**1.2. Техническое задание на разработку системы**

**1.2.1. Описание предметной области**

Интернет-магазин вещей предназначен для продажи различных товаров (одежда, аксессуары, обувь и т.д.) через веб-приложение. Система должна поддерживать взаимодействие с покупателями и администраторами. Покупатели могут искать и просматривать товары, добавлять их в корзину и оформлять заказы. Администраторы управляют ассортиментом, пользователями и анализируют данные о продажах. Внешние системы, такие как платежные шлюзы, складская система и логистика, интегрируются для выполнения определенных функций, не подлежащих проектированию в рамках этого задания.

**1.2.2. Описание требований к архитектуре решения**

Архитектура системы должна быть построена на основе трехуровневой клиент-серверной архитектуры:

1. **Клиентский уровень**:
   * Клиентское приложение покупателя (веб-приложение).
   * Клиентское приложение администратора (веб-приложение).
2. **Серверный уровень**:
   * API-сервер, обрабатывающий запросы от клиентских приложений.
   * Подсистема аутентификации и авторизации.
   * Подсистема разграничения прав доступа.
   * Подсистема логирования.
   * Подсистема мониторинга.
3. **Уровень хранения данных**:
   * Реляционная база данных для хранения основной информации о пользователях, товарах, заказах и т.д.
   * Кеширующая система для увеличения производительности.
   * Хранилище логов.

**Технологии и инструменты:**

* Frontend: React.js (покупательское приложение), Angular или Vue.js (административное приложение).
* Backend: Node.js с использованием Express.js или аналогичных фреймворков.
* База данных: PostgreSQL или MySQL.
* Кеширование: Redis.
* Журналирование: ELK Stack (Elasticsearch, Logstash, Kibana).
* Мониторинг: Prometheus, Grafana.

**1.2.4. Описание функциональных требований**

**Функциональные требования для покупательского приложения:**

1. **Регистрация и авторизация**:
   * Возможность регистрации новых пользователей.
   * Возможность авторизации зарегистрированных пользователей.
   * Восстановление пароля.
2. **Просмотр и поиск товаров**:
   * Категории товаров.
   * Фильтры и сортировка товаров.
   * Страница с детальной информацией о товаре.
3. **Корзина**:
   * Добавление и удаление товаров.
   * Изменение количества товаров.
   * Просмотр общей суммы заказа.
4. **Оформление заказа**:
   * Ввод данных для доставки.
   * Выбор способа оплаты.
   * Подтверждение заказа.
5. **Личный кабинет**:
   * Просмотр истории заказов.
   * Редактирование персональных данных.
   * Управление настройками аккаунта.

**Функциональные требования для административного приложения:**

1. **Управление товарами**:
   * Добавление нового товара.
   * Редактирование информации о товаре.
   * Удаление товара.
   * Управление категориями товаров.
2. **Управление пользователями**:
   * Просмотр списка пользователей.
   * Блокировка и разблокировка пользователей.
   * Изменение ролей пользователей.
3. **Отчеты и аналитика**:
   * Просмотр отчетов о продажах.
   * Анализ активности пользователей.
4. **Настройки системы**:
   * Управление ролями и правами доступа.
   * Настройка параметров системы.
5. **Мониторинг и журналирование**:
   * Просмотр системных логов.
   * Отслеживание состояния компонентов системы.

**1.2.5. Описание нефункциональных требований**

1. **Производительность**:
   * Система должна обрабатывать до 1000 одновременных пользователей без снижения скорости ответа.
   * Время отклика на основные действия пользователя (поиск товаров, добавление в корзину) не должно превышать 2 секунд.
2. **Безопасность**:
   * Все персональные данные пользователей должны быть зашифрованы.
   * Аутентификация и авторизация должны быть защищены от атак типа SQL-инъекций, XSS и CSRF.
   * Система должна поддерживать двухфакторную аутентификацию.
3. **Надежность**:
   * Система должна обеспечивать доступность не менее 99,9% времени в год.
   * Должна быть предусмотрена система резервного копирования данных с возможностью восстановления в случае сбоя.
4. **Масштабируемость**:
   * Система должна поддерживать возможность горизонтального и вертикального масштабирования для обеспечения роста нагрузки.
5. **Удобство использования**:
   * Интерфейсы должны быть интуитивно понятными и удобными для пользователя.
   * Все основные функции должны быть доступны в не более чем в трех кликах от главной страницы.
6. **Поддерживаемость**:
   * Кодовая база должна быть документирована и структурирована для упрощения сопровождения.
   * Должны быть предусмотрены механизмы логирования и мониторинга для выявления и устранения проблем.
7. **Совместимость**:
   * Веб-приложение должно корректно работать в основных современных браузерах (Chrome, Firefox, Safari, Edge).
   * Приложение должно быть адаптировано для мобильных устройств.