



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ _____
КАФЕДРА _____

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

К КУРСОВОМУ ПРОЕКТУ

НА ТЕМУ:

Студент _____
(Группа)

(Подпись, дата)

(И.О.Фамилия)

Руководитель курсового проекта

(Подпись, дата)

(И.О.Фамилия)

Консультант

(Подпись, дата)

(И.О.Фамилия)

**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

УТВЕРЖДАЮ
Заведующий кафедрой _____
(Индекс)

(И.О.Фамилия)
« ____ » _____ 2020 г.

**З А Д А Н И Е
на выполнение курсового проекта**

по дисциплине Базы данных

Студент группы ИУ7-65

Юмаев Артур Русланович
(Фамилия, имя, отчество)

Тема курсового проекта _____

Направленность КП (учебный, исследовательский, практический, производственный, др.) _____

Источник тематики (кафедра, предприятие, НИР) _____

График выполнения проекта: 25% к ____ нед., 50% к ____ нед., 75% к ____ нед., 100% к ____ нед.

Задание _____

Оформление курсового проекта:

Расчетно-пояснительная записка на _____ листах формата А4.

Перечень графического (иллюстративного) материала (чертежи, плакаты, слайды и т.п.)

Дата выдачи задания « ____ » _____ 2020 г.

Руководитель курсового проекта

(Подпись, дата)

(И.О.Фамилия)

Студент

(Подпись, дата)

(И.О.Фамилия)

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

Оглавление

Введение.....	5
1. Аналитическая часть.....	7
1.1 Особенности выбора языка программирования и веб-фреймворка	7
1.2 Особенности выбора СУБД.....	7
1.3 Выбор инструментов для разработки гибкого интерфейса.....	8
1.4 Технологический стек	9
2. Конструкторская часть	11
3. Технологическая часть	12
4. Экспериментальная часть.....	13
Список используемой литературы	14

Введение

В настоящее время покупки товаров через интернет стали обыденным явлением для современного человека. В 2017 году объем покупок в интернете достиг \$2.3 трлн., который, как ожидается, увеличится до \$4.5 трлн. к 2021 году. Интернет-покупатели все чаще ищут покупки за пределами своей страны, в среднем около 57% онлайн покупок совершается у зарубежных магазинов. Объем покупок в интернете в 2018 году был около \$17.2 млрд. Аналитики прогнозируют рост на 44.2% к 2023 году, что должно составить \$24.8 млрд. [1]. Статистика поведения покупателей в сфере электронной коммерции показывает, что 43% онлайн-покупателей сообщили, что совершают покупки из дома, 23% - в офисе и 20% - в ванной или в машине. Южная Корея (77%), Германия (76%), Китай (68%), Индия (68%) и Англия (67%) лидируют по покупкам одежды через интернет [2]. Таким образом, наблюдая рост отрасли электронной коммерции, компании-производители все больше заинтересованы в продаже товаров через интернет. Некоторые компании имеют собственные отделы разработки, тестирования и поддержки интернет-магазина, другие вынуждены обращаться фирмы, предоставляющие услуги по созданию сайтов.

Современные интернет-магазины, такие как Amazon, могут иметь ассортимент, состоящий из сотен миллионов товаров. На 2020 год Amazon имеет порядка 600 млн. товаров, доступных для заказа, а ежедневное пополнение составляет около 1,3 млн. товаров. За 2017 год Amazon продал 2.7 млрд. товаров через интернет.

Для обслуживания подобного объема продаж требуется ПО, позволяющее эффективно справляться с высокой нагрузкой. Для хранения информации существуют базы данных и системы управления базами данных (СУБД) с оптимизированными алгоритмами поиска, сортировки, ранжирования и группировки данных. Например Oracle или MS SQL Server.

Цель данной работы – реализовать интернет-магазин с пользовательской корзиной, а также функцией фильтрации и сортировки товаров.

Для достижения поставленной цели требуется решить следующие **задачи**.

1. выбрать язык программирования, фреймворк для создания веб-сервера и http сервер;
2. выбрать CSS библиотеку для разработки адаптивного интерфейса;
3. выбрать базу данных для хранения информации и API (Application Programming Interface) для выбранной базы данных;
4. спроектировать и разработать базу данных;
5. спроектировать и разработать веб-приложение с помощью выбранных технологий;
6. разработать гибкий интерфейс для созданного веб-приложения;
7. использовать контейнеризацию Docker для развертывания готового проекта;

1. Аналитическая часть

В данном разделе будет проведена работа по анализу существующих решений для реализации поставленной задачи, а именно создания веб-сайта для электронной коммерции.

1.1 Особенности выбора языка программирования и веб-фреймворка

На сегодняшний день существует несколько основных технологических стеков для реализации проектов в сфере веб. В них входит язык программирования Haskell (веб-фреймворки: Yesod, Snap, Misso), JavaScript (веб-фреймворки Node.js, React), Java (веб-фреймворк Spring), PHP (веб-фреймворк Laravel) а также Python в связке с фреймворками Django, Flask, либо, если требуется поддержка асинхронности на веб-сервере, Tornado или Sanic. Разрабатываемый интернет-магазин не предполагает повышенной нагрузки и сложной структуры веб-страниц, поэтому подходящим решением будет микрофреймворк, представляющий из себя минималистичный каркас веб-приложения, предоставляющий лишь самые базовые возможности.

Так как имелся опыт разработки веб-сервисов на Python, было принято решение использовать данный язык программирования в связке с Flask, так как он подходит для учебных целей и быстрой разработки простых веб-сайтов.

В качестве http сервера сравнивались Apache HTTP Server и Nginx. Nginx был создан для решения так называемой проблемы c10k – проблемы 10 тысяч соединений, что означает, что веб-сервера, использующие потоки не могут обрабатывать запросы пользователей более, чем с 10,000 подключений одновременно. Так как проект не предполагает высокой нагрузки, выбор был сделан в пользу Apache, он имеет более простую настройку и подходит для начинающих. 46% веб сайтов в сети Интернет используют данный http сервер, который также является бесплатным, даже в коммерческих целях.

1.2 Особенности выбора СУБД

Основная задача базы данных в данном проекте – сохранять и выдавать данные в нужном формате, определенном программой, при этом требуется делать это быстро, так как исследования показывают, что пользователь может ждать ответа от программы около 3

секунд, после этого у него начинает возникать дискомфорт. Также в рамках задачи становится нецелесообразно использовать внешний сервер ввиду избыточности. Следовательно, было принято решение, что данные будут храниться локально. В программе не предполагается большой (от нескольких ГБ) объем данных для хранения.

Для данной задачи подойдут встраиваемые бессерверные (serverless) СУБД, имеющие несколько следующих отличий от классических клиент-серверных.

- высокая скорость и малое потребление памяти, благодаря упрощенному API;
- локальное хранение данных, фактически СУБД является библиотекой, работающей с файловой системой ОС;
- расчет на однопользовательскую работу с небольшим (по меркам клиент-серверных СУБД) объемом данных;

Примеры встраиваемых СУБД:

- InfinityDB
- SQLite
- Microsoft SQL Server Compact

Некоторые встраиваемые СУБД также являются платными, хотя и стоят значительно дешевле, чем серверные платформы. Ключевым фактором в ценообразовании могут стать не только качественные характеристики продукта, но и дополнительные возможности, предоставляемые конечному пользователю. Например, графическая среда моделирования структуры данных, визуальные конструкторы запросов и т.д. Эффективное проектирование базы данных и ее администрирование считается трудоемкой, требующей высокой квалификации работой.

1.3 Выбор инструментов для разработки гибкого интерфейса

В настоящее время в мире высока популярность различных персональных компьютеров. Планшеты, смартфоны, умные часы с выходом в интернет, умные телевизоры и так далее. Современная реалии требуют от веб-сайта простоты и удобства использования

на устройствах с различным размером и формой экрана. Именно поэтому так важно, чтобы сайт был способен автоматически меняться в зависимости от размера и пропорций дисплея. Ассортимент инструментов для разработки гибких и адаптивных интерфейсов на данный момент не велик. Еще несколько лет назад веб-разработчики придерживались схемы так называемой блочной верстки, когда проектирование веб-страницы происходило таблицами, которые в свою очередь могли в каждой своей ячейке содержать другие таблицы. На данный момент разработчики мыслят гибкими блоками, которые с помощью JavaScript кода определяют параметры дисплея и подстраивают контент веб-страницы по нему.

Самым популярной CSS библиотекой на данный момент остается Bootstrap от компании Twitter. Она и была выбрана для разработки проекта.

1.4 Технологический стек

Развертывание проекта будет происходить на облачной платформе Microsoft Azure. Она предоставляет возможность разработки, выполнения приложений и хранения данных на серверах, расположенных в распределённых дата-центрах. В качестве виртуальной машины выбрана Ubuntu версии 16.04.

В итоге технологический стек будет состоять из двух частей: серверной (бэкенд) и клиентской (фронтенд).

Бэкенд

- Язык программирования Python
- Фреймворк Flask для быстрой реализации простых одностраничных веб-приложений на модели MVC (Model – View – Controller)
- Apache HTTP-сервер – кроссплатформенный веб-сервер для обработки соединений с юзер-агентом
- SQLite реляционная база данных
- Docker — разработка, развертывание на сервере
- Microsoft Azure – облачный сервер с Ubuntu 16.04

Фронтенд

- Верстка макета с помощью HTML, CSS, JavaScript
- Twitter Bootstrap для быстрой и адаптивной разработки интерфейсов

2. Конструкторская часть

3. Технологическая часть

4. Экспериментальная часть

Заключение

Список используемой литературы

[1] - Global Ecommerce Statistics and Trends to Launch Your Business Beyond Borders

[Электронный ресурс]. 2020. Дата обновления: 24.03.20. URL:

<https://www.shopify.com/enterprise/global-ecommerce-statistics> (дата обращения: 24.03.20).

[2] - Ecommerce Statistics for 2020 – Chatbots, Voice, Omni-Channel Marketing [Электронный

ресурс]. 2020. Дата обновления: 24.03.20. URL: <https://kinsta.com/blog/ecommerce-statistics/>

(дата обращения: 24.03.20).