

Федеральное государственное автономное образовательное учреждение высшего
образования Национальный исследовательский университет ИТМО

Факультет программной инженерии и компьютерной техники

Архитектура программных систем

Лабораторная работа №1

Выполнил: студент группы Р3308,
Васильев Никита Алексеевич

Преподаватель: Перл Иван Андреевич

Санкт-Петербург 2025

Текст задания

Выбрать любую реально существующую систему и описать её в терминах UML. Желательно, чтобы система была не полностью информационной, но опиралась на информационную систему как показано в примере на лекции (Point of sale). Необходимо описать границы системы на разных уровнях, а также описать сценарии использования для нескольких Акторов.

Отчёт по работе должен содержать:

1. Титульный лист с указанием автора и номера группы
2. Само задание
3. Описание рассматриваемой системы с требованиями к ней
4. Формальное описание системы с необходимым количеством UML диаграмм
5. Словесное описание сценариев использования для рассматриваемых акторов

Рассматриваемая система

AliExpress Russia — Российская версия международной платформы электронной коммерции AliExpress, адаптированная для пользователей из России. Цель AliExpress Russia — предоставлять пользователям широкий ассортимент товаров от российских и зарубежных продавцов с удобным поиском, рекомендациями, локальными способами оплаты и доставкой. Описанная система позволяет покупателям заказывать и оценивать товары, а продавцам — размещать, продвигать и продавать продукцию на российском рынке.

Система состоит включает в себя цифровую платформу (каталог товаров, способы оплаты и т.п.) и физическую составляющие (постаматы, пункты выдачи, сторонние службы доставки).

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.

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.
- Интеграцию с логистическими сервисами и службами доставки, ориентированными на российский рынок.

AliExpress Russia — это многофункциональная электронная торговая платформа, ориентированная на российский рынок и доступная через браузерную версию сайта, а также мобильные приложения для iOS и Android.

Платформа оперирует распределённой информационной системой, которая обрабатывает данные о товарах, продавцах, покупателях, заказах, оплатах, доставке, отзывах и возвратах. Взаимодействие между компонентами реализуется через программные интерфейсы (API), обеспечивающие целостность и актуальность данных.

Для выполнения коммерческих операций система использует интеграцию с внешними платёжными и сервисными решениями, включая банковские платёжные системы и национальные способы оплаты, а также логистические сервисы и инфраструктуру партнёров доставки.

Платформа проектируется с учётом требований к масштабируемости, отказоустойчивости и безопасности, а также должна обеспечивать эффективную работу всех категорий пользователей: покупателей, продавцов, логистических партнёров и администраторов системы.

Влияющие факторы и зависимости:

1. Зависимости от сторонних сервисов:
 - Платёжные системы (Мир, СБП, Visa, MasterCard, ЮMoney и др.) через API.
 - Логистические компании (Почта России, СДЭК), трекинг-номера для отслеживания посылок.
 - Сервисы аутентификации (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 платёжных данных (PCI DSS).

Требования к системе

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.

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.

Нефункциональные требования

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

3.2.6 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 не критичных 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 (VK, Яндекс, Mail.ru, Одноклассники).

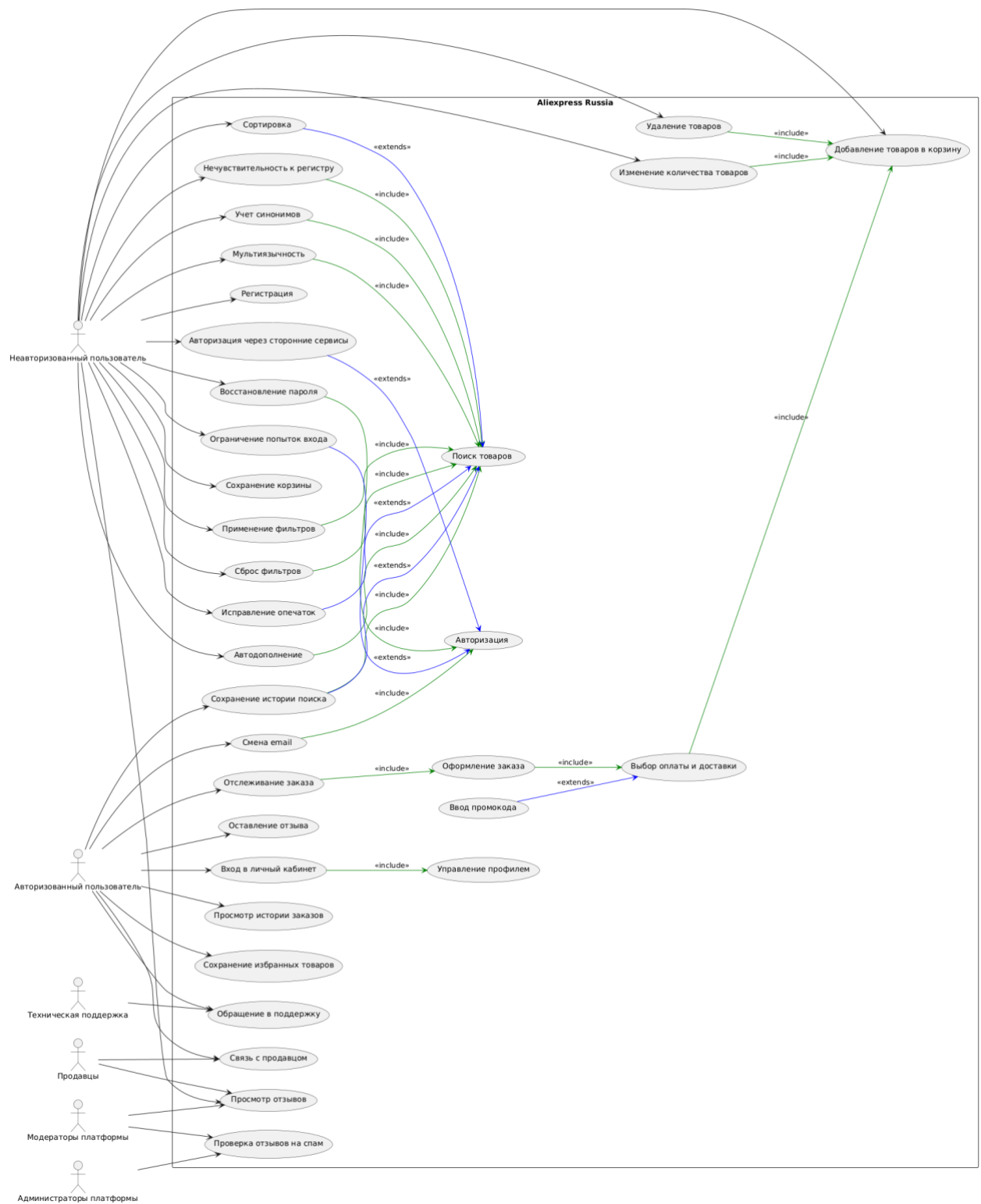
3.6.4 Communications Interfaces (Сетевые интерфейсы)

- 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.

Ограничения

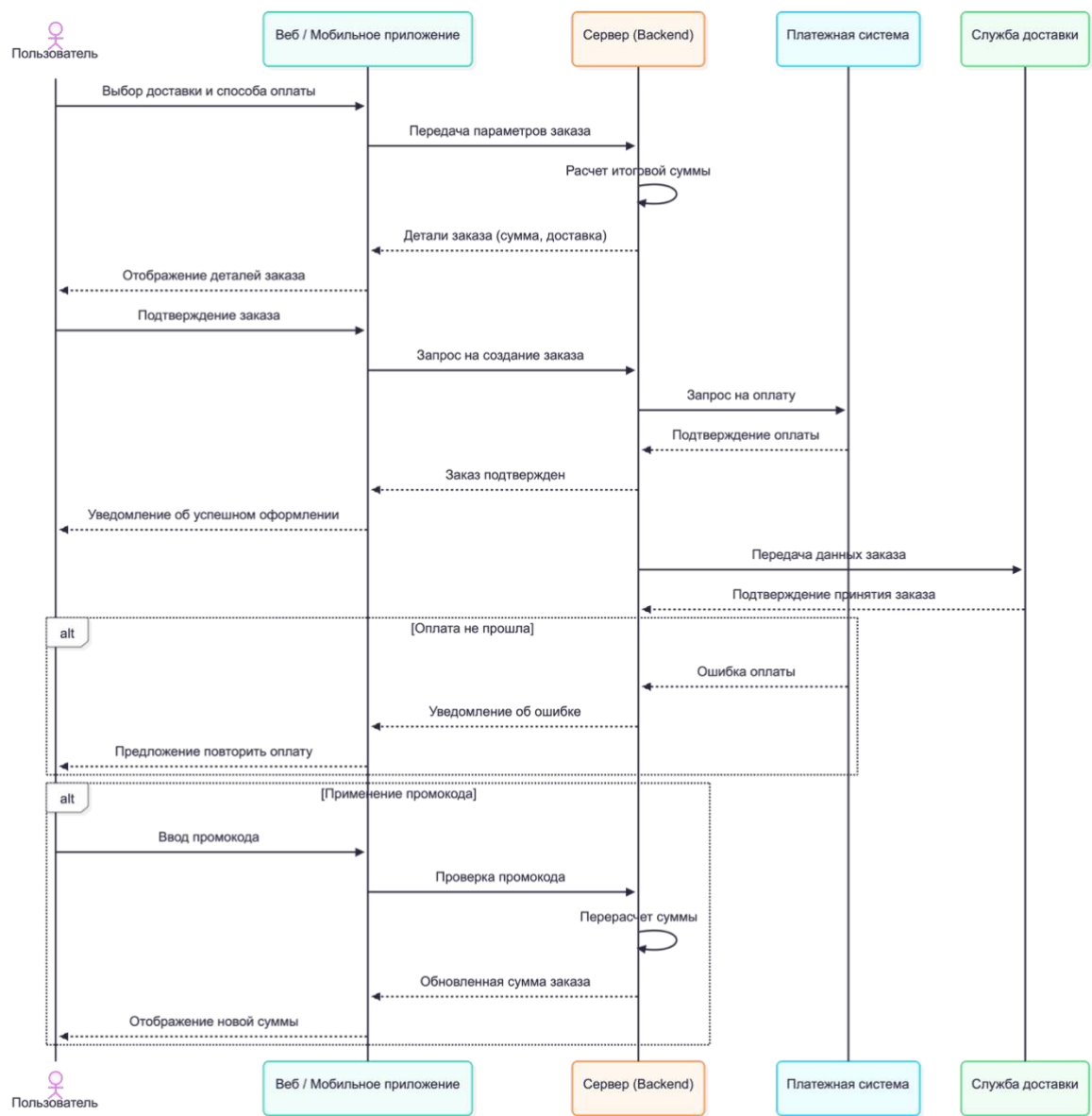
- Заказ товаров возможен только в пределах регионов, для которых платформа поддерживает доставку через собственные или партнёрские логистические каналы.
- Для оформления заказов и доступа к пользовательским функциям требуется авторизация в системе **AliExpress Russia**.
- Параметры одного заказа могут быть ограничены условиями перевозки, включая допустимый вес, размеры отправления и правила конкретной службы доставки.
- Продажа товаров, подпадающих под возрастные ограничения, допускается исключительно при подтверждении возраста получателя в момент получения заказа.

Use-case диаграмма



Sequence-диаграмма для прецедента

Прецедент: Оформление заказа



Сценарии использования

Прецедент: Поиск товаров на сайте
ID: 1
Краткое описание: Пользователь хочет найти товар, используя строку поиска и фильтры.
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 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: Фильтрация товаров по категориям и цене
ID: 2
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: <ol style="list-style-type: none">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: Добавление товара в корзину
ID: 3
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: Добавление нового адреса доставки
ID: 4
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 показывает ближайшие доступные пункты выдачи и постаматы.
4. Пользователь выбирает пункт выдачи или постамат.
5. Система подтверждает сохранение адреса.
Альтернативный поток:
1. Если адрес введен некорректно, система сообщает об ошибке.

Прецедент: Оформление заказа
ID: 5
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: Модерация отзывов
ID: 6
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:
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:
1. Если отзыв удален, пользователь получает уведомление.

Прецедент: Связь с продавцом
ID: 7
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.