**ЗАПАДНЫЙ ФИЛИАЛ РОССИЙСКОЙ АКАДЕМИИ НАРОДНОГО ХОЗЯЙСТВА И ГОСУДАРСТВЕННОЙ СЛУЖБЫ**

**Информационные системы и программирование**

КУРСОВАЯ РАБОТА

по дисциплине: «Разработка программных модулей»

на тему: «Разработка программного модуля Интернет-магазин»

Выполнил:

студент 3 курса, гр. 22ИСП3

Сакович Р. А.

Руководитель:

Мартин А. Э.

Калининград, 2025 г.

**Оглавление**

1.1. Понятие и назначение интернет-магазина 3

Основные функции интернет-магазина 3

Разновидности интернет-магазинов 3

Заключение 4

1.2. Архитектура интернет-магазина 5

Основные компоненты архитектуры интернет-магазина 5

Принципы построения архитектуры 6

Заключение 6

1.3. Используемые технологии 7

Языки программирования 7

Фронтенд (Frontend) 7

Бэкенд (Backend) 7

База данных (Database) 7

Системы контроля версий 7

Внешние API и сервисы 8

1.4. Преимущества интернет-магазина 9

Ограничения и вызовы интернет-магазинов 9

1.5. Безопасность интернет-магазина 10

Основные угрозы безопасности 10

Методы защиты интернет-магазина 10

1.6. Заключение 11

1.7. Источники 11

### Введение

В современном мире цифровые технологии играют ключевую роль в развитии торговли и бизнеса. Интернет-магазины стали неотъемлемой частью электронной коммерции, предоставляя пользователям удобный способ приобретения товаров и услуг в режиме онлайн. Их популярность обусловлена удобством, доступностью и широким функционалом, позволяющим автоматизировать процессы продаж, управления заказами и взаимодействия с клиентами.

Разработка программного модуля интернет-магазина требует комплексного подхода, включающего выбор архитектуры, технологий и инструментов, обеспечивающих надежность, безопасность и удобство использования. В данной работе будут рассмотрены основные теоретические аспекты создания интернет-магазина, включая его назначение, архитектуру, используемые технологии, преимущества и вопросы безопасности.

Ключевыми технологиями, применяемыми при разработке современных интернет-магазинов, являются JavaScript, Node.js, React, базы данных MySQL и MongoDB, а также API для интеграции платежных и логистических сервисов. Все это позволяет создавать высокопроизводительные, масштабируемые и безопасные веб-приложения, соответствующие требованиям пользователей и бизнеса.

В данной курсовой работе будет проведено исследование теоретических основ разработки интернет-магазина, что позволит получить общее представление о процессе создания подобных веб-приложений и ключевых аспектах их функционирования.

### ГЛАВА 1. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ РАЗРАБОТКИ ПРОГРАММНОГО МОДУЛЯ ИНТЕРНЕТ-МАГАЗИНА

## 1.1. Понятие и назначение интернет-магазина

Интернет-магазин – это веб-приложение, предназначенное для организации процесса покупки и продажи товаров или услуг через интернет. Он позволяет пользователям просматривать каталог продукции, оформлять заказы, осуществлять оплату и взаимодействовать с продавцом. Интернет-магазины являются важной частью электронной коммерции и широко применяются в различных сферах бизнеса, таких как розничная торговля, продажа цифровых товаров, бронирование услуг и многое другое.

****Основные функции интернет-магазина****

Современный интернет-магазин должен обладать рядом ключевых функций, обеспечивающих удобство для пользователей и автоматизацию бизнес-процессов:

* **Каталог товаров с описанием и ценами** – представляет собой структурированный список товаров с детальными характеристиками, изображениями и актуальными ценами. Это основа интернет-магазина, позволяющая покупателям ознакомиться с ассортиментом.
* **Фильтрация и поиск товаров** – удобный поиск по названию, категориям, производителям, ценовому диапазону и другим параметрам. Это снижает время, затрачиваемое на поиск нужного товара, и повышает удобство пользования.
* **Оформление заказа и корзина покупок** – система, позволяющая клиенту выбрать несколько товаров, отложить их в корзину, а затем оформить покупку. Корзина сохраняет информацию о выбранных товарах, пока пользователь не завершит оформление заказа.
* **Способы оплаты** – интеграция с платежными системами, такими как банковские карты (Mir), электронные кошельки (ЮMoney), криптовалютные платежи, а также сервисы «Купи сейчас – заплати потом» (например, Tinkoff Pay).
* **Управление заказами и их статусами** – позволяет клиенту отслеживать статус своего заказа (обработан, передан в доставку, доставлен), а продавцу – управлять процессами выполнения заказов.
* **Регистрация и авторизация пользователей** – система управления учетными записями, позволяющая клиентам сохранять историю заказов, использовать персональные скидки и бонусы, а продавцам – сегментировать аудиторию.
* **Интеграция с логистическими сервисами** – возможность выбора способов доставки (курьерские службы, самовывоз, почта) с расчетом стоимости и времени доставки в зависимости от региона.

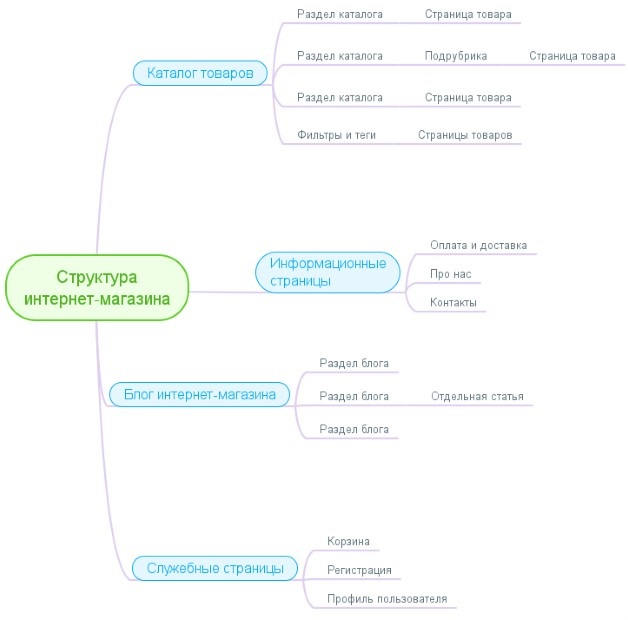
****Разновидности интернет-магазинов****

Существуют разные типы интернет-магазинов, зависящие от модели бизнеса:

* **B2C (Business-to-Consumer)** – классические интернет-магазины, продающие товары напрямую конечным потребителям (например, Ozon, Wildberries).
* **B2B (Business-to-Business)** – платформы для оптовой торговли между компаниями (например, Alibaba).
* **C2C (Consumer-to-Consumer)** – площадки для частных лиц, продающих товары друг другу (Avito, eBay).
* **D2C (Direct-to-Consumer)** – магазины, работающие без посредников, когда производитель сам продает свои товары (например, официальные магазины Apple, Nike).

****Заключение****

Интернет-магазин – это не просто сайт, а полноценная платформа для ведения бизнеса в цифровой среде. Современные технологии позволяют создать удобный и безопасный механизм для взаимодействия между покупателем и продавцом, обеспечивая высокую скорость работы, автоматизацию процессов и удобство использования.



## **1.2. Архитектура интернет-магазина**

Разработка интернет-магазина основана на клиент-серверной архитектуре, которая обеспечивает разделение логики между различными компонентами системы. Такая архитектура позволяет масштабировать проект, повышать его надежность и гибкость, а также улучшать пользовательский опыт.

****Основные компоненты архитектуры интернет-магазина****

1. **Клиентская часть (Frontend)**  
   Клиентская часть отвечает за взаимодействие с пользователем и отображение информации. Она разрабатывается с использованием веб-технологий, таких как:
   * **HTML** – структура веб-страниц.
   * **CSS** – стилизация интерфейса (оформление, анимация, адаптивность).
   * **JavaScript** – обработка событий, динамическое обновление содержимого.
   * **Фреймворки и библиотеки** (React, Vue.js, Angular) – позволяют создавать удобные и интерактивные интерфейсы.

Клиентская часть загружает данные с сервера, обрабатывает пользовательские действия (добавление товаров в корзину, оформление заказа) и отправляет запросы на серверную часть для выполнения необходимых операций.

1. **Серверная часть (Backend)**  
   Серверная часть обрабатывает запросы от клиентов, управляет бизнес-логикой, взаимодействует с базой данных и внешними сервисами. Чаще всего backend разрабатывается на следующих технологиях:
   * **Node.js** – высокопроизводительная серверная среда, позволяющая обрабатывать множество запросов одновременно.
   * **Express.js** – фреймворк для создания API и маршрутизации запросов.
   * **NestJS** – более структурированный вариант для построения сложных приложений.

Основные функции серверной части:

* + Обработка запросов от клиентской части.
  + Управление пользователями (регистрация, авторизация).
  + Работа с заказами и товарами (CRUD-операции: создание, чтение, обновление, удаление).
  + Взаимодействие с базой данных.
  + Интеграция с платежными и логистическими системами.

1. **База данных (Database)**  
   База данных хранит информацию о пользователях, товарах, заказах, платежах и других элементах интернет-магазина. Выбор базы данных зависит от требований к масштабируемости и надежности:
   * **MySQL / PostgreSQL** – реляционные базы данных, удобные для структурированных данных (каталог товаров, заказы, пользователи).
   * **MongoDB** – документно-ориентированная база данных, позволяющая хранить неструктурированные данные и работать с масштабируемыми приложениями.
2. **Внешние сервисы и API**  
   Для расширения функционала интернет-магазина часто используются сторонние сервисы:
   * **Платежные шлюзы** (ЮKassa, Сбербанк, PayPal, Stripe) – для обработки онлайн-платежей.
   * **Логистические сервисы** (CDEK, Почта России) – для расчета стоимости и оформления доставки.
   * **Системы аналитики** (Google Analytics, Яндекс.Метрика) – для анализа поведения пользователей.

****Принципы построения архитектуры****

1. **Модульность** – разделение системы на независимые компоненты (каталог товаров, управление заказами, авторизация).
2. **Масштабируемость** – возможность увеличения нагрузки без потери производительности.
3. **Безопасность** – защита данных пользователей и транзакций.
4. **Оптимизация запросов** – использование кеширования и эффективных алгоритмов обработки данных.

****Заключение****

Клиент-серверная архитектура интернет-магазина позволяет разделить ответственность между frontend и backend, обеспечивая удобство разработки и поддержку высокой нагрузки. Грамотное проектирование структуры системы делает интернет-магазин надежным, удобным и масштабируемым.

## **1.3. Используемые технологии**

Разработка интернет-магазина требует применения различных технологий для создания клиентской и серверной частей, управления данными и интеграции с внешними сервисами. Выбор инструментов зависит от требований к производительности, удобству разработки и масштабируемости системы.

****Языки программирования****

Для создания интернет-магазина используются:

* **JavaScript** – основной язык для разработки как frontend, так и backend-части.

****Фронтенд (Frontend)****

Frontend – это пользовательский интерфейс интернет-магазина, с которым взаимодействует клиент. Основные технологии для его разработки:

* **React** – библиотека для создания динамичных пользовательских интерфейсов. Позволяет разрабатывать компонентную структуру и использовать state management
* **TailwindCSS** – утилитарный CSS-фреймворк для стилизации.

****База данных (Database)****

Для хранения информации об интернет-магазине применяются:

* **json-server** – Фейковый API, для работы с локальными JSON-файлами как с БД

****Системы контроля версий****

Для управления кодом и командной работы используется **Git**. Основные платформы:

* **GitHub** – облачное хранилище репозиториев с возможностью CI/CD.

****Заключение****

Использование современных технологий, таких как Node.js, React и базы данных PostgreSQL или MongoDB, позволяет создать быстрый, надежный и масштабируемый интернет-магазин. Грамотный выбор инструментов упрощает разработку и поддержку проекта.

## **1.4. Преимущества интернет-магазина**

Интернет-магазины имеют ряд значительных преимуществ по сравнению с традиционной розничной торговлей. Их популярность стремительно растет благодаря удобству, доступности и возможностям автоматизации процессов.

****Основные преимущества интернет-магазина****

1. **Круглосуточная доступность**
   * Интернет-магазины работают 24/7 без выходных и праздников, что позволяет клиентам совершать покупки в любое удобное время.
2. **Глобальный охват**
   * Товары и услуги могут продаваться не только в одном регионе, но и по всей стране или даже миру, что значительно расширяет аудиторию.
3. **Снижение затрат**
   * В отличие от физического магазина, интернет-магазин не требует аренды помещений, оплаты коммунальных услуг и большого штата сотрудников.
4. **Автоматизация процессов**
   * Онлайн-магазины используют CRM-системы, автоматизированные платежи, электронные чеки и интеграции с логистическими сервисами, что сокращает ручной труд.
5. **Гибкость и масштабируемость**
   * Легкость изменения ассортимента, цен и скидок без необходимости перепечатывать каталоги или менять вывески.
   * Возможность быстрого масштабирования бизнеса путем расширения функционала и рекламы.
6. **Персонализация и маркетинг**
   * Современные интернет-магазины используют **Big Data**, аналитику и AI для персонализации рекомендаций, рассылки предложений и динамического ценообразования.
7. **Удобные способы оплаты и доставки**
   * Онлайн-магазины позволяют оплачивать товары различными способами (банковские карты, электронные кошельки, криптовалюты).
   * Гибкие варианты доставки (курьерские службы, пункты самовывоза, экспресс-доставка).

****Ограничения и вызовы интернет-магазинов****

Несмотря на преимущества, у интернет-магазинов есть и некоторые сложности:

* **Необходимость надежной логистики** – доставка товаров требует четкой организации.
* **Конкуренция** – рынок насыщен, и для успешного продвижения требуется качественный маркетинг.
* **Зависимость от технологий** – сбои серверов, уязвимости в безопасности и нехватка технической поддержки могут приводить к убыткам.

****Заключение****

Интернет-магазины позволяют бизнесу работать эффективнее, снижая затраты и расширяя аудиторию. Современные технологии автоматизации, персонализации и аналитики делают их удобным инструментом для электронной коммерции.

## **1.5. Безопасность интернет-магазина**

Безопасность интернет-магазина – один из важнейших аспектов разработки, поскольку веб-приложение обрабатывает личные данные пользователей, платежную информацию и коммерческую переписку. Нарушение безопасности может привести к финансовым потерям, утечке данных и ухудшению репутации бизнеса.

****Основные угрозы безопасности****

1. **Несанкционированный доступ к данным** – киберпреступники могут попытаться получить доступ к учетным записям пользователей или базе данных магазина.
2. **SQL-инъекции** – атаки, при которых злоумышленник внедряет вредоносные SQL-запросы для изменения или кражи данных.
3. **XSS (Cross-Site Scripting)** – внедрение вредоносного кода на веб-страницы для кражи пользовательских данных или изменения содержимого.
4. **CSRF (Cross-Site Request Forgery)** – атаки, при которых злоумышленник отправляет поддельные запросы от имени пользователя.
5. **Перехват данных** – если соединение не защищено (отсутствует HTTPS), хакеры могут прослушивать и изменять передаваемые данные.
6. **DDoS-атаки** – атаки на сервер с целью перегрузки и вывода магазина из строя.

****Методы защиты интернет-магазина****

1. **Шифрование данных**
   * Использование **SSL/TLS-сертификатов** для обеспечения защищенного соединения (HTTPS).
   * Хранение паролей в зашифрованном виде с помощью алгоритмов **bcrypt** или **argon2**.
2. **Защита от SQL-инъекций**
   * Использование **подготовленных запросов** (prepared statements) и ORM (например, Sequelize, TypeORM).
3. **Защита от XSS-атак**
   * Очистка входных данных с помощью библиотек (DOMPurify).
   * Использование заголовков безопасности (Content-Security-Policy).
4. **Защита от CSRF**
   * Использование **CSRF-токенов** для проверки подлинности запросов.
5. **Аутентификация и авторизация**
   * Реализация **JWT (JSON Web Token)** для безопасного хранения сеансов пользователей.
   * Ограничение числа попыток входа (Brute-force защита).
6. **Регулярные резервные копии**
   * Автоматизированное создание резервных копий базы данных для защиты от сбоев.
7. **Защита от DDoS-атак**
   * Использование **CDN (Cloudflare, AWS Shield)** для фильтрации трафика.
   * Ограничение частоты запросов (rate limiting).

****Заключение****

Безопасность интернет-магазина требует комплексного подхода: защита соединения, обработка пользовательских данных, предотвращение атак и регулярные обновления системы. Использование современных технологий и лучших практик безопасности снижает риски киберугроз и обеспечивает надежную работу магазина.

## 1.6. Заключение

Разработка интернет-магазина требует комплексного подхода, включающего выбор подходящей архитектуры, технологий, которые обеспечат надежность, скорость, стабильность и обеспечат высокий уровень безопасности. Современные инструменты, такие как Node.js и React, позволяют создать высокопроизводительное и удобное веб-приложение, отвечающее требованиям пользователей и бизнеса.

## 1.7. Источники

1. Веселов А.В. "Разработка веб-приложений на JavaScript" – СПб.: Питер, 2023.
2. Документация React: [https://react.dev](https://react.dev/)
3. "JavaScript: The Good Parts", Douglas Crockford
4. "You Don’t Know JS", Kyle Simpson
5. "React Up and Running", Stoyan Stefanov
6. OWASP
7. Cloudflare Blog