

Федеральное государственное автономное образовательное учреждение высшего
образования Национальный исследовательский университет ИТМО

Факультет программной инженерии и компьютерной техники

Основы программной инженерии

Лабораторная работа №1

Вариант № 535787

Выполнил: студент группы Р3208,
Васильев Н. А.

Преподаватель: Воронина Д. С.

Санкт-Петербург 2025

Текст задания

Вариант №535787: aliexpress.ru - <https://aliexpress.ru/>

Составить список требований, предъявляемых к разрабатываемому веб-сайту (в соответствии с вариантом). Требования должны делиться на следующие категории:

- Функциональные.
 - Требования пользователей сайта.
 - Требования владельцев сайта.
- Нефункциональные.

Требования необходимо оформить в соответствии с шаблонами RUP (документ SRS - Software Requirements Specification). Для каждого из требований нужно указать его атрибуты (в соответствии с методологией RUP), а также оценить и аргументировать приблизительное количество часов, требующихся на реализацию этого требования.

Для функциональных требований нужно составить UML UseCase-диаграммы, описывающие реализующие их прецеденты использования.

Software Requirements Specification

1. Introduction (Введение)

1.1 Purpose

Этот документ определяет требования к разработке веб-сайта AliExpress Russia, включая функциональные и нефункциональные требования. Он служит основой для проектирования, разработки, тестирования и развертывания системы.

1.2 Scope (Область применения)

Данный документ создан для разработки сайта <https://aliexpress.ru/>. AliExpress Russia – это крупная онлайн-платформа электронной коммерции, предоставляющая пользователям возможность покупать товары у различных продавцов.

Этот документ послужит базой для разработки, тестирования и реализации функциональности системы согласно установленным требованиям. Также он будет использоваться в качестве справочника для технической поддержки и будущего расширения сервиса.

1.3 Definitions, Acronyms and Abbreviations (Определения и аббревиатуры)

AliExpress Russia – Российская версия международной платформы электронной коммерции AliExpress, адаптированная для пользователей из России.

Sass (Syntactically Awesome Style Sheets) – Препроцессор CSS, добавляющий функции вроде переменных, вложенных правил и миксинов.

CSS-in-JS – это метод стилизации, с помощью которого JavaScript используется для стилизации компонентов. Когда этот JavaScript анализируется, CSS генерируется и прикрепляется к DOM.

Header – это верхняя часть веб-страницы, содержащая основные элементы навигации и идентификации сайта.

Footer – это нижняя часть веб-страницы, содержащая вспомогательную информацию и ссылки.

JavaScript – Язык программирования для создания интерактивных веб-страниц и приложений.

TypeScript – Надмножество JavaScript с добавлением статической типизации.

React – JavaScript-библиотека с открытым исходным кодом для разработки пользовательских интерфейсов.

API – Программный интерфейс приложений.

TLS (Transport Layer Security) – Протокол безопасности для защиты данных пользователей.

CDN (Content Delivery Network) – Сеть доставки контента для ускорения загрузки страниц.

Webpack – Инструмент для сборки и оптимизации JavaScript-приложений.

Babel – Транспайлер для преобразования современного JavaScript в код, совместимый со старыми браузерами.

ESLint – Инструмент для статического анализа JavaScript/TypeScript кода.

Java – Высокоуровневый объектно-ориентированный язык программирования.

Spring Boot – Фреймворк для создания приложений на Java с минимальной конфигурацией.

JUnit – Фреймворк для модульного тестирования Java-приложений.

PostgreSQL – Реляционная база данных с открытым исходным кодом.

Redis – Хранилище данных типа «ключ-значение» с поддержкой кэширования.

GitHub – Платформа для хостинга и совместной разработки Git-репозитория.

CI/CD (Continuous Integration/Continuous Delivery) – Практика автоматизации интеграции кода и его доставки в production.

GitHub Actions – Платформа, которая позволяет автоматизировать рабочие процессы прямо в репозитории.

Docker – Программное обеспечение для автоматизации развёртывания и управления приложениями в средах с поддержкой контейнеризации, контейнеризатор приложений.

Code Review – Систематическая проверка исходного кода программы с целью обнаружения и исправления ошибок, которые остались незамеченными в начальной фазе разработки.

Мерж – В системе контроля версий слияние — это фундаментальная операция, которая согласовывает изменения, внесенные в коллекцию файлов с контролем версий. Чаще всего это необходимо, когда файл модифицируется в двух независимых ветках и впоследствии объединяется.

Пуш – Это одна из консольных команд Git. Она позволяет передать изменения из локального репозитория.

Jenkins – Инструмент для автоматизации CI/CD-процессов.

Agile – Гибкая методология разработки, ориентированная на итеративное выполнение задач.

WCAG 2.1 (Web Content Accessibility Guidelines) – Рекомендации по обеспечению доступности веб-контента.

MTBF (Mean Time Between Failures) – Среднее время между сбоями системы.

MTTR (Mean Time To Repair) – Среднее время восстановления системы после сбоя.

Failover – Автоматическое переключение на резервные системы при сбое.

AES-256 (Advanced Encryption Standard) – Алгоритм симметричного шифрования с ключом длиной 256 бит.

DDoS (Distributed Denial of Service) – Атака на систему с целью вызвать отказ в обслуживании.

TTL (Time To Live) – Время жизни данных в кэше или пакета в сети.

HTTP (HyperText Transfer Protocol) – Протокол для передачи данных в вебе.

Kubernetes – Система оркестрации контейнеров для управления масштабируемыми приложениями.

GZIP – Алгоритм сжатия данных для уменьшения размера файлов.

Brotli – Современный алгоритм сжатия данных, более эффективный, чем GZIP.

Jira – Инструмент для управления проектами и задачами, часто используемый в Agile-командах.

CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart)/Капча – автоматизированный тест Тьюринга, предназначенный для различения людей и ботов.

Критические сбои – сбои в системе, которые напрямую влияют на её основную функциональность, делают её недоступной или нарушают её работу на уровне, который затрудняет выполнение ключевых операций.

Некритические сбои – сбои, которые не мешают основным операциям системы, но могут повлиять на дополнительные функции или удобство пользователя.

Инвалидация кэша – это процесс удаления всех кешированных объектов, связанных с изменениями в состоянии модели.

Шардирование данных — это принцип проектирования базы данных, при котором данные разбиваются на части и размещаются на разных шардах. Каждый шард представляет собой отдельный узел внутри кластера, который может состоять из одной или нескольких реплик.

1.4 References (Ссылки)

- Разрабатываемый веб-сайт: <https://aliexpress.ru/>
- Веб-сайт разработчиков AliExpress Russia: <https://it.aliexpress.ru/>
- Payment Card Industry Data Security Standard (PCI DSS): https://docs-prv.pcisecuritystandards.org/PCI%20DSS/Standard/PCI-DSS-v4_0_1.pdf
- Федеральный закон "О персональных данных" от 27.07.2006 N 152-ФЗ: https://www.consultant.ru/document/cons_doc_LAW_61801/
- Федеральный закон "О внесении изменений в отдельные законодательные акты Российской Федерации в части уточнения порядка обработки персональных данных в информационно-телекоммуникационных сетях" от 21.07.2014 N 242-ФЗ: https://www.consultant.ru/document/cons_doc_LAW_165838/
- Закон РФ от 07.02.1992 N 2300-1 "О защите прав потребителей": https://www.consultant.ru/document/cons_doc_LAW_305/
- Постановление Правительства РФ от 31.12.2020 N 2463: https://www.consultant.ru/document/cons_doc_LAW_373622/
- Федеральный закон "О национальной платежной системе" от 27.06.2011 N 161-ФЗ: https://www.consultant.ru/document/cons_doc_LAW_115625/
- Федеральный закон "О рекламе" от 13.03.2006 N 38-ФЗ: https://www.consultant.ru/document/cons_doc_LAW_58968/
- Федеральный закон "О применении контрольно-кассовой техники при осуществлении расчетов в Российской Федерации" от 22.05.2003 N 54-ФЗ: https://www.consultant.ru/document/cons_doc_LAW_42359/

1.5 Overview (Обзор документа)

Документ структурирован следующим образом:

- Раздел 2 (Overall Description) описывает общую информацию о сайте AliExpress Russia, его пользователях и зависимостях от внешних сервисов.
- Раздел 3 содержит функциональные требования, включая требования пользователей и владельцев сайта и нефункциональные требования, такие как производительность, безопасность и доступность.

2. Overall Description (Общее описание)

2.1 Product functions (Функциональность продукта)

Разработанный веб-сайт должен поддерживать:

- Просмотр, поиск и фильтрацию товаров.
- Оформление и оплату заказов.
- Взаимодействие с продавцами (чат, отзывы).
- Управление заказами и возвратами.
- Инструменты аналитики и управления для продавцов.
- Программы лояльности, скидки и акции.
- Поддержку различных способов оплаты и доставки.

2.2 User characteristics (Описание пользователей)

Основные группы пользователей:

1. Покупатели:

1.1 Неавторизованные пользователи:

- Искать товары.
- Просматривать страницы товаров, изучать описание, фото и характеристики.
- Читать отзывы и оценки других покупателей.
- Ознакомиться с условиями доставки и оплаты.
- Добавлять товары в корзину (данные о товарах хранятся в локальном хранилище браузера).
- Авторизоваться или зарегистрироваться для оформления заказа.

1.2 Авторизованные пользователи:

- Все действия, доступные неавторизованным пользователям.
- Оформлять заказы и оплачивать товары.
- Отслеживать статус заказов.
- Оставлять отзывы и оценки товаров.
- Настраивать личный профиль.
- Взаимодействовать с продавцами через чат.
- Сохранять избранные товары и подписываться на продавцов.

2. Продавцы:

- Размещать и редактировать товары.
- Управлять заказами: подтверждать, отправлять, отслеживать статус, обрабатывать возвраты.
- Отвечать на вопросы покупателей в чате.
- Получать и анализировать отзывы на свои товары.
- Настраивать скидки, акции и купоны.
- Анализировать статистику продаж и доходности.

3. Администраторы платформы:

- Управлять акциями, скидками и промо-кампаниями.

4. Модераторы платформы:

- Проверять контент на соответствие правилам площадки.

5. Техническая поддержка:

- Обрабатывать обращения пользователей.
- Консультировать покупателей и продавцов.
- Решать проблемы с заказами, оплатами и доступом к аккаунтам.

2.3 Assumptions and dependencies (Влияющие факторы и зависимости)

1. Зависимости от сторонних сервисов:

- Платежные системы (Мир, СБП, Visa, MasterCard, ЮMoney и др.) через [API](#).
- Логистические компании (Почта России, СДЭК), трекинг-номера для отслеживания посылок.
- Сервисы аутентификации (Google, Яндекс, VK).
- [Капча](#) для предотвращения ботов (reCAPTCHA v3).

2. Технические зависимости:

- Использование [CDN](#) для ускорения загрузки страниц.
- Необходимость масштабируемой серверной инфраструктуры.

3. Регуляторные ограничения:

- Соответствие требованиям обработки персональных данных (152-ФЗ) – шифрование персональных данных ([AES-256](#)), возможность удаления аккаунта по запросу пользователя.
- Соответствие требованиям безопасности платежных данных (PCI DSS).

2.4 Constraints (Ограничения)

2.4.1 Все персональные данные должны обрабатываться в соответствии с законодательством (152-ФЗ).

2.4.2 Хранение данных пользователей из России должно осуществляться на серверах в РФ (242-ФЗ).

2.4.3 Система должна поддерживать возвраты товаров и средств в соответствии с ФЗ №2300-1. Условия возврата должны соответствовать Постановлению Правительства РФ №2463.

2.4.4 Для пользователей из России и СНГ система должна учитывать: требования к хранению данных (152-ФЗ, 242-ФЗ), национальные стандарты платежных систем (161-ФЗ), ограничения на продажу определенных товаров (38-ФЗ, 54-ФЗ).

2 Specific Requirements (Спецификация требований)

3.1 Functionality (Функциональные требования)

Поиск и фильтрация товаров:

3.1.1 Система должна предоставлять любому пользователю возможность поиска товаров по ключевым словам.

3.1.2 Модуль поиска должен быть нечувствителен к регистру. Пример: запросы «смартфон» и «Смартфон» должны давать одинаковые результаты.

3.1.3 Система должна поддерживать исправление опечаток в поисковых запросах любых пользователей. Например, если пользователь вводит «смортфон», система должна предложить «смартфон».

3.1.4 Система должна отображать автодополнение при вводе запроса на основе популярных запросов и истории поиска пользователя (для авторизованных пользователей).

3.1.5 При поиске система должна учитывать синонимы и альтернативные названия товаров и включать их в релевантные результаты.

3.1.6 Модуль поиска должен поддерживать русский и английский языки, с возможностью расширения на другие языки в будущем.

3.1.7 Система должна сохранять историю поисковых запросов и предоставлять возможность повторного использования ранее введенных запросов для авторизованных пользователей.

3.1.8 Модуль поиска должен сортировать результаты по релевантности, учитывая соответствие ключевым словам, популярности товара, рейтинга товара и количеству отзывов, актуальности.

3.1.9 Система должна предоставлять возможность пользователю применять фильтры (категория и подкатегория, ценовой диапазон, рейтинг товара, бренд, способ доставки) при поиске товаров.

3.1.10 Система должна предоставлять любому пользователю возможность применять несколько фильтров при поиске одновременно.

3.1.12 Система должна предоставлять любому пользователю возможность сбросить все фильтры одним действием.

3.1.13 Модуль результатов поиска должен отображать результаты в виде списка или сетки, с возможностью переключения между видами.

Пользовательская регистрация и авторизация:

3.1.14 Система должна позволять неавторизованным пользователям создавать учетные записи с использованием электронной почты/номера телефона и пароля.

3.1.15 Система должна использовать тест, что запрос выполняется человеком.

3.1.16 Система должна поддерживать авторизацию пользователей через сторонние сервисы.

3.1.17 Система должна предоставлять возможность авторизованным пользователям восстановление пароля через электронную почту или номер телефона, но не более трех запросов.

3.1.18 Система должна устанавливать ограничение попыток входа: блокировка на 5 минут после 3 неудачных попыток.

3.1.19 В системе должно быть реализовано логирование всех попыток входа (успешных/неудачных) с указанием IP, времени и устройства.

3.1.20 Система должна позволять авторизованному пользователю сменить email (с повторной верификацией).

Оформление заказа:

3.1.21 Система должна позволять любому пользователю добавлять товары в корзину со страницы товара или из списка результатов.

3.1.22 Система должна устанавливать максимальное количество одного товара в корзине.

3.1.23 Система должна предоставлять любому пользователю возможность увеличивать/уменьшать количество товаров через кнопки «+» и «-» или вручную вводить значение.

3.1.24 Система должна предоставлять любому пользователю возможность удалить товар из корзины в один клик.

3.1.25 Система должна автоматически пересчитывать сумму при изменении количества товаров или выборе доставки.

3.1.26 Система должна сохранять корзину для авторизованных пользователей между сессиями.

3.1.27 Система должна сохранять корзину для неавторизованных в локальном хранилище браузера до 7 дней.

3.1.28 Система должна предоставлять возможность авторизованному пользователю выбрать способ оплаты.

3.1.29 Система должна предоставлять возможность авторизованному пользователю выбрать способ доставки.

3.1.30 Система должна предоставлять авторизованному пользователю возможность пользователю отслеживать заказ.

3.1.31 Система должна предоставлять авторизованному пользователю возможность ввести адрес вручную.

3.1.32 Система должна предоставлять авторизованному пользователю возможность выбрать адрес доставки из сохраненных.

3.1.33 Система должна поддерживать автодополнение адреса.

3.1.34 Система должна предоставлять пользователю возможность проверить детали заказа и подтвердить его.

3.1.35 Система должна предоставлять авторизованным пользователям возможность ввести промокод при оформлении заказа для получения скидки.

3.1.36 Система должна предоставлять авторизованному пользователю возможность получить уведомление об успешном оформлении после нажатия «Оформить заказ». На email/номер телефона отправляется письмо/SMS с деталями заказа и номером отслеживания (если доступно).

Отзывы:

3.1.37 Система должна позволять авторизованным пользователям оставлять отзывы и оценки для товаров.

3.1.38 Система должна автоматически проверять отзывы на отсутствие нецензурной лексики и спама.

3.1.39 Система должна предоставлять пользователю возможность отфильтровать и отсортировать отзывы по дате, оценке, только с медиафайлами.

3.1.40 Система должна отображать средний рейтинг товара на основе оценок пользователей. Рейтинг автоматически пересчитывается при добавлении нового отзыва.

3.1.41 Система должна отображать рейтинг каждого товара в карточке товара, результатах поиска и на странице продавца.

3.1.42 Система должна отображать рейтинг продавца и количество подписчиков. Авторизованные подписчики получают уведомления о новых товарах и акциях.

3.1.43 Система должна автоматически проверять ограничение: 1 отзыв на товар от одного пользователя.

Управление профилем пользователя:

- 3.1.44 Система должна предоставлять авторизованному пользователю зайти в личный кабинет.
- 3.1.45 Система должна предоставлять пользователям возможность управлять своими данными и настройками через личный кабинет.
- 3.1.46 Система должна позволять пользователю редактировать свои личные данные (имя, фамилия, email, телефон, паспортные данные).
- 3.1.47 Система должна предоставлять пользователю просмотр истории заказов и текущих заказов.
- 3.1.48 Система должна предоставлять пользователю возможность сохранения избранных товаров и продавцов.
- 3.1.49 Система должна предоставлять пользователям возможность отслеживать свои заказы и управлять ими.
- 3.1.50 Система должна позволять отменить заказ до его отправки.
- 3.1.51 Система должна позволять связаться с продавцом, чтобы запросить возврат или обмен товара.
- 3.1.52 Система должна уведомлять пользователей о важных событиях, связанных с их аккаунтом и заказами.
- 3.1.53 Система должна предоставлять авторизованным пользователям возможность обращаться в службу поддержки.

3.2 Usability (Требования к удобству использования)

- 3.2.1 Интерфейс должен быть, основан на знакомых пользователям паттернах взаимодействия, логично структурирован, с четкими инструкциями.
- 3.2.2 Единый стиль ошибок и предупреждений (цвета, иконки).
- 3.2.3 В системе должна быть реализована удобная навигация (доступная в основных разделах страницы, дающая пользователю однозначное представление о том, куда ведет ссылка), позволяющая пользователю перейти в нужный раздел.
- 3.2.4 Система должна быть адаптирована под работу с устройств с разным расширением экрана. (Мобильные устройства: 320px–768px, планшеты: 769px–1024px, десктоп 1025px+).
- 3.2.5 В системе должен быть реализован [header](#), включающий в себя логотип, поисковую строку, корзину, ссылку на профиль, и [footer](#) с возможностью перейти на основные страницы сайта (контакты поддержки, ссылки на политики (конфиденциальность, возврат), социальные сети).
- 3.2.6 Система должна предоставлять пользователю возможность зарегистрироваться и совершить покупку в течение 10 минут после первого посещения сайта.
- 3.2.7 Система должна быть адаптирована к региональным форматам, соответствующем региону пользователя.
- 3.2.8 Соответствие стандартам [WCAG 2.1](#) для доступности.

3.2.9 Система должна предоставлять веб-интерфейс, совместимый с последними версиями браузеров Chrome 133+, Firefox 135+, Safari и Edge 133+.

3.2.10 Система должна обновляться в реальном времени результаты поиска при применении или изменении фильтров без перезагрузки страницы.

3.3 Reliability (Требования к надежности)

3.3.1 Система должна быть доступна 99.9% времени в течение года (максимальное время простоя — 8 часов 45 минут в год).

3.3.2 Плановое техническое обслуживание должно проводиться в период минимальной нагрузки и не чаще 1 раза в месяц.

3.3.3 Среднее время между [критическими сбоями \(MTBF\)](#) должно составлять не менее 1000 часов. Для [некритичных компонентов](#) допустимо $MTBF \geq 500$ часов.

3.3.4 Среднее время восстановления после сбоя ([MTTR](#)) не должно превышать 1 час для критических сбоев. Для некритичных компонентов — $MTTR \leq 4$ часа.

3.3.5 Автоматическое переключение на резервные серверы при сбоях ([failover](#)) в течение 5 минут.

3.3.6 Должно производиться ежедневное резервное копирование и восстановление баз данных и пользовательских файлов. Восстановление данных из резервной копии — не более 2 часов.

3.3.7 Система должна использовать шифрование данных. [TLS 1.3](#) для передачи данных. Шифрование конфиденциальных данных (пароли, платежные реквизиты) в хранилище с использованием [AES-256](#).

3.3.8 Для защиты от [DDoS](#) атак необходимо использовать облачные сервисы для фильтрации трафика.

3.4 Performance (Требования к производительности)

3.4.1 Максимальное время ответа для пользовательских операций — 2 секунды при скорости интернет-соединения 100 Мбит/с.

3.4.2 Среднее время ответа по всем операциям — 500 мс при скорости интернет-соединения 100 Мбит/с.

3.4.3 Время загрузки страницы для десктопа — $\leq 1,5$ секунды, для мобильных устройств — ≤ 3 секунды (при 3G-соединении).

3.4.4 Обработка 10 000 поисковых запросов в секунду в пиковые периоды.

3.4.5 Система должна кэшировать часто запрашиваемые данные. Результаты поиска и фильтрации — [TTL](#) 15 минут, статические ресурсы — TTL 1 год с хешированием версий, [API](#)-ответы для неавторизованных пользователей — TTL 5 минут. Использование стратегии [инвалидации кэша](#) при изменении данных.

3.4.6 Система должна быть масштабируемой для обработки растущего количества товаров и пользователей. Горизонтальное масштабирование: добавление серверов при нагрузке больше 80% (на основе метрик CPU/RAM),

поддержка работы в облачной среде с использованием [Kubernetes](#). Вертикальное масштабирование: [шардирование данных](#) по категориям товаров.

3.4.8 Время обработки заказа (от подтверждения до ответа платежного шлюза) — не более 5 секунд. Асинхронная обработка некритичных операций (например, отправка email-уведомлений).

3.4.9 Оптимизация ресурсов, такая как, сжатие данных, например, [GZIP/Brotli](#) для текстовых ресурсов, использование [CDN](#) для статических файлов и медиаконтента и геораспределенные точки присутствия (Россия, СНГ).

3.5 Design Constraints (Ограничения разработки)

3.5.1 Технологический стек: frontend должен быть разработан с помощью [JavaScript/TypeScript](#) с использованием фреймворка [React 18+](#), [CSS-in-JS](#) (Styled Components) или [Sass](#), [Webpack 5](#), [Babel](#), [ESLint](#), библиотеки core-js 3.32.2+; backend должен быть реализован с помощью языка [Java 17+](#) с фреймворком [Spring Boot 3+](#), средство для управления базами данных – [PostgreSQL](#) и [Redis](#) (для кэширования).

3.5.2 Процесс разработки должен следовать методологии Scrum ([Agile](#)) с двухнедельными спринтами.

3.5.3 Для управление задачами используется [Jira](#) с интеграцией в [GitHub](#).

3.5.4 Для управления версиями кода должен использоваться Git. Должен быть установлен обязательный [Code Review](#) перед [мерджем](#) в main и защита ветки main (запрет прямых [пушей](#)).

3.5.5 Для непрерывной интеграции и доставки ([CI/CD](#)) должен использоваться [Jenkins](#) + [GitHub Actions](#) (для автоматизации тестов). Для контейнеризации должен использоваться [Docker](#).

3.5.6 При разработке необходимо проводить юнит-тесты ([JUnit 5](#)) с покрытием $\geq 80\%$.

3.6 Interfaces (Интерфейсы)

3.6.1 User Interfaces (Пользовательские интерфейсы)

- Пользовательский интерфейс должен предоставляться в виде веб-страниц, доступных через стандартные веб-браузеры

3.6.2 Hardware Interfaces (Аппаратные интерфейсы)

- Система должна быть совместима с облачными провайдерами.
- Система должна поддерживать распределенных файловых систем.

3.6.3 Software Interfaces (Программные интерфейсы)

- Система должна интегрироваться с платежными шлюзами с поддержкой токенизации платежных данных (PCI DSS compliance).
- Система должна поддерживать [API](#) для взаимодействия с внешними сервисами. Автоматический расчет сроков и стоимости доставки на основе данных партнеров.

- Система должна поддерживать авторизацию через OAuth 2.0 (Google, VK, Яндекс, Mail.ru, Одноклассники).

3.6.4 Communications Interfaces (Сетевые интерфейсы)

- Система должна использовать протокол [HTTPS](#) с использованием [TLS](#) 1.3+ для всех внешних запросов для безопасной передачи данных.
- Система должна поддерживать WebSocket для realtime-обновлений.
- Минимальная пропускная способность для серверов должна быть 1 Гбит/с.

3.7 Licensing Requirements (Требования к лицензированию)

- Система должна быть разработана как проприетарное программное обеспечение с закрытым исходным кодом.
- Использование сторонних библиотек должно соответствовать их лицензионным соглашениям.

Риски

Ресурсные:

- RR1. Нехватка квалифицированных специалистов.
- RR2. Превышение бюджета из-за роста стоимости серверов, лицензий или внешних сервисов.

Бизнес-риски:

- BR1. Быстрое внедрение новых технологий конкурентами.
- BR2. Разработка функций, не востребованных пользователями.
- BR3. Сбои API платежных систем, логистических сервисов.

Технические:

- TR1. Некорректная архитектура, затрудняющая масштабирование системы.
- TR2. Выбор неподходящих технологий, что усложняет поддержку и развитие проекта.

Политические:

- PR1. Введение новых требований к обработке персональных данных.
- PR2. Ограничение работы международных сервисов.

Форс-мажор:

- FM1. Природные катастрофы, влияющие на работу серверных мощностей.
- FM2. Массовые сбои в работе дата-центров и облачных провайдеров.

Риск	Оценка вероятности	Оценка ущерба (руб/день)
RR1	0.4	8 000 000
RR2	0.5	12 000 000
BR1	0.6	20 000 000
BR2	0.3	10 000 000
BR3	0.4	15 000 000

TR1	0.35	14 000 000
TR2	0.25	9 000 000
PR1	0.4	18 000 000
PR2	0.35	25 000 000
FM1	0.1	50 000 000
FM2	0.15	40 000 000

Требования

Требование	Описание	Приоритет	Трудоемкость	Стабильность	Риск
3.1.1	Поиск товаров по ключевым словам	Must	40	Высокая	TR1
3.1.2	Нечувствительность к регистру	Should	10	Высокая	TR1
3.1.3	Исправление опечаток в поиске	Should	30	Средняя	TR1
3.1.4	Автодополнение запроса	Should	25	Средняя	TR2
3.1.5	Учет синонимов и альтернативных названий	Should	20	Средняя	TR2
3.1.6	Поддержка русского и английского языков	Should	15	Высокая	TR2
3.1.7	Сохранение истории поиска	Could	15	Средняя	BR2
3.1.8	Сортировка результатов по релевантности	Must	40	Высокая	TR1
3.1.9	Фильтрация товаров по категориям и цене	Must	30	Высокая	TR1
3.1.10	Применение нескольких фильтров	Should	25	Высокая	TR1
3.1.11	Сброс всех фильтров	Should	10	Высокая	TR1
3.1.12	Отображение результатов списком или сеткой	Could	15	Средняя	BR2
3.1.13	Регистрация с email/телефоном	Must	20	Высокая	PR1
3.1.14	Защита от ботов (CAPTCHA)	Must	15	Средняя	PR1
3.1.15	Авторизация через сторонние сервисы	Should	25	Средняя	PR2
3.1.16	Восстановление пароля	Must	20	Высокая	PR1
3.1.17	Ограничение попыток входа	Must	10	Высокая	TR1
3.1.18	Логирование попыток входа	Must	15	Высокая	TR1
3.1.19	Смена email с верификацией	Should	15	Высокая	PR1

3.1.20	Добавление товаров в корзину	Must	20	Высокая	BR3
3.1.21	Ограничение количества одного товара	Must	10	Средняя	BR3
3.1.22	Изменение количества товаров	Must	15	Высокая	TR1
3.1.23	Удаление товара из корзины	Must	10	Высокая	TR1
3.1.24	Пересчет суммы при изменении количества	Must	15	Высокая	TR1
3.1.25	Сохранение корзины для авторизованных	Must	20	Высокая	BR3
3.1.26	Сохранение корзины для неавторизованных	Should	15	Средняя	BR3
3.1.27	Выбор способа оплаты	Must	20	Высокая	BR3
3.1.28	Выбор способа доставки	Must	20	Высокая	BR3
3.1.29	Отслеживание заказа	Should	25	Высокая	BR3
3.1.30	Ввод адреса вручную	Must	15	Высокая	TR1
3.1.31	Выбор сохраненного адреса	Should	10	Средняя	TR1
3.1.32	Автодополнение адреса	Could	15	Средняя	TR1
3.1.33	Проверка и подтверждение заказа	Must	20	Высокая	BR3
3.1.34	Ввод промокода	Must	10	Средняя	BR2
3.1.35	Уведомление об успешном оформлении	Should	15	Высокая	BR3
3.1.36	Оставление отзывов	Should	25	Высокая	BR2
3.1.37	Проверка отзывов на спам	Should	20	Высокая	TR1
3.1.38	Фильтрация отзывов	Could	15	Средняя	TR1
3.1.39	Отображение среднего рейтинга товара	Must	20	Высокая	TR1
3.1.40	Рейтинг товара в карточке и поиске	Should	20	Высокая	TR1
3.1.41	Рейтинг продавца	Should	20	Высокая	BR2
3.1.42	Ограничение: 1 отзыв на товар от пользователя	Must	15	Высокая	TR1
3.1.43	Доступ в личный кабинет	Must	15	Высокая	PR1
3.1.44	Управление данными и настройками	Must	20	Высокая	PR1

3.1.45	Редактирование личных данных	Must	15	Высокая	PR1
3.1.46	Просмотр истории заказов	Must	20	Высокая	BR3
3.1.47	Сохранение избранных товаров	Should	15	Средняя	BR3
3.1.48	Отслеживание заказов	Must	25	Высокая	BR3
3.1.49	Отмена заказа до отправки	Must	20	Высокая	BR3
3.1.50	Связь с продавцом	Must	20	Высокая	BR3
3.1.51	Уведомления о событиях	Should	20	Высокая	BR3
3.1.52	Обращение в службу поддержки	Must	25	Высокая	BR3
3.1.53	Фильтрация товаров по категориям и цене	Should	30	Высокая	TR1
3.2.1	Логично структурированный интерфейс	Must	30	Высокая	BR2
3.2.2	Единый стиль ошибок и предупреждений	Must	15	Высокая	BR2
3.2.3	Удобная навигация	Must	25	Высокая	BR2
3.2.4	Адаптивность интерфейса	Must	30	Высокая	TR1
3.2.5	Header/Footer с ключевыми ссылками	Must	20	Высокая	TR1
3.2.6	Регистрация и покупка ≤ 10 минут	Must	20	Средняя	BR2
3.2.7	Адаптация к региональным форматам	Should	25	Средняя	PR1
3.2.8	Соответствие стандартам WCAG 2.1	Should	30	Высокая	PR1
3.2.9	Совместимость с последними браузерами	Must	20	Высокая	TR1
3.2.10	Обновление результатов поиска без перезагрузки	Should	25	Высокая	TR1
3.3.1	Доступность 99.9% времени	Must	40	Высокая	FM2
3.3.2	Техническое обслуживание 1 раз в месяц	Should	15	Средняя	FM2
3.3.3	MTBF ≥ 1000 часов (крит.) / 500 часов (некрит.)	Must	30	Высокая	FM2
3.3.4	MTTR ≤ 1 час (крит.) / 4 часа (некрит.)	Must	20	Высокая	FM2
3.3.5	Failover за 5 минут	Must	30	Высокая	FM2

3.3.6	Ежедневное резервное копирование	Should	20	Высокая	FM2
3.3.7	TLS 1.3 + AES-256 для шифрования	Must	25	Высокая	TR1
3.3.8	Защита от DDoS-атак	Must	30	Высокая	FM2
3.4.1	Максимальное время ответа ≤ 2 сек	Must	30	Высокая	TR1
3.4.2	Среднее время ответа ≤ 500 мс	Must	30	Высокая	TR1
3.4.3	Загрузка страницы $\leq 1,5$ сек (десктоп), ≤ 3 сек (моб.)	Must	25	Высокая	TR1
3.4.4	Обработка 10 000 запросов в секунду	Must	35	Высокая	TR1
3.4.5	Кэширование данных	Must	30	Высокая	TR1
3.4.6	Горизонтальное и вертикальное масштабирование	Must	40	Высокая	TR1
3.4.8	Обработка заказа ≤ 5 секунд	Must	25	Высокая	TR1
3.4.9	Оптимизация ресурсов (CDN, сжатие)	Should	30	Высокая	TR1
3.5.1	Определенный технологический стек	Should	50	Высокая	TR2
3.5.2	Методология разработки Scrum	Should	20	Высокая	RR1
3.5.3	Jira для управления задачами	Could	15	Высокая	RR1
3.5.4	Git + Code Review перед мерджем	Must	20	Высокая	RR1
3.5.5	CI/CD: Jenkins + GitHub Actions	Must	30	Высокая	TR2
3.5.6	Юнит-тестирование с покрытием $\geq 80\%$	Must	40	Высокая	TR2

UML UseCase-диаграмма

Предусловия: Пользователь находится на главной странице сайта или в разделе поиска.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь вводит ключевое слово в строку поиска. 2. Система предлагает автодополнение на основе популярных запросов. 3. Пользователь выбирает запрос из предложенных вариантов или завершает ввод. 4. Система выполняет поиск и отображает релевантные результаты. 5. Пользователь может применить фильтры. 6. Система обновляет результаты поиска без перезагрузки страницы. 7. Пользователь выбирает нужный товар и переходит на страницу товара.
Альтернативный поток: <ol style="list-style-type: none"> 1. Пользователь выбирает категорию товаров вместо ввода запроса. 2. Система отображает список товаров, соответствующих категории. 3. Пользователь применяет фильтры и сортировку. 4. Система обновляет результаты.

Прецедент: Поиск товаров на сайте
ID: 2
Краткое описание: Пользователь хочет выполнить поиск товара, не беспокоясь о регистре букв.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь находится на главной странице сайта или в разделе поиска.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь вводит поисковый запрос. 2. Система приводит запрос к единому регистру. 3. Выполняется поиск товаров независимо от регистра. 4. Пользователь получает релевантные результаты.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если введен запрос, содержащий разные регистры, система корректно выполняет поиск.

Прецедент: Автоматическое исправление опечаток в поисковом запросе
ID: 3
Краткое описание: Пользователь вводит поисковый запрос с опечатками, и система автоматически предлагает исправленный вариант.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь находится на главной странице сайта или в разделе поиска.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь вводит запрос с опечатками. 2. Система анализирует запрос и предлагает исправленный вариант. 3. Пользователь может подтвердить исправленный запрос или оставить оригинальный. 4. Система выполняет поиск с исправленным запросом и отображает релевантные результаты.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если опечатка нераспознаваема, система выполняет поиск по оригинальному запросу. 2. Если исправленный запрос все же не дает результатов, система предлагает альтернативные товары.

Прецедент: Автодополнение поискового запроса
ID: 4
Краткое описание: Пользователь вводит поисковый запрос, и система предлагает варианты автодополнения.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь начинает ввод поискового запроса.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь вводит первые символы запроса. 2. Система анализирует популярные запросы и историю поиска пользователя (если он авторизован). 3. Система отображает список вариантов автодополнения. 4. Пользователь выбирает один из предложенных вариантов или завершает ввод вручную.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если подходящих вариантов автодополнения нет, система не предлагает подсказки. 2. Если пользователь отключил историю поиска, система предлагает только популярные запросы.

Прецедент: Учет синонимов и альтернативных названий в поиске
ID: 5
Краткое описание: Пользователь вводит название товара, но система также учитывает синонимы и альтернативные названия.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь находится на странице поиска.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь вводит поисковый запрос. 2. Система проверяет синонимы и альтернативные названия. 3. Поиск выполняется сразу по нескольким вариантам. 4. Система отображает релевантные результаты, включая товары с разными названиями.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если товар с таким названием отсутствует, система предлагает альтернативные товары.

Прецедент: Учет синонимов и альтернативных названий в поиске
ID: 6
Краткое описание: Пользователь вводит название товара, но система также учитывает синонимы и альтернативные названия.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь находится на странице поиска.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь вводит поисковый запрос. 2. Система проверяет синонимы и альтернативные названия. 3. Поиск выполняется сразу по нескольким вариантам. 4. Система отображает релевантные результаты, включая товары с разными названиями.
Альтернативный поток:

1. Если товар с таким названием отсутствует, система предлагает альтернативные товары.
--

Прецедент: Поддержка русского и английского языков в поиске
ID: 7
Краткое описание: Пользователь выполняет поиск на русском или английском языке, и система корректно обрабатывает запрос.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь вводит запрос на одном из поддерживаемых языков.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь вводит запрос на русском языке. 2. Система выполняет поиск среди русскоязычных товаров. 3. Если есть совпадения на английском языке, система также их отображает. 4. Пользователь получает релевантные результаты.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если пользователь вводит запрос на английском языке, система также ищет русские эквиваленты. 2. Если товар найден только на одном языке, система предлагает его без перевода.

Прецедент: Сортировка результатов по релевантности
ID: 8
Краткое описание: Пользователь хочет отсортировать результаты поиска по релевантности, чтобы сначала видеть наиболее подходящие товары.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь выполнил поисковый запрос и видит список товаров.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь открывает выпадающий список сортировки. 2. Выбирает вариант «По релевантности». 3. Система пересчитывает релевантность товаров, учитывая: соответствие ключевым словам, популярность товара, рейтинг и количество отзывов, актуальность. 4. Система обновляет список товаров, отображая наиболее подходящие сверху.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если пользователь не выбирает сортировку, система использует сортировку по умолчанию. 2. Если товаров слишком много, система ограничивает количество отображаемых страниц.

Прецедент: Фильтрация товаров по категориям и цене
ID: 9
Краткое описание: Пользователь хочет отфильтровать товары по категории и цене, чтобы быстрее найти нужный вариант.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь находится на странице каталога или результатов поиска.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь открывает меню фильтров. 2. Выбирает категорию товара.

3. Устанавливает диапазон цен с помощью ползунка или ввода чисел.
4. Система применяет фильтры и обновляет список товаров.
Альтернативный поток:
1. Если в выбранной категории нет товаров в указанном диапазоне цен, система сообщает об этом.
2. Если пользователь изменяет фильтры, результаты обновляются в реальном времени.

Прецедент: Управление скидками и акциями продавцом
ID: 10
Краткое описание: Продавец хочет создать скидку или акцию на определенные товары.
Главный актер: Продавец
Второстепенные акторы: Нет
Предусловия: Продавец авторизован и имеет активные товары в каталоге.
Основной поток:
1. Продавец открывает раздел «Акции и скидки».
2. Выбирает товар или группу товаров.
3. Устанавливает размер скидки и срок ее действия.
4. Система проверяет параметры и применяет скидку.
Альтернативный поток:
1. Если скидка превышает допустимый лимит, система уведомляет об ошибке.
2. Если акция уже активна, продавец может ее изменить или удалить.

Прецедент: Сброс всех фильтров
ID: 11
Краткое описание: Пользователь хочет сбросить все примененные фильтры одним нажатием.
Главный актер: Продавец
Второстепенные акторы: Нет
Предусловия: Пользователь применил несколько фильтров.
Основной поток:
1. Пользователь нажимает кнопку «Сбросить все фильтры».
2. Система удаляет все активные фильтры.
3. Список товаров обновляется без ограничений.
Альтернативный поток:
1. Если фильтры уже сброшены, кнопка становится неактивной.

Прецедент: Отображение результатов списком или сеткой
ID: 12
Краткое описание: Пользователь хочет изменить представление списка товаров — в виде списка или сетки.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь находится на странице каталога или результатов поиска.
Основной поток:
1. Пользователь выбирает один из режимов отображения: сетка (крупные карточки товаров), список (компактное отображение с характеристиками).
2. Система обновляет отображение товаров в соответствии с выбранным режимом.

Прецедент: Добавление нового товара продавцом
ID: 13
Краткое описание: Продавец хочет добавить новый товар на платформу.
Главный актер: Продавец
Второстепенные акторы: Нет
Предусловия: Продавец авторизован и имеет доступ к личному кабинету.
Основной поток: <ol style="list-style-type: none"> 1. Продавец заходит в раздел «Добавить товар». 2. Вводит название, описание, цену, фото, параметры товара. 3. Система проверяет корректность данных. 4. Если данные корректны, товар отправляется на модерацию. 5. После успешной проверки товар появляется в каталоге.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если товар нарушает правила, система отклоняет заявку и сообщает причину. 2. Если не загружены обязательные поля (фото, описание и т. д.), система требует их заполнения.

Прецедент: Авторизация пользователя
ID: 14
Краткое описание: Пользователь хочет войти в свою учетную запись.
Главный актер: Неавторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь находится на странице входа.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь вводит email/номер телефона и пароль. 2. Система проверяет корректность данных. 3. Если данные верны, пользователь успешно входит в аккаунт.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если введены неверные данные, система сообщает об ошибке. 2. После 3 неудачных попыток система блокирует вход на 5 минут.

Прецедент: Защита от ботов
ID: 15
Краткое описание: Система должна проверять, является ли пользователь человеком, чтобы предотвратить автоматизированные действия со стороны ботов.
Главный актер: Неавторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь выполняет действие, требующее подтверждения.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь заполняет форму. 2. Система анализирует поведение пользователя и решает, требуется ли дополнительная проверка. 3. Если проверка нужна, система отображает CAPTCHA. 4. Пользователь выполняет задание CAPTCHA. 5. Система проверяет правильность ответа. 6. Если проверка пройдена, система разрешает продолжение действия.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если пользователь вводит неправильный ответ, система предлагает повторить попытку. 2. Если попыток слишком много, система временно блокирует выполнение запроса.

3. Если система подозревает бота, она может потребовать дополнительную проверку.
--

Прецедент: Авторизация через сторонние сервисы
ID: 16
Краткое описание: Система должна проверять, является ли пользователь человеком, чтобы предотвратить автоматизированные действия со стороны ботов.
Главный актер: Неавторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь выполняет действие, требующее подтверждения.
Основной поток: <ol style="list-style-type: none">Пользователь заполняет форму.Система анализирует поведение пользователя и решает, требуется ли дополнительная проверка.Если проверка нужна, система отображает CAPTCHA.Пользователь выполняет задание CAPTCHA.Система проверяет правильность ответа.Если проверка пройдена, система разрешает продолжение действия.
Альтернативный поток: <ol style="list-style-type: none">Если пользователь вводит неправильный ответ, система предлагает повторить попытку.Если попыток слишком много, система временно блокирует выполнение запроса.Если система подозревает бота, она может потребовать дополнительную проверку.

Прецедент: Восстановление пароля
ID: 17
Краткое описание: Пользователь забыл пароль и хочет восстановить доступ к учетной записи.
Главный актер: Неавторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь находится на странице входа и не помнит пароль.
Основной поток: <ol style="list-style-type: none">Пользователь нажимает «Забыли пароль?».Вводит email или номер телефона.Система отправляет код подтверждения на указанный email/номер.Пользователь вводит код и задает новый пароль.Система подтверждает изменение пароля.
Альтернативный поток: <ol style="list-style-type: none">Если пользователь вводит неверный код 3 раза, система предлагает повторный запрос.Если email/номер телефона не зарегистрирован, система сообщает об ошибке.

Прецедент: Удаление учетной записи
ID: 18
Краткое описание: Пользователь хочет удалить свою учетную запись.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь авторизован и находится в настройках аккаунта.
Основной поток: <ol style="list-style-type: none">Пользователь открывает раздел «Настройки аккаунта».

2. Нажимает «Удалить учетную запись».
3. Система запрашивает подтверждение через email/SMS.
4. Пользователь подтверждает удаление, и аккаунт деактивируется.
Альтернативный поток:
1. Если у пользователя есть активные заказы, система предлагает дождаться их завершения.
2. Если пользователь передумает в течение 30 дней, он может восстановить аккаунт.

Прецедент: Добавление товара в корзину
ID: 19
Краткое описание: Пользователь хочет добавить товар в корзину для последующего оформления заказа.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь находится на странице товара или в каталоге.
Основной поток:
1. Пользователь нажимает кнопку "Добавить в корзину".
2. Система добавляет товар в корзину.
3. Если товар уже в корзине, система увеличивает его количество.
Альтернативный поток:
1. Данные корзины синхронизируются с сервером.

Прецедент: Ограничение количества одного товара в корзине
ID: 20
Краткое описание: Пользователь не может добавить в корзину больше определенного количества единиц одного товара.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь находится на странице товара или в корзине.
Основной поток:
1. Пользователь добавляет товар в корзину.
2. Система проверяет, не превышен ли лимит на количество данного товара.
3. Если количество в пределах допустимого, товар добавляется в корзину.
4. Если лимит превышен, система уведомляет пользователя и не позволяет добавить больше товаров.

Прецедент: Изменение количества товаров в корзине
ID: 21
Краткое описание: Пользователь хочет изменить количество товаров в корзине перед оформлением заказа.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь находится в корзине, где уже добавлены товары.
Основной поток:
1. Пользователь изменяет количество товара с помощью кнопок «+» и «-» или вводит число вручную.
2. Система проверяет, не превышен ли лимит количества товара.
3. Система автоматически обновляет сумму заказа.
Альтернативный поток:

1. Если пользователь вводит 0, товар удаляется из корзины.
2. Если товар закончился на складе, система уведомляет пользователя и устанавливает максимальное доступное количество.

Прецедент: Удаление товара из корзины
ID: 22
Краткое описание: Пользователь хочет удалить товар из корзины перед оформлением заказа.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: В корзине есть добавленные товары.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь нажимает кнопку «Удалить» рядом с товаром. 2. Система удаляет товар из корзины. 3. Система обновляет итоговую сумму заказа.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если корзина пуста после удаления, система предлагает перейти в каталог товаров.

Прецедент: Пересчет суммы при изменении количества товаров
ID: 23
Краткое описание: Система должна автоматически пересчитывать сумму заказа при изменении количества товаров в корзине.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь добавляет товар в корзину со страницы товара или из каталога.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь добавляет товар в корзину или изменяет количество текущего товара. 2. Система автоматически пересчитывает общую сумму заказа. 3. Если пользователь изменяет способ доставки, система обновляет стоимость доставки.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если товар удаляется, сумма заказа пересчитывается без него. 2. Если пользователь применяет промокод, система обновляет сумму с учетом скидки.

Прецедент: Сохранение корзины для авторизованных пользователей
ID: 24
Краткое описание: Корзина пользователя сохраняется между сессиями, если он авторизован.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь авторизован и добавил товары в корзину.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь добавляет товары в корзину. 2. Покидает сайт или выходит из аккаунта. 3. При следующем входе в аккаунт корзина сохраняется в том же состоянии.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если товар больше не доступен, система уведомляет пользователя при следующем входе.

Прецедент: Сохранение корзины для неавторизованных пользователей
ID: 25
Краткое описание: Корзина пользователя сохраняется локально в браузере в течение 7 дней.
Главный актер: Неавторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь не авторизован и добавил товары в корзину.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь добавляет товары в корзину. 2. Система сохраняет корзину в локальном хранилище браузера. 3. При повторном посещении сайта в течение 7 дней корзина загружается автоматически.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если пользователь очистил кеш браузера, корзина будет пустой. 2. Если пользователь авторизуется, корзина объединяется с его сохраненной корзиной в аккаунте.

Прецедент: Выбор и изменение способа оплаты
ID: 26
Краткое описание: Пользователь хочет выбрать или изменить способ оплаты перед оформлением заказа.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь добавил товары в корзину и перешел к оформлению заказа.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь выбирает способ оплаты. 2. Вводит платежные данные или выбирает из ранее сохраненных. 3. Система проверяет корректность данных. 4. При успешной проверке система сохраняет данные. 5. Пользователь завершает оформление заказа.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если оплата не прошла, система предлагает повторить попытку или выбрать другой метод.

Прецедент: Выбор способа доставки
ID: 27
Краткое описание: Пользователь хочет выбрать удобный способ доставки при оформлении заказа.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь добавил товары в корзину и перешел к оформлению заказа.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь открывает страницу оформления заказа. 2. В разделе «Способ доставки» видит доступные варианты доставки. 3. Выбирает удобный вариант. 4. Система обновляет стоимость и сроки доставки. 5. Пользователь подтверждает выбор и переходит к оплате.
Альтернативный поток:

1. Если для указанного региона доступен только один способ доставки, система автоматически его выбирает.
2. Если пользователь изменяет адрес доставки, система пересчитывает доступные варианты.
3. Если выбранный способ доставки недоступен для конкретного товара (например, крупногабаритные товары нельзя доставить почтой), система уведомляет об этом.

Прецедент: Добавление нового адреса доставки
ID: 28
Краткое описание: Пользователь хочет добавить новый адрес для доставки.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь находится в разделе «Адреса доставки».
Основной поток:
1. Пользователь нажимает «Добавить новый адрес».
2. Вводит адрес и сохраняет его.
3. Система подтверждает сохранение адреса.
Альтернативный поток:
1. Если адрес введен некорректно, система сообщает об ошибке.

Прецедент: Оформление заказа
ID: 29
Краткое описание: Пользователь хочет оформить заказ на выбранные товары.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Платежная система, служба доставки
Предусловия: Пользователь добавил товары в корзину и перешел к оформлению.
Основной поток:
1. Пользователь выбирает способ доставки и оплаты.
2. Система рассчитывает итоговую сумму и отображает детали заказа.
3. Пользователь подтверждает заказ.
4. Система отправляет запрос в платежную систему.
5. После успешной оплаты система подтверждает заказ и уведомляет пользователя.
Альтернативный поток:
1. Если оплата не прошла, система предлагает повторить попытку.
2. Если пользователь применяет промокод, система пересчитывает сумму.

Прецедент: Использование промокода при оформлении заказа
ID: 30
Краткое описание: Пользователь хочет применить промокод для получения скидки.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь находится на странице оформления заказа.
Основной поток:
1. Пользователь вводит промокод в соответствующее поле.
2. Система проверяет корректность промокода.
3. Если код действителен, система пересчитывает итоговую сумму заказа.
4. Пользователь подтверждает заказ.
Альтернативный поток:

1. Если промокод недействителен или срок его действия истек, система выводит ошибку.
2. Если промокод уже использован, система сообщает об этом.

Прецедент: Оставление отзыва о товаре
ID: 31
Краткое описание: Пользователь хочет оставить отзыв о купленном товаре.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Модератор платформы
Предусловия: Пользователь купил товар и заказ был завершен.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь заходит в раздел «Заказы». 2. Выбирает товар и нажимает «Оставить отзыв». 3. Вводит текст отзыва, оценивает товар (1-5 звезд). 4. Система проверяет отзыв на спам и нецензурную лексику. 5. Если проверка пройдена, отзыв публикуется.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если отзыв содержит запрещенные слова, система предлагает его изменить. 2. Если товар еще не получен, кнопка «Оставить отзыв» недоступна.

Прецедент: Модерация отзывов
ID: 32
Краткое описание: Модератор проверяет отзывы пользователей на соответствие правилам.
Главный актер: Модератор
Второстепенные акторы: Нет
Предусловия: Пользователь оставил отзыв, он отправлен на проверку.
Основной поток: <ol style="list-style-type: none"> 1. Система автоматически проверяет отзыв на запрещенные слова и спам. 2. Если система не находит нарушений, отзыв публикуется. 3. Если есть подозрение на нарушение, отзыв отправляется модератору. 4. Модератор вручную проверяет отзыв и принимает решение
Альтернативный поток: <ol style="list-style-type: none"> 1. Если отзыв удален, пользователь получает уведомление.

Прецедент: Фильтрация и сортировка отзывов
ID: 33
Краткое описание: Пользователь хочет отфильтровать и отсортировать отзывы, чтобы увидеть наиболее полезные или актуальные.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: Пользователь находится на странице товара и видит раздел с отзывами
Основной поток: <ol style="list-style-type: none"> 1. Пользователь открывает раздел отзывов. 2. Выбирает фильтр. 3. Система обновляет список отзывов в соответствии с выбранным фильтром.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если по заданным критериям нет отзывов, система сообщает об этом. 2. Если пользователь снимает фильтры, система возвращает полный список отзывов.

Прецедент: Отображение среднего рейтинга товара
ID: 34
Краткое описание: Система должна рассчитывать средний рейтинг товара на основе пользовательских оценок.
Главный актер: Пользователь (неавторизованный, авторизованный)
Второстепенные акторы: Нет
Предусловия: На товар оставлено хотя бы несколько отзывов с оценками.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь заходит на страницу товара. 2. Система рассчитывает средний рейтинг на основе всех оценок. 3. Система отображает средний рейтинг и количество отзывов. 4. Если пользователь оставляет новый отзыв, система автоматически пересчитывает рейтинг.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если у товара пока нет оценок, система отображает «Нет отзывов». 2. Если отзыв удален модератором, рейтинг обновляется без учета этого отзыва.

Прецедент: Вход в личный кабинет
ID: 35
Краткое описание: Авторизованный пользователь хочет зайти в личный кабинет для управления своими данными и заказами.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь авторизован на сайте.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь нажимает на иконку профиля. 2. Система открывает страницу личного кабинета.

Прецедент: Получение уведомлений о статусе заказа
ID: 36
Краткое описание: Пользователь хочет получать уведомления о статусе своего заказа.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: У пользователя есть оформленный заказ.
Основной поток: <ol style="list-style-type: none"> 1. Система автоматически отправляет уведомления о статусе заказа. 2. Пользователь получает уведомление по email/SMS/в мобильном приложении. 3. При изменении статуса (отправка, доставка) система уведомляет пользователя.
Альтернативный поток: <ol style="list-style-type: none"> 1. Пользователь может отключить уведомления в настройках.

Прецедент: Оформление возврата товара
ID: 37
Краткое описание: Пользователь хочет вернуть товар после получения.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Служба поддержки, продавец

Предусловия: Пользователь получил заказ, возврат возможен в рамках политики магазина.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь открывает раздел «Заказы». 2. Выбирает заказ и нажимает «Оформить возврат». 3. Указывает причину возврата и загружает фото (если требуется). 4. Система проверяет, соответствует ли товар условиям возврата. 5. Если товар подлежит возврату, система предоставляет инструкции по отправке. 6. После получения товара продавец или платформа одобряет возврат. 7. Деньги возвращаются пользователю.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если возврат невозможен (например, истек срок), система сообщает об этом. 2. Если продавец отклоняет возврат, система предлагает пользователю обратиться в поддержку.

Прецедент: Подписка на продавца
ID: 38
Краткое описание: Пользователь хочет подписаться на продавца, чтобы получать уведомления о новых товарах и скидках.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь авторизован и находится на странице продавца.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь нажимает кнопку «Подписаться». 2. Система добавляет продавца в список подписок. 3. Пользователь получает уведомления о новых товарах и акциях.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если пользователь передумал, он может отписаться от продавца. 2. Если подписки отключены в настройках, уведомления не приходят.

Прецедент: Изменение персональных данных пользователя
ID: 39
Краткое описание: Пользователь хочет изменить свои личные данные.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь авторизован и находится в личном кабинете.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь открывает раздел «Личные данные». 2. Изменяет нужные поля. 3. Система запрашивает подтверждение (через email/SMS). 4. Пользователь подтверждает изменения, и они сохраняются.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если пользователь вводит некорректный email/телефон, система сообщает об ошибке. 2. Если подтверждение не пройдено, изменения не сохраняются.

Прецедент: Просмотр истории заказов
ID: 40
Краткое описание: Пользователь хочет посмотреть историю своих заказов.

Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: У пользователя есть завершенные заказы.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь открывает раздел «Заказы». 2. Видит список всех прошлых заказов. 3. Может просмотреть детали заказа, включая способ оплаты и доставки.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если у пользователя нет заказов, система сообщает об этом.

Прецедент: Добавление товара в избранное
ID: 41
Краткое описание: Пользователь хочет сохранить товар в список избранных.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь находится на странице товара или в каталоге.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь нажимает на значок «Избранное». 2. Система добавляет товар в список избранных. 3. В разделе «Избранное» пользователь видит сохраненные товары.
Альтернативный поток: <ol style="list-style-type: none"> 1. Если пользователь нажимает повторно, товар удаляется из списка. 2. Если пользователь не авторизован, система предлагает войти в аккаунт.

Прецедент: Отслеживание доставки заказа
ID: 42
Краткое описание: Пользователь хочет проверить статус доставки своего заказа.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Заказ был отправлен, и у него есть трек-номер.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь открывает раздел «Мои заказы». 2. Выбирает заказ и нажимает «Отследить доставку». 3. Система получает информацию о статусе доставки. 4. Пользователь видит текущее местоположение заказа.
Альтернативный поток: <ol style="list-style-type: none"> 5. Если информация о доставке временно недоступна, система сообщает об этом.

Прецедент: Отмена заказа
ID: 43
Краткое описание: Пользователь хочет отменить заказ до его отправки.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Заказ находится в статусе «Ожидает отправки».
Основной поток: <ol style="list-style-type: none"> 1. Пользователь открывает раздел «Заказы». 2. Выбирает заказ и нажимает «Отменить заказ». 3. Система запрашивает подтверждение. 4. После подтверждения заказ отменяется, деньги возвращаются пользователю.

Альтернативный поток:
1. Если заказ уже отправлен, отмена невозможна, но доступен запрос на возврат.
2. Если пользователь передумал, он может закрыть окно отмены заказа.

Прецедент: Связь с продавцом
ID: 44
Краткое описание: Пользователь хочет задать вопрос продавцу о товаре или условиях доставки.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Продавец
Предусловия: Пользователь находится на странице продавца.
Основной поток:
1. Пользователь нажимает кнопку «Связаться с продавцом».
2. Вводит сообщение.
3. Система отправляет запрос продавцу.
4. Продавец отвечает, пользователь получает уведомление.
Альтернативный поток:
1. Если сообщение содержит запрещенные слова, система уведомляет пользователя.

Прецедент: Настройка уведомлений пользователя
ID: 45
Краткое описание: Пользователь хочет настроить, какие уведомления он будет получать от платформы.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Нет
Предусловия: Пользователь авторизован и находится в личном кабинете.
Основной поток:
1. Пользователь открывает раздел «Уведомления».
2. Видит список доступных типов уведомлений (email, SMS, социальные сети).
3. Выбирает, какие уведомления хочет получать.
4. Система сохраняет настройки.
Альтернативный поток:
1. Если произошел сбой при сохранении, система предлагает повторить попытку.

Прецедент: Запрос технической поддержки
ID: 46
Краткое описание: Пользователь хочет обратиться в техническую поддержку по проблеме с сайтом или заказом.
Главный актер: Авторизованный пользователь
Второстепенные акторы: Техническая поддержка
Предусловия: Пользователь авторизован и находится в разделе поддержки.
Основной поток:
1. Пользователь открывает форму обращения.
2. Выбирает категорию проблемы.
3. Описывает проблему и прикрепляет файлы (если нужно).
4. Отправляет заявку.
5. Система создает тикет и передает его оператору поддержки.
Альтернативный поток:
1. Если тикет заполнен некорректно, система предлагает исправить ошибки.

- | |
|---|
| 2. Если проблема решена автоматически (FAQ, бот), система предлагает готовое решение. |
|---|

Вывод

Проделанная работа по составлению документа Software Requirements Specification (SRS) для веб-сайта AliExpress Russia позволила систематизировать и детализировать ключевые аспекты разработки платформы.