

Сервис онлайн-магазина свечей

Описание проекта

Это онлайн-платформа для покупки свечей ручной работы. Пользователи могут просматривать каталог, фильтровать товары по аромату, цвету, форме и цене, добавлять понравившиеся свечи в корзину и оформлять заказ с оплатой через интернет. Производители имеют возможность добавлять свои изделия, редактировать ассортимент и получать отзывы от покупателей.

Фронтенд

Интерфейс сайта реализован на React с использованием Zustand для управления состоянием приложения. Навигация осуществляется с помощью React Router. Для взаимодействия с сервером применяется стандартный fetch API. Это одностраничное приложение, которое позволяет быстро загружать и отображать данные без перезагрузки страницы. Формы на сайте проверяются с помощью встроенной валидации браузера, что гарантирует корректность введённых пользователем данных, таких как email и адрес доставки.

Бэкенд

Серверная часть написана на Node.js с использованием Express. Сервер принимает REST-запросы, обрабатывает их, обеспечивает аутентификацию пользователей через JWT-токены, выполняет операции с базой данных по товарам, заказам и пользователям. Для обработки платежей интегрирована платёжная система Stripe, которая позволяет принимать безопасные онлайн-платежи. После подтверждения оплаты сервер создаёт заказ, сохраняет его в базе и отправляет пользователю подтверждение по электронной почте.

База данных

Для хранения всех данных используется PostgreSQL. В базе хранятся сведения о товарах, пользователях, заказах и отзывах. Таблицы связаны между собой, что позволяет эффективно организовать поиск и обновление данных. Используются индексы для повышения скорости выборки, а также механизмы целостности данных, такие как ограничения и связи.

Уведомления

Уведомления отправляются пользователям по электронной почте с помощью nodemailer и SMTP-сервера. Кроме того, реализован WebSocket-сервер, который позволяет отправлять уведомления в режиме реального времени, например о подтверждении заказа или изменении его статуса. Это обеспечивает быстрое информирование пользователя без необходимости обновлять страницу.

Сервис каталога

Отвечает за хранение и управление данными о свечах, включая информацию о аромате, цвете, форме и цене. Реализован как отдельный модуль на Node.js с использованием Express. Взаимодействует с базой данных PostgreSQL для получения и обновления информации о товарах, поддерживает поиск и фильтрацию по заданным параметрам.

Сервис корзины

Хранит текущее состояние корзины пользователя в памяти с помощью Redis, что обеспечивает быструю обработку запросов и кэширование данных. Реализован на Node.js с Express и поддерживает синхронизацию содержимого

корзины между клиентом и сервером, позволяя автоматически обновлять данные при изменениях.

Сервис заказов

Отвечает за создание, обновление и хранение заказов, управление их статусами и историей. Реализован на Node.js с Express, хранит данные в PostgreSQL. Интегрируется с платёжными сервисами для проверки оплаты и передаёт данные в другие сервисы для дальнейшей обработки.

Сервис пользователей

Обрабатывает регистрацию и аутентификацию пользователей, управляет профилями и сессиями с помощью JWT. Реализован на Node.js с Express и работает с PostgreSQL для хранения данных о пользователях, настройках и адресах доставки.

Сервис оплаты

Отвечает за приём и проверку платежей через платёжную систему Stripe. Реализован на Node.js с Express. Обрабатывает транзакции, проверяет их статус и сообщает сервису заказов о результатах для дальнейшей обработки.

Сервис доставки

Интегрируется с внешним API службы доставки СДЭК для расчёта стоимости и сроков доставки, а также для генерации трекинг-номеров. Реализован на Node.js и обменивается данными с другими сервисами для отображения актуальной информации пользователю.

Брокер сообщений

Используется RabbitMQ для организации асинхронного обмена событиями между сервисами. Обеспечивает передачу сообщений, например, о подтверждении оплаты, которые запускают отправку уведомлений или обновление статусов.

