Комментарии к презентации

**Слайд 1. Вступление**

Добрый день, сегодня мы хотим продемонстрировать презентацию на тему курсовой работы: “создание прототипа интернет-магазина”. Выполнили работу студенты 2 курса. Маркевич Александр Александрович и Максименко Александр Михайлович.

**Слайд 2. Введение**

Интернет-магазины – новое слово в коммерции. Всё больше людей предпочитают покупать товары онлайн. Создание и развитие интернет-магазина стали популярными направлениями в сфере веб-разработки. В современном мире все больше людей предпочитают покупать товары онлайн, и интернет-магазины становятся неотъемлемой частью коммерческой деятельности.

**Слайд 3. Цели и задачи**

**Цель данного** проекта реализовать прототип интернет-магазина, а также протестировать его функционал.

**Задачи.**

**1)** Выбрать стек технологий, на котором будет реализован сайт.

**2)** Подключение всех требуемых библиотек в проект.

**3)** Разработать дизайн web-приложения.

**4)** Разработать клиентскую и серверную часть приложения

**5)** Создать базу данных и заполнить ее тестовыми данными.

**6)** Объединить все в единую систему.

**Слайд 4. Стек технологий используемых в web-приложении**

Выбор frontend фреймворка.

При выборе frontend фреймворка для прототипа интернет-магазина следует остановить свой выбор на таком популярном фреймворке как Angular, так как он обеспечат гибкость и производительность при создании сайта.

Выбор backend фреймворка

Для backend прототипа магазина лучше всего подойдет такой фреймворк как .net core, так как он позволят легко интегрировать API и базу данных для управления каталогом и взаимодействием с ним. А также в дальнейшей разработке позволит реализовать многозадачность в приложении.

Выбор базы данных

При выборе СУБД для прототипа интернет-магазина был сделан выбор в пользу реляционной БД PostgreSQL из-за ее надежности и масштабируемости.

\*Тестирование

Был выбран Swagger для удобного тестирования api и базы данных.

**Слайд 5. Функционал и возможности**

Каталог товаров с описанием

На прототипе создается каталог товаров с описанием для проверки удобства интерфейса, а также проверки функционала фильтрации продуктов.

Поиск товаров

Реализован “умный” поиск товаров, в который можно вводить, как название товаров, так и хештеги.

Сортировки товаров

В приложения имеются сортировки, благодаря которым возможно быстро отыскать нужный товар.

**Слайд 6. ИНТЕРФЕЙС WEB-ПРИЛОЖЕНИЯ**

Интерфейс web-приложения нужен для навигации и удобства для пользователя. Сейчас продемонстрирую наш интерфейс.

**Слайд 7. Начальный экран web-приложения**

На начальном экране приложения отображаются товары в фиолетовых блоках. В каждом блоке имеется описание, цена и хештеги. В меню сайта располагается текстовое поля, а справа находится кнопка поиск.

**Слайд 8.** **После поиска товара экран web-приложения**

После поиска товара слева отображается меню сортировки, а в поиске уже заданно название товара.

**Слайд 9. АЛГОРИТМЫ РЕАЛИЗОВАННЫЕ НА КЛИЕНТСКОЙ ЧАСТИ**

Были реализованы сортировки по возрастанию, убыванию, в диапазоне, а также от а до я.

**Слайд 10-14.**

**isThePriceSuitable(fromWhatPrice: number, upToWhatPrice: number, price: number): boolean {**

**return (price >= fromWhatPrice && price <= upToWhatPrice) ? true : false;**

**}**

**\_sortDataInAscendingOrder(): void {**

**this.products.sort(function(obj1, obj2) {**

**return obj1.price-obj2.price;**

**})**

**this.filteredListOfProducts = this.products;**

**}**

**\_sortDataInDescendingOrder(): void {**

**this.products.sort(function(obj1, obj2) {**

**return obj2.price-obj1.price;**

**})**

**this.filteredListOfProducts = this.products;**

**}**

**sortDataInAscendingOrder(array: Products[]): Products[] {**

**array.sort(function(obj1, obj2) {**

**return obj1.price-obj2.price;**

**})**

**return array;**

**}**

**sortFromAToZ(): void {**

**this.products.sort(function(obj1, obj2) {**

**if (obj1.productName < obj2.productName) return -1;**

**if (obj1.productName > obj2.productName) return 1;**

**return 0;**

**})**

**this.filteredListOfProducts = this.products;**

**}**

**Слайд 15. ОСНОВНЫЕ КОМПОНЕНТЫ СЕРВЕРНОЙ ЧАСТИ**

Создание Http-методов позволяет использовать основные операции,например Get запрос для получения нужной веб-клиенту информации, использование web-api для взаимодействия между frontend’ом и backend’ом, также подключение БД позволяет взаимодействовать с данными.

**Слайд 16. Как выполняется алгоритм поиска на примере**

На данном слайде представлен пример алгоритма поиска.

**Слайд 17-20. Тестирование на Swagger’e**

Improvisation.

**Слайд 21. ВОЗНИКШИЕ ТРУДНОСТИ В СОЗДАНИИ WEB-ПРИЛОЖЕНИЯ**

При создании прототипа интернет-магазина возникли трудности, с совместимостью поиска из frontend’a к backend’y засчет неправильного восприятия API backend’a символов frontend’a, конвертацией csv файла в DB, nuget пакетами.

**Слайд 22. Что не удалось реализовать**

Не был реализован вывод товаров с картинками. Также хотелось исправить проблему с вводом с ошибочной раскладкой (если ввод производится с помощью ошибочной русской раскладкой) и замена е на ё

**Слайд 23. ВЫВОД**

Были реализованы алгоритмы для поиска и сортировки сайта. Также был разработан интерфейс приложения.