

Сюэ Пэн, Магистрант,
Амурский государственный университет

ПРОЕКТИРОВАНИЕ АРХИТЕКТУРЫ И ФУНКЦИОНАЛЬНОСТИ ИНТЕРНЕТ-САЙТА АПТЕКИ: ОТ КОНЦЕПЦИИ ДО ЛОГИЧЕСКОЙ МОДЕЛИ

Аннотация. В статье рассматривается комплексный процесс проектирования интернет-сайта аптеки, учитывающий специфику фармацевтической отрасли. Основное внимание уделяется формированию трехуровневой клиент-серверной архитектуры, методологии проектирования базы данных от концептуальной ER-модели до детализированной логической схемы в MySQL, а также описанию ключевых функциональных модулей, реализующих бизнес-процессы платформы. Работа синтезирует общие принципы веб-разработки и проектирования баз данных со строгими требованиями к безопасности и нормативному регулированию в сфере онлайн-продаж лекарственных средств.

Abstract. The article examines the comprehensive design process for an online pharmacy website, taking into account the specifics of the pharmaceutical industry. The focus is on the formation of a three-tier client-server architecture, the database design methodology from a conceptual ER model to a detailed logical schema in MySQL, as well as the description of key functional modules that implement the platform's business processes. The work synthesizes general principles of web development and database design with strict requirements for security and regulatory compliance in the online sale of medicines.

Ключевые слова: Интернет-сайт аптеки, проектирование архитектуры, база данных, ER-модель, функциональные модули, фармацевтическая электронная коммерция.

Keywords: Online pharmacy website, architecture design, database, ER model, functional modules, pharmaceutical e-commerce.

Стремительная цифровизация экономики и повседневной жизни коренным образом трансформирует традиционные отрасли, включая сферу здравоохранения и фармации. Интернет-продажи, ставшие «наиболее революционной формой бизнеса в эпоху цифровой экономики» [4], создали предпосылки для появления специализированных платформ медицинской электронной коммерции. Эти платформы, сочетая в себе удобство онлайн-шопинга и строгие требования фармацевтического рынка, призваны решать важные социальные задачи: повышать доступность лекарственных средств, оптимизировать логистические цепочки и формировать новые модели взаимодействия между потребителем, аптекой и производителем.

Однако проектирование и реализация веб-сайта аптеки – задача, выходящая за рамки стандартной разработки интернет-магазина. Она требует учета целого комплекса специфических факторов: жёсткого нормативно-правового регулирования оборота лекарственных препаратов, повышенных требований к безопасности и конфиденциальности медицинских данных, а также необходимости интеграции сложной бизнес-логики, связанной с управлением рецептами, остатками и сроками годности [4]. Успех такого проекта в значительной степени определяется качеством его фундаментальных этапов – системного и архитектурного проектирования.

Архитектурный фундамент: трехуровневая модель и ролевая структура. Проектирование архитектуры является отправной точкой, определяющей производительность, надежность и масштабируемость будущей системы. Ядром технического решения является выбор трехуровневой клиент-серверной архитектуры. Данный подход



обеспечивает четкое разделение ответственности между компонентами системы, что упрощает разработку, тестирование, развертывание и дальнейшее сопровождение [1, 3].

Уровень представления (frontend), отвечающий за взаимодействие с конечным пользователем, может быть реализован с использованием современных фреймворков, таких как Vue.js или React. Эти технологии позволяют создавать динамичные, отзывчивые и интуитивно понятные интерфейсы. Вся бизнес-логика – обработка заказов, управление товарами, аутентификация пользователей – сосредоточена на серверном уровне (backend). Для его реализации часто выбирают мощные и хорошо зарекомендовавшие себя инструменты, например, язык программирования PHP в сочетании с фреймворком Laravel. Уровень данных, представленный системой управления базами данных (СУБД) MySQL, отвечает за целостное, структурированное и безопасное хранение всей информации [1, 3].

Архитектура системы напрямую отражает ее ролевую модель, которая включает три основных типа участников: обычных пользователей (покупателей), аптеки-партнеры и администраторов платформы. Каждая роль обладает строго определенным набором прав и доступом к специфическим функциям. Такое разделение обеспечивает безопасность данных и эффективность бизнес-процессов.

Концептуальное проектирование базы данных: модель «сущность-связь»
Прежде чем приступить к созданию физических таблиц в СУБД, необходимо разработать концептуальную модель данных, которая абстрактно описывает ключевые информационные объекты (сущности) предметной области и отношения между ними. Наиболее наглядным инструментом для этого служит ER-диаграмма (Entity-Relationship Diagram) [2].

В контексте сайта аптеки центральными сущностями являются: Пользователь (yonghu), Аптека (yaodian), Лекарственный препарат (yaopinxi), Заказ (orders), Корзина покупок (cart). Вспомогательные сущности, такие как Адрес доставки (address), Сообщения (messages) и Отзывы (discussyaopinxi), обогащают функционал платформы. Диаграмма наглядно показывает связи «один-ко-многим» (например, один пользователь может оформить множество заказов) и «многие-ко-многим» (один препарат может быть в наличии во многих аптеках, и одна аптека предлагает много препаратов). Этот этап позволяет выявить и устранить смысловые противоречия в данных до начала программирования [2].

Логическое проектирование: от концепции к реляционной схеме

Логическое проектирование – это этап трансформации концептуальной модели в конкретную схему, готовую для реализации в выбранной реляционной СУБД (в данном случае – MySQL). Каждая сущность превращается в таблицу, а ее атрибуты – в столбцы с четко определенными типами данных (INT, VARCHAR, DECIMAL и т.д.), размерами и ограничениями (NOT NULL, UNIQUE) [2].

Ключевым аспектом является нормализация – процесс устранения избыточности данных и аномалий при их обновлении. Например, в таблице medicines (лекарственные препараты) сохраняется только уникальный идентификатор аптеки (pharmacy_id как внешний ключ), который ссылается на основную таблицу pharmacies. Это обеспечивает целостность данных. Для каждой таблицы определяется первичный ключ (часто это автоинкрементное поле id). Для ускорения выполнения частых запросов проектируются индексы. Особое внимание уделяется проектированию таблиц orders (заказы) и carts (корзины), где в заказе часто хранится «снимок» данных о товаре, защищающий историю транзакций.

Функциональные модули: воплощение бизнес-логики в интерфейсе

Спроектированная архитектура и модель данных реализуются через набор взаимосвязанных функциональных модулей.

Модуль управления пользователями и аутентификации реализует регистрацию, вход, восстановление доступа и управление ролями.



Модуль каталога и поиска товаров обеспечивает просмотр, структурирование по категориям (`medicine_categories`), и мощный поиск с фильтрацией.

Модуль корзины и оформления заказа обеспечивает плавный пользовательский путь от выбора к покупке, интегрирован с данными пользователя и остатками.

Модуль управления контентом предоставляет аптекам-партнерам инструменты для управления ассортиментом (`medicines`), а администраторам – для управления платформой.

Модуль обработки заказов и аналитики позволяет аптекам управлять статусами заказов и логистикой (`logistics`), а администраторам – анализировать агрегированную статистику (`sales_statistics`).

Список литературы:

1. Дунаев, В. В. (2020). Веб-программирование на PHP и MySQL. Москва: БХВ-Петербург.
2. Смилянский, М. Л. (2019). Проектирование баз данных. От концептуальной модели к реализации в СУБД. Санкт-Петербург: Питер.
3. Кузнецов, М. В., & Симдянов, И. В. (2021). Интернет-магазин с нуля. PHP, JavaScript, CSS, HTML, MySQL. Санкт-Петербург: БХВ-Петербург.
4. Полякова, Е. А., & Козлов, Д. Н. (2018). Особенности проектирования информационных систем в фармации и здравоохранении. Фармация, 67(5), 45-51.
5. Ахметов, Р. Р. (2022). Архитектура высоконагруженных веб-сервисов. Москва: ДМК Пресс

