**Определение целей и объёма проекта**

***Тип ПО для разработки:***веб приложение.

***Основная цель проекта:***Создание интернет-магазина аксессуаров, предоставляющего пользователям удобный способ просмотра каталога товаров (портфели, рюкзаки и прочее), фильтрации и сортировки, добавления товаров в корзину, управления количеством выбранных позиций и оформления фиктивного заказа.

***Задачи проекта***

1. Ознакомление с функциональностью и дизайном популярных интернет-магазинов аксессуаров для изучения лучших практик, удобства навигации и организации каталога.

2. Разработка архитектуры приложения: клиент-серверная.

3. Выбор технологий:

Разработка серверной части (BackEnd) – PHP, Короткин А.А.

База данных – MySQL

Хостинг для репозиторий - GitHub

Разработка клиентской части (FrontEnd) – HTM, CSS, JS, Короткин А.А.

Дизайн – Figma, Короткин А.А.

Тестирование – Короткин А.А.

Документирование – Короткин А.А.

4. Обеспечение функциональности реального времени:

* Обеспечение надёжной работы при высоких нагрузках.

5. Тестирование и отладка: проведение модульного и интеграционного тестирования.

Процесс создания включает:

1. Планирование и проектирование:

* Определение ключевых функций.

2. Разработка серверной части:

* Сервер на PHP обеспечивал маршрутизацию запросов и взаимодействие с базой данных.
* Подключение к базе данных MySQL.

Разработка клиентской части:

* Интерфейс пользователя (UI): создан удобный интерфейс на HTML с использованием css, js.
* Интеграция с серверной частью: организована передача данных между клиентом и сервером через API.

Сроки – система должна быть разработана и внедрена в течении 3 месяцев.

Объём проекта: разработка интернет-магазина аксессуаров – многопользовательской системы, позволяющей эффективно взаимодействовать с каталогом товаров, осуществлять фильтрацию и сортировку, управлять корзиной и оформлять фиктивные заказы.

Функциональные требования: просмотр каталога товаров, фильтрация и сортировка по различным параметрам, управление корзиной (добавление, изменение количества, удаление товаров), оформление фиктивного заказа.

Нефункциональные требования: высокая производительность и быстрая обработка запросов (не более 3 секунд), надёжная защита данных пользователей с использованием современных методов аутентификации.

**Сбор и анализ требований**

**Сбор информации:**

Проведение интервью с потенциальными пользователями для выявления предпочтений в онлайн-шопинге, удобства использования каталога и быстроты оформления заказа.

Опрос клиентов, чтобы определить, что для них важнее: подробные описания товаров, фильтрация, сортировка или скорость загрузки страниц.

**Анализ:**

*Проблемы:* необходимость создания интернет-магазина, который обеспечивает удобство выбора и оформления заказов, а также быстрый отклик системы при высоком трафике.

*Потребности:* пользователи хотят быстро находить интересующие товары, просматривать детальную информацию, управлять корзиной и оформлять заказы без лишних сложностей.

**Формулирование:**

*Функциональные потребности:*

* Просмотр и поиск товаров в каталоге.
* Фильтрация и сортировка.
* Управление корзиной.
* Оформление заказа (фиктивного).

*Нефункциональные потребности:*

* Интерфейс должен быть интуитивно понятным.
* Обеспечение надёжной работы при высоких нагрузках.
* Система должна обрабатывать запросы и запускаться не больше 3 с.

**Проектирование архитектуры и интерфейса**

Архитектура интернет-магазина включает следующие компоненты:

1. Сервер приложения – основной компонент, обрабатывающий все запросы пользователей, управляющий данными каталога, корзины и заказов. Он принимает запросы, обрабатывает их и передаёт ответы клиентскому приложению.
2. База данных - хранилище информации о товарах реализованное с использованием MySQL.
3. Система интеграции - позволяет интернет-магазину взаимодействовать с внешними сервисами (например, для симуляции платежей или аналитики), что расширяет функциональные возможности проекта.

Взаимодействие компонентов осуществляется через HTTP API, обеспечивая быструю и безопасную передачу данных между клиентской и серверной частями.

**Создание дорожной карты (Roadmap)**

**Фаза 1: Планирование**

* *Задача 1:* сбор и анализ требований (1-13 сентября 2024)
* *Задача 2:* разработка проектного плана и дорожной карты (14-25 сентября 2024)

**Фаза 2: Проектирование**

* *Задача 1:* Создание прототипов и макетов (13-25 сентября 2024)
* *Задача 2:* Утверждение дизайна (26 сентября 2024)

**Фаза 3: Разработка**

* *Задача 1:* Разработка клиентской части, фронтенда (27 сентября – 11 ноября 2024)
* *Задача 2:* Разработка серверной части, базы данных, бэкенда (11 октября – 11 ноября 2024)

**Фаза 4: Тестирование**

* *Задача 1:* Проведение функционального тестирования (11 ноября – 1 декабря 2024)
* *Задача 2:* Проведение нагрузочного тестирования (11 ноября – 1 декабря 2024)

**Фаза 5: Внедрение**

* *Задача 1:* Подготовка к запуску (1 декабря 2024)
* *Задача 2:* Запуск приложения (2 декабря 2024)

**Фаза 6: Поддержка**

* *Задача 1:* Техническая поддержка и исправление ошибок ( 3 декабря – 15 декабря 2024)

**Оценка рисков и план управления рисками**

*Оценка рисков:*

* Риски: задержка в разработке, технические проблемы, недостаточная пользовательская поддержка.
* План управления рисками: создание резервного плана, привлечение дополнительных ресурсов.

*Управление рисками:*

* Риски: задержка в разработке, проблема с интеграцией.
* План управления: регулярные проверки статуса, резервация времени для решения проблем.

**Оценка бюджета и ресурсов**

**Бюджет**

*Оценка затрат:* зарплата разработчиков, лицензии, серверы, среди них – 150 – 200 р. На хостинг.

*Управление бюджетом:* мониторинг затрат еженедельно, отчётность.

**Ресурсы**

*Основные этапы:* анализ и планирование, разработка, тестирование, внедрение, обучение.

*Ресурсы:* команда из 1 разработчика, сервер

**Подготовка документации**