала добавления, удаления и редактирования товаров для администратора..

### 

### 3. Подходы к тестированию

* **Smoke тестирование**: Начальный этап тестирования для проверки основных функций, таких как регистрация, авторизация, добавление товаров в корзину.
* **Ручное тестирование**: Проверка сценариев, не поддающихся автоматизации, или сложных случаев, требующих креативного подхода (например, исследовательское тестирование).
* **Автоматизированное тестирование**: Использование автоматических скриптов (например, с использованием Selenium) для проверки повторяющихся тестов.
* **Тестирование производительности**: Использование инструментов, таких как JMeter, для оценки нагрузки на систему.
* **Интеграционное тестирование (API тестирование)**: Проверка корректности работы API, через которые осуществляется взаимодействие между модулями (например, создание заказа или изменение данных).

### 4. Инструменты для тестирования

* **Postman** — для тестирования REST API, включая запросы к серверу и проверку данных, передаваемых между фронтендом и бэкендом.
* **JMeter** — для тестирования производительности и нагрузки, имитации многократных запросов пользователей к сайту.
* **Selenium** — для автоматизированного тестирования функциональности сайта, например, проверка добавления товара в корзину, выполнения логина.
* **Jira** — для регистрации и отслеживания дефектов и багов.
* **Google Docs** — для ведения тестовой документации, отчетности, чек-листов, баг-репортов.

### 5. Среда тестирования

#### 5.1 Устройства и ОС:

* **MacOS 15.3.2** — для тестирования на настольной платформе.
* **IPadOS 17.7.5, IOS 18.3.2** — для тестирования мобильной версии сайта.

#### 5.2 Браузеры:

* **Google Chrome**
* **Safari**
* **Mozilla Firefox**

### 6. Тестовая документация

Тестовая документация нужна для структурирования процесса тестирования, обеспечения его прозрачности и повторяемости. Она помогает команде отслеживать статус проверок, анализировать результаты и эффективно исправлять дефекты. Кроме того, документация снижает зависимость от конкретных специалистов, позволяя новым сотрудникам быстро вникать в процесс. Данный тестовый проект будет включать следующие артефакты:

#### **6. 1 Чек-листы**

Для проведения тестирования будут использоваться чек-листы, выполняемые вручную. Они позволят структурировать процесс тестирования, обеспечивая контроль за ключевыми аспектами работы системы. Каждый чек-лист будет содержать информацию о тестируемом проекте, цели проверки, среде тестирования и исполнителе.

**Будут использоваться следующие чек-листы:**

1. Чек-лист исследовательского тестирования. Он включает тестирование функционала и удобства использования. Позволяет выявить ошибки в функционале и оценить удобство работы с системой.
2. Чек-лист тестирование функциональности каждого участка сайта
3. Чек-лист тестирования API. Позволяет проверить корректность работы API, соответствие ответов спецификациям и отработку ошибок.
4. Чек-лист тестирование производительности. Направлен на проверку скорости работы системы, стабильности под нагрузкой и потребления ресурсов.

**Структура чек-листа**

* **Название** - название продукта
* **Цель проверки** - описание текущего тестирования
* **ID** - Идентификатор чек-листа
* **Дата проведения**
* **Исполнитель** - ответственный за проверку
* **Среда тестирования -** устройства, версия ПО, описание браузеров и их версии.

**Разделы чек-листа**

* **Модуль** - часть системы, которая будет тестироваться
* **Требование** - ссылка на требование, которое проверяется
* **Название проверки** - краткое описание теста
* **Статус** - Passed / Failed
* **Комментарий** - необязательное поле для дополнительных замечаний

Этот формат обеспечит удобное проведение тестирования и фиксацию его результатов.

Все чек-листы будут записаны в формате Excel.

#### 6.2 Тест-кейсы

Документ с подробным описанием шагов для выполнения каждого теста, ожидаемого результата и критериев успеха.  
Тест-кейсы будут проводиться на важный функционал, а именно такие модули как: регистрация - авторизация - добавление в корзину - оформление заказа - отслеживание заказа, без авторизации отслеживание заказа, оплата заказа и товар начал доставку, возврат заказа и тд

Каждый Тест-кейс будет включать следующие параметры:

* **Тест-кейс ID** - Идентификатор Тест-кейса
* **Требование** - ссылка на требование, которое проверяется
* **Тест приоритет**
* **Среда тестирования -** устройства, версия ПО, описание браузеров и их версии.
* **Модуль** - часть системы, которая будет тестироваться
* **Название проверки** - краткое описание теста
* **Предусловие - условия, которые должны быть выполнены перед воспроизведением**
* **Шаги воспроизведения** - четкая последовательность действий
* **Ожидаемый результат** - как система должна работать согласно требованиям.
* **Фактический результат -**  что происходит на самом деле при выполнении шагов

#### 6.3 Баг-репорты

Это документ, описывающий ошибку, найденную во время тестирования. Он должен быть четким, понятным и содержать всю необходимую информацию для разработчиков

**Структура баг-репорта:**

* **ID** -Идентификатор Бага
* **Заголовок** - краткое и точное описание проблемы
* **Среда тестирования -** устройства, версия ПО, описание браузеров и их версии.
* **Критичность** - Насколько критична ошибка
* **Предусловие** - условия, которые должны быть выполнены перед воспроизведением бага
* **Шаги воспроизведения** - четкая последовательность действий для вызова ошибки.
* **Ожидаемый результат** - как система должна работать согласно требованиям.
* **Фактический результат** - что происходит на самом деле при выполнении шагов
* **Вложения** - скриншоты, видео, логи или любые другие доказательства бага

#### 6.4 Тестовые отчеты

Это документ, который содержит информацию о результатах тестирования. Он нужен для подведения итогов тестирования, оценки качества продукта и принятия решений о его готовности к релизу.

**Структура тестового отчета**

* **Название -** Идентификатор Тестового отчета (TR-XXX)
* **Проект -** Название проекта
* **Дата** - Дата проведения тестирования
* **Тестировщик -** Имя кто создал отчет о тестировании
* **Инструменты тестирования -** Используемые инструменты для тестирования
* **Среда тестирования -** устройства, версия ПО, описание браузеров и их версии.
* **Цель тестирования -** Краткое описание, что проверяется
* **Риски -** Возможные проблемы, влияющие на тестирование
* **Методология тестирования -** Используемый тип тестирования, техники тест-дизайна, подходы к тестированию
* **Пройденные тесты и результаты -** Краткое описание проведенных тестов
* **Метрики**
* **Заключение** - подведение итогов, рекомендации

### 7. Критерии начала и окончания тестирования

#### 7.1 Критерии начала тестирования:

* Доступность среды для тестирования.
* Подготовка и утверждение тест-кейсов и чек-листов.
* Определение участников тестирования и распределение ролей.

#### 7.2 Критерии окончания тестирования:

* Все критические баги устранены или зарегистрированы с фиксированным сроком исправления.
* Пройдено Smoke тестирование.
* Завершено регрессионное тестирование после исправлений.
* Подготовлены и утверждены тестовые отчеты и баг-репорты.

### 8. Возможные риски

* **Недоступность тестируемой среды**: Ошибки или проблемы с сервером, которые могут привести к невозможности тестирования.
* **Ограничения в доступе к административной панели**: Недостаточный доступ для выполнения проверок в административной части сайта.
* **Возможные изменения в OpenCart Demo**: Изменения в исходном коде платформы, которые могут повлиять на стабильность и функциональность сайта.

### 9. График тестирования

График тестирования будет зависеть от размера команды и выделенного времени. Например:

* Smoke тестирование: 1 день
* Функциональное тестирование: 5 дней
* Производительность (JMeter): 2 дня
* Интеграционное тестирование (API): 3 дня
* Регрессионное тестирование: 2 дня

### 10. Отчет о зарегистрированных дефектах

Каждый найденный дефект будет зарегистрирован в **Jira** с указанием:

* Статуса дефекта (например, новый, в процессе, исправлен)
* Приоритета дефекта (низкий, средний, высокий)
* Шагов воспроизведения
* Ожидаемого и фактического результатов
* Примечаний и рекомендаций

Также будет составлен отдельный .doc отчет с итогами тестирования.