Распишите нефункциональные требования для разработки сайта онлайн магазина

1. У магазина должен быть свой сайт, оформленный в цветах брендбука.
2. Пользователь может видеть товары, сгруппированные по разделам (бакалея, молочные продукты, мясные продукты и прочее).
3. Пользователь может видеть весь перечень товаров, участвующих в акции на текущий период. Пользователь может добавлять товары в корзину.
4. Пользователь может писать отзывы на товары, которые он ранее приобретал.
5. Пользователь может видеть сумму стоимости товаров, которые сложил в корзину с учётом скидки.
6. Пользователь может ввести номер своей дисконтной карты при оформлении заказа и стоимость заказа будет автоматически пересчитана.
7. Пользователь может указать адрес доставки или выбрать самовывоз при выборе способа доставки.
8. Пользователь видит либо режим работы пункта выдачи, либо диапазоны времени, когда может приехать курьер заказом.

**Нефункциональные требования для разработки сайта онлайн магазина**

1. **Производительность**

Без подсказок:

* Время загрузки страницы не более 2 секунды
* Время загрузки страницы в пиковые часы с 19:00 по 21:00 в будние дни и с 11:00 по 21:00 в выходные не более 5 секунды
* Время загрузки страницы Корзина после изменения не более 3 секунды
* Количество одновременно работающих пользователей не менее 1000
* Размер поля Отзыв не более 250 символов

Со помощью интернет:

* Главная страница должна загружаться не более чем за 2 секунды при скорости интернета 10 Мбит/с.
* Страницы категорий товаров должны загружаться не более чем за 3 секунды.
* Страницы товаров должны загружаться не более чем за 2,5 секунды
* Время отклика сервера на запросы (добавление товара в корзину, обновление страницы) не должно превышать 500 мс при пиковой нагрузке.
* Время обработки запросов API (для интеграции с платежными системами или службами доставки) не должно превышать 1 секунды.
* Сайт должен выдерживать одновременную работу до 10 000 пользователей без снижения производительности.
* При увеличении нагрузки на 50% время отклика системы не должно увеличиваться более чем на 20%.
* Время обновления корзины при добавлении или удалении товара не должно превышать 1 секунды.
* Время пересчёта стоимости заказа с учётом скидки или дисконтной карты не должно превышать 1 секунды.
* Пропускная способность (запросов в секунду)
* скорости и пропускной способности каналов связи
* Время загрузки главной страницы сайта не должно превышать 2 секунд при скорости интернета 10 Мбит/с.
* Сайт должен корректно отображаться и функционировать при одновременном посещении до 1000 пользователей.
* Время отклика при поиске товаров по ключевым словам не должно превышать 1,5 секунд.
* Время отклика при фильтрации товаров по категориям, ценам или другим параметрам не должно превышать 2 секунд.
* Время загрузки отзывов на странице товара не должно превышать 1,5 секунд.
* Время обработки запроса на добавление отзыва не должно превышать 1 секунды.
* Время загрузки страницы с товарами, участвующими в акции, не должно превышать 3 секунд.
* Время обновления информации о скидках и акциях на сайте не должно превышать 1 секунды.
* Время загрузки информации о доступных способах доставки и времени доставки не должно превышать 2 секунд.
* Время обработки запроса на выбор способа доставки или указание адреса не должно превышать 1 секунды.
* Статические данные (изображения товаров, информация о категориях) должны кэшироваться для уменьшения времени загрузки страниц.
* Данные о товарах, участвующих в акциях, должны кэшироваться для быстрого доступа.
* База данных должна поддерживать до 10 терабайт данных.

*Это нормально так много расписывать (про обновление каждой странички?)*

1. **Безопасность**

Без подсказок:

* Обмен сообщениями по протоколу HTTPS
* Двухфакторная аутентификация пользователей с SMS
* Обязательная смена пароля пользователя каждые 6 месяцев
* Требование к паролю: не менее 12 символов, Наличие заглавной буквы, строчной буквы, цифры и специального символа

Со помощью интернет:

* Использование протокола HTTPS для шифрования данных, передаваемых между пользователем и сервером.
* Все пользовательские данные должны быть зашифрованы как при передаче, так и при хранении.
* Шифрование данных: AES-256
* Пароли пользователей должны храниться в хэшированном виде с использованием bcrypt.
* Ограничение количества попыток входа для предотвращения brute-force атак: Система должна блокировать учетную запись пользователя после 5 неудачных попыток ввода пароля в течение 5 минут. После блокировки пользователь должен получить уведомление и возможность восстановить доступ через email или SMS.
* Защита от SQL-инъекций: все запросы к базе данных должны использовать параметризованные запросы.
* Защита от XSS (межсайтового скриптинга): входные данные пользователя должны валидироваться и экранироваться.
* Защита от CSRF (межсайтовая подделка запроса): использование CSRF-токенов для всех форм и критических операций.
* Защита от DDoS-атак: использование Cloudflare (средство для предотвращения распределенных атак).
* Ежедневное резервное копирование данных с хранением копий в безопасном месте.
* Разграничение прав доступа для администраторов и пользователей (администраторы могут редактировать товары, а пользователи — только просматривать).
* Ведение журнала аудита для отслеживания действий администраторов и пользователей.
* Ограничение доступа к данным пользователей только для авторизованного персонала.
* Регулярное тестирование на уязвимости с использованием OWASP ZAP.
* *Использование токенов OAuth 2.0 для авторизации API. (все что про API лишнее т.к. не следует из условия задачи?)*
* *Ограничение частоты запросов к API для предотвращения злоупотреблений.*
* *Валидация всех входных данных, передаваемых через API*.
* Регулярные обновления: Обновления для CMS не реже 1 раза в месяц.Мониторинг безопасности системы в реальном времени с использованием SIEM-решений.
* Уведомление пользователей о подозрительных действиях (вход с нового устройства, изменение пароля).
* Предоставление пользователям возможности отслеживать активность своей учетной записи.
* Обучение пользователей распознаванию фишинговых атак (через информационные материалы на сайте).
* Использование DMARC, SPF и DKIM для защиты электронной почты от подделки.
* Соответствие требованиям GDPR (для пользователей из ЕС) для защиты персональных данных.
* Соответствие требованиям локальных законов о защите данных ФЗ-152 (для пользователей РФ).
* Регулярный аудит прав доступа: периодическая проверка и обновление прав доступа пользователей 1 раз в квартал.

1. **Удобство использования**

* Интуитивно понятный интерфейс
* Доступно для людей с ограниченными возможностями по зрению и слуху
* Поддержка опционально белорусского, китайского, английского языка
* Веб-сайт должен иметь четкую и последовательную навигационную структуру, позволяющую пользователям легко находить категории продуктов (продукты питания, молочные продукты, мясные продукты).
* Должна быть реализована навигационная цепочка Breadcrumbs, чтобы помочь пользователям понять их текущее местоположение в иерархии сайта.
* Кнопки «Добавить в корзину», «Написать отзыв» и «Оформить заказ», должны быть видны и легко различимы, чтобы направлять пользователей в процессе покупки.
* Должна быть доступна строка поиска с функциями автозаполнения и фильтрации, чтобы помочь пользователям быстро находить определенные продукты или категории.
* Товары должны отображаться в виде коллажной сетки с четкими изображениями, ценами и этикетками (например, «Распродажа» или «Скидка»).
* Рекламные товары должны быть выделены на видном месте на главной странице.
* Корзина должна быть легко доступна с любой страницы, и пользователи должны иметь возможность просматривать, изменять или удалять товары без лишних шагов.
* Общая стоимость, включая скидки, должна отображаться четко и динамически обновляться по мере внесения изменений.
* На веб-сайте должны быть четкие сообщения об ошибках и запросы на проверку (для недействительных номеров дисконтных карт или неполной информации о доставке), чтобы помочь пользователям легко исправить ошибки.
* Варианты доставки, часы работы пункта выдачи и доступные временные интервалы для курьерской доставки должны быть четко и заметно отображены во время оформления заказа.
* Раздел справки, FAQ (часто задаваемые вопросы) должны быть легкодоступны.
* Сайт должен соответствовать стандарту WCAG 2.1 уровня AA для обеспечения доступности для пользователей с ограниченными возможностями.
* Цветовая схема должна обеспечивать достаточный контраст для комфортного восприятия текста (минимальное соотношение контрастности 4.5:1).
* Новые пользователи должны иметь возможность завершить процесс регистрации менее чем за 3 минуты.
* Система должна иметь оценку удовлетворенности пользователей не менее 90%.

1. **Масштабируемость**

* Возможность увеличения количества пользователей путем увеличения мощности сервера (добавления новых серверов)
* Горизонтальное масштабирование (добавление большего количества серверов)
* Вертикальное масштабирование (увеличение мощности сервера)
* возможность переноса приложений на более мощные SMP-системы
* Использование балансировщиков нагрузки (NGINX) для распределения запросов между серверами.
* Использование Kubernetes для управления кластером серверов приложений.
* Использование Redis Cluster для распределенного хранения кэшированных данных.
* Использование асинхронных задач для обработки фоновых операций, таких как отправка email или обработка заказов.

1. **Надёжность**

* Система должна обеспечивать доступность сайта не менее 99,9% в год.
* Среднее время между отказами (MTBF, Mean Time Between Failures) для критических компонентов (сервер базы данных, сервер приложений) должен составлять не менее 10000 часов.
* MTBF для второстепенных компонентов (модуль отзывов, система рекомендаций) должен составлять не менее 5 000 часов.
* Среднее время восстановления (MTTR, Mean Time To Repair) для критических компонентов (сервер базы данных, сервер приложений) должен составлять не более 30 минут.
* MTTR для второстепенных компонентов (модуль отзывов, система рекомендаций) должен составлять не более 2 часов.
* Все транзакции должны регистрироваться для обеспечения возможности восстановления.
* В случае аварийного сбоя система должна восстанавливать работоспособность в течение 5 минут.
* Критические функции (просмотр товаров, добавление в корзину, оформление заказа) должны быть приоритетными при восстановлении.
* Резервное копирование данных (товары, заказы, отзывы, пользовательские данные) должно выполняться ежедневно.
* Резервные копии должны храниться не менее 30 дней.
* Восстановление данных из резервной копии должно занимать не более 1 часа.
* Система должна быть спроектирована так, чтобы отказ одного компонента (например, сервера базы данных) не приводил к полной недоступности сайта.
* Должна быть предусмотрена возможность автоматического переключения на резервные серверы в случае сбоя.

1. **Доступность (отказоустойчивость)**

* Сайт должен быть доступен 24/7.
* Максимально допустимое время простоя системы (не более 10 минут в сутки, не более 60 минут в месяц).
* Система должна обеспечивать доступность сайта не менее 99,9% в год.

1. **Техническое обслуживание**

* Модульная архитектура позволяет обновлять части системы независимо.
* Система должна поддерживать модульные обновления без простоев.
* Архитектура сайта должна позволять легко добавлять новые страницы и функциональные блоки без нарушения общего стиля и цветовой схемы.
* Система управления контентом (CMS) должна поддерживать возможность быстрого обновления цветовой палитры в случае изменения брендбука.
* Код должен быть документирован и соответствовать стандартам кодирования: Airbnb JavaScript Style Guide —для JavaScript.

1. **Совместимость**

* Совместимость с iOS, Android
* Сайт должен корректно отображаться во всех современных браузерах (Chrome, Firefox, Safari, Edge) начиная с версий, выпущенных не позднее 2 лет назад.
* Система должна интегрироваться с существующими системами CRM и ERP.
* API должен поддерживать протоколы REST и SOAP

1. **Переносимость (возможность работы приложения на разных устройствах)**

* Сайт должен работать на Windows, macOS и Linux.
* Сайт должен работать на разных мобильных устройствах (смартфоны, планшеты).
* Сайт должен быть адаптирован для мобильных устройств (смартфоны, планшеты) и десктопов с разрешением экрана от 320px до 1920px.
* Сайт должен быть оптимизирован для работы при низкой скорости интернета (например, 3G).
* Должна быть предусмотрена возможность работы в оффлайн-режиме (например, кэширование данных для просмотра товаров).

**Дополнительно Требования к дизайну и визуальному стилю**

* Цветовая палитра по брендбука (см. стандарт)
* Тип шрифтов по бренбуку
* Картинки д.б. анимационные (см. стандарт)
* Наименование товаров, участвующих в акции на текущий период, выделено красным цветом
* Сайт должен строго соответствовать цветовой палитре, указанной в брендбуке, с допустимым отклонением не более 5% по цветовому коду RGB.
* Шрифты, используемые на сайте, должны соответствовать шрифтам, утвержденным в брендбуке.
* Все графические элементы (иконки, кнопки, изображения) должны быть выдержаны в едином стиле, соответствующем брендбуку.

Воу, крутое ДЗ. Комменты оставила. Очень понравилось, что ты реально старался и искал информацию. Я все таки нашла пару функциональных требований, но это абсолютно не критично, особенно на таком объеме))). Так, теперь смотри, у тебя в доке было и про балансировщики, и про алгоритмы шифрования, алгоритмы предотвращения атак, обеспечение безопасности - это все очень здорово, но помним, что такие вещи уточняем у разработчиков /архитекторов/специалистов по ИТ безопасности, прежде чем начать пилить продукт и сдавать доки Заказчику. Спасибо за крутую работу. Домашка принята!

Яна, добрый день! А как на счёт NFTs в моем ДЗ без коментов. Их можно включать или они лишние? И насколько объемный этот раздел NFT? И в какой документ они входят?

на практике у меня НФТ около 30 листов в самом тех проекте, и еще отдельно в каждом ЧТЗ к проекту указаны кастомные НФТ (листов 3- 5), которые дополняют основные.

Что значит лишнее? Ты нашел их - уже хорошо, а надо ли оно твоей системе - решается только коллегиально, ну или если ты арх - можешь, конечно, сам

мы же тут без архитекторов пишем и без разработчиков, у тебя там есть дубли, плюс выбор конкретной технологии, определение скоупа тех же атак, от которых будете систему защищать, нужны ли вам все - это только коллективно обсуждать, потому что каждая примочка = деньги+время на разработку и ресурс.

вот пример, моей малюськи из общего тех проекта:

Требуемый режим работы и численность обслуживающего персонала Системы

Режим работы обслуживающего персонала должен соответствовать действующему законодательству Российской Федерации и обеспечивать работоспособность Системы согласно требованиям, предъявленным в ЧТЗ.

Должна быть учтена возможность сменного режима работы обслуживающего персонала Системы. При этом должна учитываться возможность круглосуточного подключения к работам специалистов, обеспечивающих функционирование Системы (обслуживающий персонал), для решения проблем по обеспечению работоспособности информационных ресурсов Системы.

Численность персонала должна определяться, исходя из количества необходимых автоматизированных рабочих мест на всех уровнях управления Системой и объемов выполняемых работ, и должна быть достаточной для функционирования Системы в соответствии с требованиями, приведенными в ЧТЗ.

А вот прям цифры для данного пункта указваются в ЧТЗ, которое прикладывается к общему тех проекту, потому что такая договоренность.

draw.io