МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования «Самарский национальный исследовательский университет имени академика С.П. Королева»

(Самарский университет)

Институт информатики и кибернетики

Кафедра информационных систем и технологий

Отчет по дисциплине

«Обеспечение качества программного обеспечения»

на тему «Тест-план»

Выполнил:

Иваев Д.Ш,   
Поздеев С.А.,  
Сластунов Я.Г.,   
Немова Н. А.

Проверил:

к.т.н., доцент Тимофеев А.В.

Самара 2024

**SustainBuy**

**Тест-план**

**История изменений**

|  |  |  |  |
| --- | --- | --- | --- |
| Дата | Версия | Автор | Описание |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Содержание**

[1. Тезаурус 4](#_Toc179146783)

[2. Введение 6](#_Toc179146784)

[3. Состав работ 7](#_Toc179146785)

[4. Критерии качества 9](#_Toc179146786)

[5. Факторы успеха 10](#_Toc179146787)

[6. Оценка рисков 13](#_Toc179146788)

[7. Ресурсы 17](#_Toc179146789)

[8. Тестовая документация и поставка 19](#_Toc179146790)

[9. Календарный план 22](#_Toc179146791)

[10. Схема сетевого окружения 23](#_Toc179146792)

# Тезаурус

**SustainBuy** — это маркетплейс, предлагающий экологичные и биоразлагаемые товары, включая функционал для расчета углеродного следа и рекомендации по его компенсации через участие в эко-проектах.

**Тестирование** — это процесс проверки функционала, добавленного в рамках текущей итерации разработки, для обеспечения его корректной работы, соответствия требованиям, отсутствия багов и предоставления ожидаемого пользовательского опыта.

**Команда разработки** — это группа специалистов, проводящая тестирование, обеспечивающая качество продукта на стадии итерации и предоставляющая результаты для принятия решений о готовности выпуска новой версии.

**Пользователи продукта** — это покупатели, заинтересованные в экологичных товарах и осознанном потреблении, стремящиеся минимизировать негативное воздействие на окружающую среду и компенсировать углеродный след.

**Критические функции** — это важнейшие элементы системы, такие как расчет углеродного следа и рекомендации по компенсации, которые требуют основного внимания при тестировании.

**Объем тестирования** — это минимальный набор тестов для критических функций, автоматизация повторяющихся тестов и smoke-тесты для основных сценариев пользователя.

**Время тестирования** — это длительность тестирования (3-4 недели), приоритизация тестов и быстрая проверка багфиксов для оптимизации процесса.

**Бюджет тестирования** — это ограничение затрат на тестирование с использованием open-source инструментов и обеспечение масштабируемости тестирования.

**Факторы успеха тестирования** — это наличие четких требований, доступ к тестовой среде и данным, квалифицированные тестировщики и заморозка требований.

**Ключевые ресурсы** — это команда тестирования, оборудование, тестовые инструменты и доступ к тестовой среде, необходимые для успешного проведения тестирования.

**QA Lead** — это руководитель тестирования, координирующий все процессы, планирование и отчетность.

**Тестировщики** — это специалисты, отвечающие за создание и выполнение тестов.

**Automation QA** — это специалисты по автоматизации тестирования, разрабатывающие скрипты для автоматизированного тестирования.

**DevOps инженер** — это специалист, ответственный за настройку тестовой среды и CI/CD процессов.

**Selenium** — это инструмент для автоматизации UI тестов веб-приложения.

**Jest / Mocha** — это инструменты для unit-тестирования функциональных модулей.

**Postman** — это инструмент для тестирования API.

**JIRA / Trello** — это системы для управления задачами тестирования и отслеживания багов.

**GitLab CI/CD или Jenkins** — это инструменты для автоматизации выполнения тестов в процессе непрерывной интеграции.

**JMeter** — это инструмент для тестирования производительности и нагрузки.

**TestRail** — это система для управления тест-кейсами и отчетности по результатам тестирования.

# Введение

* Какова функциональность продукта?

SustainBuy — это маркетплейс, предлагающий экологичные и биоразлагаемые товары. Он включает функционал для расчета углеродного следа от покупок и предоставляет рекомендации по его компенсации через участие в эко-проектах.

* Что мы будем делать в рамках тестирования?

Мы будем тестировать функционал, добавленный в рамках текущей итерации разработки, длительностью 3-4 недели., расчет углеродного следа, отображение товаров, рекомендации по компенсации и оформление заказов.

* Зачем?

Цель тестирования — убедиться, что разработанный функционал работает корректно, соответствует требованиям, не имеет багов и предоставляет пользователям ожидаемый опыт. Это также важно для проверки, что новый функционал не нарушает уже существующие функции продукта.

* Для кого?

Для этих стейкхолдеров тестирование важно с точки зрения соблюдения сроков, бюджета и качества поставляемого продукта. Оно позволяет убедиться, что все ключевые требования и цели проекта выполнены.

* Кто будет пользоваться продуктом? Конечные потребители - пользователи.

# Состав работ

1. Компоненты и функции, которые НАДО тестировать:

* Модуль расчета углеродного следа: Проверка корректности расчетов углеродного следа на основе введённых данных о покупках.
* Функция рекомендаций по компенсации: Проверка алгоритма рекомендаций, предлагаемых пользователю для компенсации углеродного следа через эко-проекты.
* Интеграция расчета углеродного следа с корзиной покупок: Убедиться, что расчет углеродного следа корректно связан с данными о покупках в корзине.
* Интерфейс для отображения результатов: Тестирование пользовательского интерфейса для отображения углеродного следа и рекомендаций.
* Функция выбора и подтверждения участия в эко-проектах: Проверка возможности выбора предложенных эко-проектов и их интеграции с корзиной покупок или профилем пользователя.

2. Компоненты и функции, которые НЕ НАДО тестировать:

* Каталог товаров: Если в текущей итерации не вносились изменения в отображение и структуру каталога, его тестирование не является приоритетом.
* Система оплаты: Тестирование платежной системы не требуется, если она не была изменена в рамках этой итерации.
* Авторизация и регистрация: Если текущая итерация не предполагает доработок или изменений в механизмах входа и регистрации, их тестирование можно исключить.

3. Сторонние компоненты:

* API для эко-проектов: Используемые API сторонних эко-проектов для получения данных о возможностях компенсации углеродного следа.
* Сторонние библиотеки или сервисы для расчетов углеродного следа: Если используется внешняя библиотека для расчета, важно учитывать её стабильность.
* Интеграция с платежными системами: Сторонние системы для оплаты участия в эко-проектах, если требуется оформление платежа.

# Критерии качества

* Проверить, что расчет углеродного следа и генерация рекомендаций выполняются в приемлемое время, без значительных задержек при высокой нагрузке (например, при одновременном использовании функции большим количеством пользователей).
* Обеспечить, что функционал корректно работает на различных браузерах и устройствах (например, Chrome, Firefox, Safari, мобильные устройства).
* Проверить, что при высоких нагрузках или возникновении непредвиденных ситуаций (например, отказ в доступе к экосистемным сервисам) система сохраняет свою стабильность и не падает.
* При ограниченном бюджете можно использовать автоматизированные тесты для ключевых функциональностей (например, расчет углеродного следа и оформление заказов), чтобы снизить затраты на ручное тестирование в будущем.
* Убедиться, что пользователи могут корректно оформлять и завершать покупки, не сталкиваясь с блокирующими ошибками. Функционал оформления должен работать гладко и без сбоев.

# Факторы успеха

1. **Четкие и стабильные требования:**
   * **Неизменность требований**: Один из ключевых факторов успешного тестирования — это стабильные, не меняющиеся требования. Когда требования заморожены на определённой стадии разработки, это позволяет тестировщикам точно планировать сценарии и процесс тестирования. Изменения в требованиях на поздних этапах могут привести к необходимости пересмотра тестов, что может создать путаницу и увеличить риски ошибок.
2. **Отсутствие задержек в разработке:**
   * Задержки в разработке негативно влияют на тестирование, так как сжатые сроки в конечной фазе проекта могут привести к поверхностному тестированию или к его ускорению за счет качества. Чтобы минимизировать риски, важно, чтобы команды разработчиков и тестировщиков работали синхронно и задачи передавались вовремя.
3. **Климат в коллективе**: Позитивная атмосфера в команде способствует эффективному сотрудничеству, что в свою очередь позволяет быстрее выявлять и устранять ошибки. Открытое и честное общение между участниками команды также способствует снижению стресса и улучшению общего настроя.
4. **Лицензионное программное обеспечение**: Использование лицензионного ПО гарантирует доступ к последним обновлениям, технической поддержке и дополнительным возможностям, что способствует более качественному тестированию. Легальность использования инструментов помогает избежать правовых проблем и возможных перебоев в работе.
5. **Ранняя и постоянная вовлеченность тестировщиков:**
   * Чтобы правильно работать, тестировщики должны быть вовлечены на самых ранних этапах разработки, желательно начиная с этапа проектирования. Это позволяет понять специфику системы и разработать тесты на основе чётких критериев. Хорошее понимание системы, требований и её архитектуры позволит эффективно разрабатывать тестовые кейсы и снизить вероятность пропуска важных тестов.
6. **Надёжные инструменты и среда тестирования:**
   * Для выполнения успешного тестирования необходимо обеспечить качественную среду, в которой тестировщики смогут работать без перебоев. Это включает в себя как стабильные тестовые окружения (реплики продакшена), так и надёжные инструменты для автоматизации тестирования (если речь идёт об автоматизированных тестах). Эти элементы обеспечат точность тестов и отсутствие влияния внешних факторов, не связанных с системой.
7. **Адекватное время на тестирование:**
   * Тестировщики должны иметь достаточное время для проведения всех запланированных тестов. Отсутствие изменений в последний момент позволяет следовать плану тестирования, не жертвуя его качеством и полнотой. Если изменения всё же необходимы, их нужно внедрять с пониманием того, как они повлияют на график тестирования и общее качество продукта.
8. **Коммуникация и координация между командами:**
   * Оперативное взаимодействие между командами разработки, тестирования и управления проектом обеспечивает устранение любых задержек или недоразумений. Это особенно важно, когда нужно быстро реагировать на выявленные ошибки или внедрять изменения.
9. **Автоматизация и тестовое покрытие:**
   * Для успешного выполнения тестирования на большом проекте, как SustainBuy, необходимо применять автоматизацию для повторяющихся тестов (например, расчёт углеродного следа или рекомендации для пользователей), чтобы обеспечить высокое покрытие тестами без чрезмерной траты ресурсов.

# Оценка рисков

### 1. Неясные или неполные требования

**Описание риска:** Требования к функционалу приложения могут быть нечеткими или неполными, что затруднит разработку тестов.

**Тип:** Операционный

**Категория:** Требования

**Потенциальное воздействие:** Высокая вероятность обнаружения критических ошибок на этапе тестирования, что приведет к задержкам.

**Вероятность наступления (1-5):** 2

**Влияние (1-5):** 5

**Уровень:** 15 (высокий)

**Стратегия реагирования:** Проведение регулярных встреч с командой разработки для уточнения требований и их актуализации.

**План управления:** Включить регулярные проверки требований и добавление их в тест-план.

### 2. Задержки в разработке

**Описание риска:** Неожиданные задержки в разработке могут привести к сжатию времени на тестирование.

**Тип:** Операционный

**Категория:** Процесс

**Потенциальное воздействие:** Недостаточное время на тестирование, что увеличит вероятность пропуска критических ошибок.

**Вероятность наступления (1-5):** 1

**Влияние (1-5):** 5

**Уровень:** 8 (средний)

**Стратегия реагирования:** Подключать доп. сотрудников или вводить сверхурочные, чтобы разгрести долги

**План управления:** Создание резервного времени для тестирования и регулярный мониторинг прогресса разработки.

### 3. Изменения в требованиях на поздних этапах

**Описание риска:** Неожиданные изменения требований могут возникнуть после завершения тестирования.

**Тип:** Операционный

**Категория:** Требования

**Потенциальное воздействие:** Необходимость пересмотра тестов и повторного тестирования, что может привести к сжатию сроков.

**Вероятность наступления (1-5):** 1

**Влияние (1-5):** 3

**Уровень:** 9 (средний)

**Стратегия реагирования:** Установление заморозки требований на этапе тестирования.

**План управления:** Включить процесс управления изменениями в тест-план.

### 4. Проблемы с тестовой средой

**Описание риска:** Тестовая среда может оказаться нестабильной или неадекватной для проведения тестирования.

**Тип:** Операционный

**Категория:** Технология

**Потенциальное воздействие:** Сложности в воспроизведении ошибок и недостоверные результаты тестирования.

**Вероятность наступления (1-5):** 1

**Влияние (1-5):** 4

**Уровень:** 8 (средний)

**Стратегия реагирования:** Регулярное тестирование и обновление тестовой среды. Наличие альтернативных платформ для тестирования.

**План управления:** Включить проверки и тесты среды в общий план тестирования.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Описание риска** | **Тип** | **Категория** | **Потенциальное воздействие** | **Вероятность (1-5)** | **Влияние (1-5)** | **Уровень** | **Стратегия реагирования** | **План управления** |
| 1 | Неясные или неполные требования | Операционный | Требования | Высокая вероятность обнаружения критических ошибок на этапе тестирования, что приведет к задержкам. | 2 | 5 | 15 | Проведение регулярных встреч с командой разработки для уточнения требований и их актуализации. | Включить регулярные проверки требований и добавление их в тест-план. |
| 2 | Задержки в разработке | Операционный | Процесс | Недостаточное время на тестирование, что увеличит вероятность пропуска критических ошибок. | 1 | 5 | 8 | Подключать доп. сотрудников, чтобы разгрести долги, или вводить сверхурочные | Создание резервного времени для тестирования и регулярный мониторинг прогресса разработки. |
| 3 | Изменения в требованиях на поздних этапах | Операционный | Требования | Необходимость пересмотра тестов и повторного тестирования, что может привести к сжатию сроков. | 1 | 3 | 9 | Установление заморозки требований на этапе тестирования. | Включить процесс управления изменениями в тест-план. |
| 4 | Проблемы с тестовой средой | Операционный | Технология | Сложности в воспроизведении ошибок и недостоверные результаты тестирования. | 1 | 4 | 8 | Регулярное тестирование и обновление тестовой среды. Наличие альтернативных платформ для тестирования. | Включить проверки и тесты среды в общий план тестирования. |

# Ресурсы

**Техническое задание**

- **Команда тестирования**: Специалисты с опытом тестирования веб-приложений и экологических данных, способные разрабатывать и выполнять тест-кейсы для расчета углеродного следа и рекомендаций.

- **Оборудование**: Достаточная вычислительная мощность для выполнения тестов на различных устройствах и платформах.

- **Тестовые инструменты**: Наличие автоматизированных тестовых средств и систем для ручного тестирования.

- **Доступ к тестовой среде:** Полноценная среда, которая моделирует реальные условия работы приложения с корректными данными.

1. **Команда тестирования**

- QA Lead: Руководитель тестирования, координирующий все процессы тестирования, планирование, мониторинг и отчетность.

- Тестировщики: Специалисты по ручному и автоматизированному тестированию, отвечающие за создание и выполнение тестов.

- Automation QA: Специалисты по автоматизации тестирования, разрабатывающие скрипты для автоматизированного тестирования функциональности.

- DevOps инженер: Ответственный за настройку и поддержку тестовой среды, CI/CD процессов и интеграцию тестов в общую систему разработки.

2. **Оборудование**

- Серверы для тестовой среды: Выделенные серверы или облачные ресурсы для выполнения тестирования функционала, в том числе тестов производительности.

- Рабочие станции тестировщиков: Машины, оснащенные необходимым ПО для выполнения тестов (как ручных, так и автоматизированных).

- Мобильные устройства и браузеры: Набор устройств (мобильных и настольных) для тестирования кросс-платформенной функциональности, в том числе различные браузеры (Chrome, Firefox, Safari).

3. **Тестовые инструменты** (ПО)

- Selenium: Для автоматизации UI тестов веб-приложения.

- Jest / Mocha: Для unit-тестирования функциональных модулей, включая расчеты углеродного следа.

- Postman: Для тестирования API, особенно расчета углеродного следа и получения рекомендаций по эко-проектам.

- JIRA / Trello: Для управления задачами тестирования, отслеживания багов и координации команды.

- GitLab CI/CD или Jenkins: Для автоматизации выполнения тестов в процессе непрерывной интеграции и доставки.

- JMeter: Для тестирования производительности и нагрузки (например, проверка расчетов углеродного следа под высокой нагрузкой).

- Allure или TestRail: Для управления тест-кейсами и отчетности по результатам тестирования.

# Тестовая документация и поставка

### Тест-план

**Основные элементы:**

* **Цели:** Определение целей тестирования.
* **Объем:** Описание тестируемых компонентов и функций.
* **Задачи:** Установление задач для команды тестирования.
* **Критерии успеха:** Условия, определяющие успешность тестирования.
* **Риски:** Идентификация возможных рисков.
* **Ресурсы:** Назначение необходимых ресурсов.
* **Подход:** Общая стратегия тестирования для итерации.

### Тестовые сценарии

**Описание:** Подробные шаги для выполнения тестов, включая:

* Условия
* Ожидаемые результаты
* Критерии успеха

Сценарии могут быть как для ручного, так и для автоматизированного тестирования.

### Программа и методика испытаний (ПМИ)

**Содержание:**

* Методологии тестирования
* Порядок выполнения тестов
* Список тестов и компонентов
* Процедуры проверки функциональности, производительности и безопасности

### Отчет о проведении тестирования

**Содержит:**

* Результаты тестов
* Статистику выполненных тестов
* Процент покрытия
* Выявленные дефекты
* Статус выполнения тестов

### Отчеты о дефектах

**Содержит:** Записи о найденных дефектах с описанием, шагами для воспроизведения и текущим статусом (открыт, исправлен, закрыт).

### Поставка

**Ответственные:**

* **QA Lead:** Координация тестирования и поставка отчетов.
* **Тестировщики:** Создание и выполнение тест-сценариев.
* **Automation QA:** Разработка автоматизированных тестов и отчетов.
* **DevOps:** Поддержка автоматизации и управление тестовой средой.

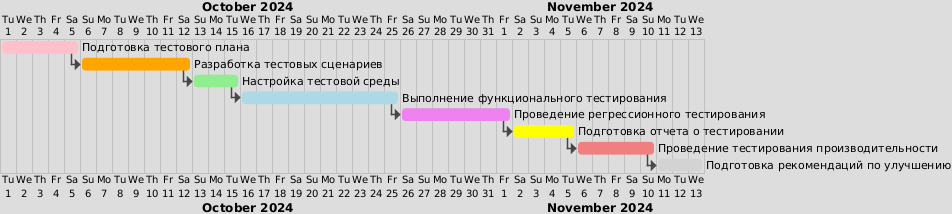
**Частота поставки:**

* **Тест-план:** Перед началом итерации.
* **Тестовые сценарии:** До тестирования с обновлениями по мере необходимости.
* **ПМИ:** Перед тестированием.
* **Отчет о тестировании:** По завершении каждой итерации (раз в 3-4 недели).
* **Отчеты о дефектах:** По мере выявления дефектов.

### Процесс

1. **Создание документации:** QA Lead разрабатывает тест-план и ПМИ; тестировщики готовят сценарии.
2. **Проведение тестирования:** Выполнение тестов с отслеживанием дефектов.
3. **Сбор отчетов:** Итоговые отчеты о тестировании и дефектах с анализом результатов.
4. **Частота поставки:** Документы обновляются в начале каждой итерации, отчеты предоставляются по завершении тестов.

# Календарный план



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Описание задачи** | **Длительность (дней)** | **Дата начала** | **Дата окончания** | **Назначенный сотрудник** |
| Подготовка тестового плана | 5 | 01.10.2024 | 05.10.2024 | QA Lead |
| Разработка тестовых сценариев | 7 | 06.10.2024 | 12.10.2024 | QA Engineer |
| Настройка тестовой среды | 3 | 13.10.2024 | 15.10.2024 | DevOps Engineer |
| Выполнение функционального тестирования | 10 | 16.10.2024 | 25.10.2024 | QA Engineer |
| Проведение регрессионного тестирования | 7 | 26.10.2024 | 02.11.2024 | QA Engineer |
| Подготовка отчета о тестировании | 4 | 03.11.2024 | 06.11.2024 | QA Lead |
| Проведение тестирования производительности | 5 | 07.11.2024 | 11.11.2024 | Performance Tester |
| Подготовка рекомендаций по улучшению | 3 | 12.11.2024 | 14.11.2024 | QA Lead |

# Схема сетевого окружения

