



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(ДГТУ)**

ФАКУЛЬТЕТ «Информатика и вычислительная техника»
КАФЕДРА «Кибербезопасность информационных систем»

КУРСОВАЯ РАБОТА

Тема: «Разработка WEB - приложения интернет-магазина по продаже обуви»

Дисциплина: «WEB-программирование»

Направление подготовки: 09.03.01 Информатика и вычислительная техника

Направленность: Информатика и вычислительная техника

Обозначение курсового проекта WP.710000.000 Группа ВИ-42

Обучающийся _____ А. А. Грибко
(подпись, дата)

Курсовая работа защищена с оценкой _____

Руководитель работы _____ доцент, И. А. Сосновский
(подпись, дата)

Ростов-на-Дону
2024



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(ДГТУ)**

ФАКУЛЬТЕТ «Информатика и вычислительная техника»
КАФЕДРА «Кибербезопасность информационных систем»

ЗАДАНИЕ
на выполнение курсовой работы

Тема «Разработка WEB - приложения интернет-магазина по продаже обуви»

Дисциплина: WEB-программирование

Обучающийся: Александр Александрович Грибко

Обозначение курсовой работы WП.430000.000 Группа: ВИ-42

Срок представления проекта (работы) к защите «__» _____ 20__ г.

Исходные данные для курсового проекта:

1. Учебные материалы, электронные ресурсы, периодические материалы.
2. Официальный сайт организации, для которой разрабатывается web-приложение, сайты аналогичных компаний.
3. Рекомендации по выполнению работы.
4. Рекомендации по оформлению работы.

Содержание пояснительной записки

Введение.

1. Теоретические основы разработки сайтов.
2. Сбор информации об объекте.
3. Создание базы данных.
4. Создание карты сайта или макета сайта.
5. Создание шаблона дизайна страницы.
6. Формирование контента.
7. Вёрстка сайта.
8. Проверка Web-приложения на работоспособность.

Заключение.

Перечень использованных информационных ресурсов.

Приложение А.

Руководитель проекта _____ И. А. Сосновский
(подпись, дата)

Задание принял к исполнению _____ А. А. Грибко
(подпись, дата)

СОДЕРЖАНИЕ

Введение.....	4
1. Теоретические основы разработки сайтов.....	6
1.1 Язык гипертекстовой разметки HTML.....	6
1.2 Язык программирования JavaScript.....	7
1.3 Язык программирования PHP.....	7
1.4 Сервер баз данных MySQL.....	8
2. Сбор информации об объекте.....	10
3. Создание базы данных.....	13
4. Создание карты сайта или макета сайта.....	16
5. Создание шаблона дизайна страницы.....	19
6. Формирование контента.....	21
7. Вёрстка сайта.....	23
8. Тестирование и размещение сайта в сети.....	26
Заключение.....	28
Перечень использованных информационных ресурсов.....	30
Приложение А.....	31

					ИП.430000.000 ПЗ			
Изм.	Лист	№ докум.	Подпись	Дата				
Разраб.		Грибко А. А.			Разработка WEB - приложения интернет-магазина по продаже обуви Пояснительная записка	Лит.	Лист	Листов
Провер.		Сосновский И.А.					3	44
Реценз						ДГТУ кафедра «КБИС»		
Н. Контр.								
Утверд.		Сафарьян О.А.						

Введение

Каждый раз, заходя в глобальную сеть, мы сталкиваемся с сайтами. Именно на них мы просматриваем интересующие нас новости, товары, услуги и т.д. Глобальная цифровизация и гаджетизация привела к тому, что видоизменяются целые отрасли экономики. Продажа всей номенклатуры товаров всё больше переходить на онлайн платформы, тоже касается и предоставления банковских, юридических, образовательных и многих других. Исходя из этого, рассматриваемая тематика курсовой работы направленная на разработку сайтов с функционалом взаимодействия с клиентом является актуальной с практической точки зрения.

Актуальность WEB - приложения интернет-магазина по продаже компьютерных комплектующих в современных условиях велика в связи с увеличивающимся спросом на компьютеры и комплектующие к ним. С развитием технологий и повсеместным использованием компьютеров в различных областях жизни, потребность в покупке компьютерных комплектующих становится все более актуальной.

WEB - приложение интернет-магазина позволит покупателям удобно и быстро выбирать и приобретать необходимые комплектующие, сравнивать цены и характеристики товаров, а также осуществлять онлайн-оплату и отслеживать статус доставки. Это сэкономит время покупателей и сделает процесс покупки более удобным.

Также WEB - приложение может предоставлять информацию о новинках и акциях, что будет стимулировать покупателей совершать покупки.

Таким образом, WEB - приложение интернет-магазина по продаже компьютерных комплектующих будет актуальным и востребованным инструментом как для покупателей, так и для продавцов.

					ВП.430000.000 ПЗ	Лист
						4
Изм	Лист	№ документа	Подпись	Дата		

Организация, которая будет использовать WEB – приложение, специализируется на продаже компьютерных комплектующих в интернете. В ассортименте имеются процессоры, видеокарты, материнские платы, оперативная память и другие комплектующие для сборки и обновления компьютеров. Клиенты могут выбирать из широкого ассортимента товаров, сравнивать цены и характеристики.

					ВП.430000.000 ПЗ	Лист
						5
Изм	Лист	№ документа	Подпись	Дата		

1 Теоретические основы разработки сайтов

1.1 Язык гипертекстовой разметки HTML

HTML (HyperText Markup Language) — это стандартный язык разметки, используемый для создания WEB - приложений. Он состоит из различных элементов, которые используются для определения структуры содержимого WEB - приложений.

HTML используется для создания текстовых блоков, заголовков, списков, таблиц, ссылок, изображений, форм и других элементов, которые делают WEB – приложения интерактивными и информативными. В HTML также присутствуют элементы для разметки мультимедийного контента, такого как видео, аудио, gif - изображения.

Одной из особенностей HTML является возможность создания гиперссылок, которые позволяют пользователям переходить с одной веб-страницы на другую.

HTML обладает возможностью подключения стилей через CSS. Благодаря чему HTML обладает гибкостью в оформлении и распределении элементов в WEB - приложении, что позволяет создавать привлекательный дизайн.

HTML также обладает возможностью валидации, что позволяет разработчикам убедиться, что их код соответствует определенным стандартам и правилам.

В целом, HTML играет ключевую роль в создании WEB - приложений, и он является основой для большинства WEB - приложений в интернете.

1.2 Язык программирования JavaScript

JavaScript - это высокоуровневый интерпретируемый язык программирования, который часто используется для создания интерактивных WEB - приложений. Он предоставляет возможность добавления динамического поведения к WEB - приложениям, включая анимацию, изменение содержимого страницы без перезагрузки, работу с формами, валидацию введенных данных и многое другое.

JavaScript поддерживается всеми современными браузерами и является одним из ключевых языков для разработки WEB - приложений. Он также может использоваться для написания серверной части приложений, с использованием платформы Node.js.

JavaScript предоставляет разработчикам широкий спектр возможностей, включая манипуляцию DOM (Document Object Model), взаимодействие с пользователем, асинхронную загрузку данных с сервера без перезагрузки страницы (AJAX), создание анимаций и многое другое. Он также обладает мощными инструментами для работы с событиями, таймерами и обработками ошибок.

Кроме того, с появлением новых стандартов языка JavaScript получил множество новых возможностей, таких как стрелочные функции, параметры по умолчанию, классы, модули и другие синтаксические улучшения, что делает его еще более мощным инструментом для разработки веб-приложений.

1.3 Язык программирования PHP

PHP (Hypertext Preprocessor) - это интерпретируемый скриптовый язык программирования общего назначения, который преимущественно применяется для разработки WEB - приложений и динамических веб - сайтов. PHP выполняется на сервере, а результаты его исполнения

отправляются в виде HTML клиентскому браузеру, что позволяет создавать динамические веб - страницы и обрабатывать данные на стороне сервера.

PHP широко используется в сочетании с HTML, что позволяет встраивать PHP - код непосредственно в HTML - страницы. Он также обладает мощными возможностями взаимодействия с базами данных, включая поддержку большинства типов баз данных, таких как MySQL, PostgreSQL, Oracle и многих других.

Язык PHP поддерживает различные парадигмы программирования, такие как процедурное программирование и объектно - ориентированное программирование, что делает его гибким инструментом разработки.

PHP также предоставляет функциональные возможности для взаимодействия с файловой системой, отправки электронной почты, управления сессиями пользователей и другие возможности, что делает его идеальным для создания WEB - приложений. Кроме того, благодаря наличию большого сообщества разработчиков и обширной документации PHP является одним из наиболее популярных языков программирования для веб - разработки.

1.4 Сервер баз данных MySQL

MySQL - это универсальная система управления реляционными базами данных (СУБД), которая широко используется в веб - разработке для хранения и управления данными. MySQL имеет открытый исходный код и широко поддерживается сообществом разработчиков.

MySQL поддерживает множество функций и возможностей реляционных баз данных, включая создание и управление базами данных, таблицами, индексами, хранимыми процедурами, триггерами, функциями. Благодаря этим возможностям, MySQL может обрабатывать большие объемы данных, обеспечивая производительность и надежность. Он также обладает

мощными инструментами для обеспечения безопасности данных, включая различные механизмы авторизации, шифрование и управление доступом.

MySQL может быть легко интегрирован с различными языками программирования, такими как PHP, Python, Java и другими, что делает его популярным выбором для веб - приложений.

Благодаря своей простоте в установке и настройке, а также в сочетании с высокой производительностью и масштабируемостью, MySQL является одной из самых используемых СУБД во всем мире.

При разработке WEB - приложения использовался инструмент управления базами данных - phpMyAdmin. Он предоставляет удобный способ управления базами данных MySQL без необходимости использования командной строки или других клиентских приложений.

					ВП.430000.000 ПЗ	Лист
						9
Изм	Лист	№ документа	Подпись	Дата		

2 Сбор информации об объекте

WEB – приложение по продаже компьютерных комплектующих предназначено для того, чтобы предоставить пользователям возможность приобретения различных деталей и компонентов для своих компьютеров, ноутбуков и других устройств. Здесь можно найти процессоры, материнские платы, видеокарты, оперативную память и другие товары, которые необходимы для сборки или улучшения персонального компьютера.

Назначение такого сайта включает в себя предоставление пользователям широкого выбора комплектующих различных производителей, информацию о их характеристиках, совместимости с другими компонентами, а также возможность быстрого и удобного онлайн - заказа. Кроме того, пользователи могут получить консультацию по выбору комплектующих, сравнить цены и условия доставки различных поставщиков, а также ознакомиться с отзывами других покупателей.

Такой сайт удобен как для тех, кто собирает компьютерную систему с нуля, так и для тех, кто хочет заменить или улучшить отдельные компоненты своего устройства. Все это делает сайт по продаже компьютерных комплектующих необходимым инструментом для людей, кто интересуется техникой, в частности компьютерами.

WEB - приложение по продаже компьютерных комплектующих должно помочь пользователям достичь следующих целей:

1. Удовлетворение потребностей в комплектующих: Пользователи могут легко найти необходимые комплектующие для сборки или обновления своей компьютерной системы.

2. Предоставление широкого ассортимента товаров: Предоставление большого ассортимента комплектующих различных производителей, чтобы пользователи могли выбрать оптимальное решение в соответствии с их требованиями и бюджетом.

3. Информирование и консультация: Предоставление информации о характеристиках комплектующих, их совместимости с другими устройствами, а также консультации для помощи пользователям в выборе подходящих компонентов.

4. Обработка заказов: Заказ и оплата комплектующих должны быть удобными и безопасными для пользователей, чтобы они могли получить свои товары быстро и без проблем.

5. Поддержка пользователя: Предоставление возможности для пользователей обмениваться мнениями, задавать вопросы и получать помощь от других покупателей или специалистов.

Таким образом, WEB - приложение по продаже компьютерных комплектующих должно помочь пользователям облегчить и ускорить процесс выбора, приобретения и использования комплектующих для их компьютерных систем.

Целевая аудитория нашего WEB - приложения по продаже компьютерных комплектующих включает в себя разнообразные группы потребителей. Можно выделить несколько основных:

1. Геймеры и энтузиасты компьютерных игр: Люди, которые уделяют большое внимание производительности своих компьютерных систем для запуска требовательных игр.

2. Профессиональные пользователи: Это могут быть профессиональные редакторы видео и фотографии, разработчики программного обеспечения, а также инженеры и архитекторы, нуждающиеся в мощных компьютерных системах.

3. Системные администраторы и IT-специалисты: Люди, которые работают в области информационных технологий и нуждаются в комплектующих для обслуживания и обновления компьютерных сетей и серверов.

4. Студенты и учащиеся: Люди, учащиеся в образовательных учреждениях и нуждающиеся в комплектующих для учебных или домашних компьютеров.

5. Обычные пользователи: Люди, которые желают улучшить или заменить комплектующие своего домашнего компьютера или ноутбука.

Целевая аудитория также может быть разделена по демографическим данным, таким как возраст, пол, доходы и т. д. Каждая из этих групп может иметь разные потребности и предпочтения при выборе компьютерных комплектующих, и WEB - приложение должно учитывать эти различия в своей маркетинговой стратегии.

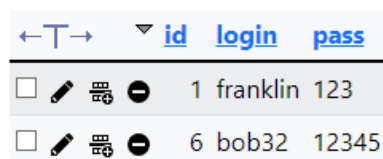
					ВП.430000.000 ПЗ	Лист
						12
Изм	Лист	№ документа	Подпись	Дата		

3 Создание базы данных

При разработке WEB - приложения использовался инструмент управления базами данных - phpMyAdmin. Он предоставляет удобный способ управления базами данных MySQL без необходимости использования командной строки или других клиентских приложений. При разработке WEB – приложения необходимо реализовать личный кабинет пользователя, поэтому обязательна таблица пользователей. Помимо этого, обязательной таблицей является таблица товаров, которая будет содержать всю информацию о товарах, а именно компьютерных комплектующих.

База данных будет содержать несколько таблиц:

1. Таблица пользователей (рисунок 1). В таблице пользователей будут содержаться данные о пользователях, их ID, логин и пароль, которые необходимы для авторизации на сайте.



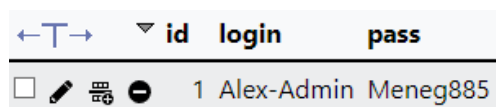
The screenshot shows a table with columns 'id', 'login', and 'pass'. There are two rows of data: one with id '1', login 'franklin', and pass '123'; the other with id '6', login 'bob32', and pass '12345'. Each row has icons for insert, edit, delete, and a checkbox.

	id	login	pass
<input type="checkbox"/>	1	franklin	123
<input type="checkbox"/>	6	bob32	12345

Рисунок 1 – таблица пользователей

2. Таблица данных пользователей. В таблице данных пользователей размещаются личные данные пользователя, которые отображаются в его личном кабинете – ID, ФИО, возраст, город проживания. Данная таблица имеет связь с таблицей пользователей по ID.

3. Таблица администраторов (рисунок 2). В таблице администраторов содержатся данные, необходимые для авторизации под администратором – ID, login, pass.



The screenshot shows a table with columns 'id', 'login', and 'pass'. There is one row of data: id '1', login 'Alex-Admin', and pass 'Meneg885'. There are icons for insert, edit, delete, and a checkbox.

	id	login	pass
<input type="checkbox"/>	1	Alex-Admin	Meneg885

Рисунок 2 – таблица администраторов

4. Таблица товаров (рисунок 3). В таблице товаров содержатся данные о товарах – название, тип товара, его изображение и доступное количество в данный момент. Администратор может изменить эти данные через админ – панель – добавить, изменить или удалить определённый товар.

	id	name	izo	nazv	kol-vo			
<input type="checkbox"/>	1	nike		2	nike		Nike x Dior Air Jordan 1 Retro High	5
<input type="checkbox"/>	3	adi		Forum 84 Low	25			
<input type="checkbox"/>	4	adi		Rivalry High	14			
<input type="checkbox"/>	5	adi		Campus 00s Grey White	20			
<input type="checkbox"/>	6	dior		B57	4			
<input type="checkbox"/>	7	dior		B27	7			

Рисунок 3 – таблица товаров

На рисунке 4 представлена модель базы данных со связями.

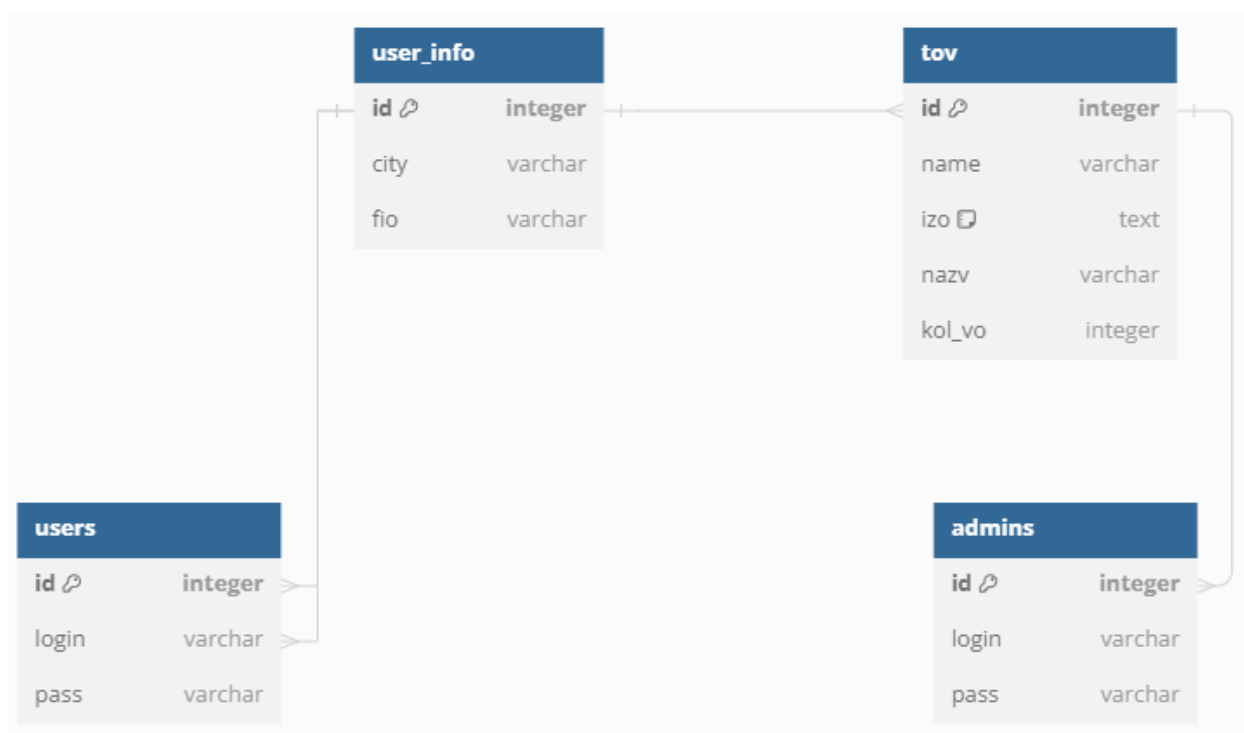


Рисунок 4 – связи между таблицами в базе данных

На изображении представлены четыре таблицы – users – таблица с паролями и логинами пользователя; uslc – таблица с личными данными пользователя, отображаемые в личном кабинете пользователя; tov – таблица с товарами; admins – таблица с паролями и логинами администраторов.

Таблица uslc связана с таблицами users – данные для входа в личный кабинет и tov – возможность оформления заказов. Таблица admins связана с таблицей tov – редактирование информации о товарах.

					ВП.430000.000 ПЗ	Лист
						15
Изм	Лист	№ документа	Подпись	Дата		

4 Создание карты сайта или макета сайта

Карта сайта представляет собой структурированное описание всего содержания веб - ресурса, разбитого на разделы и подразделы. Она помогает пользователям легко найти нужную информацию и обеспечивает удобную навигацию по сайту. Карта сайта также помогает поисковым системам быстрее индексировать страницы веб - сайта. Создание карты сайта важно для оптимизации его работы и повышения удобства использования. На рисунке 5 представлена карта сайта нашего WEB - приложения.

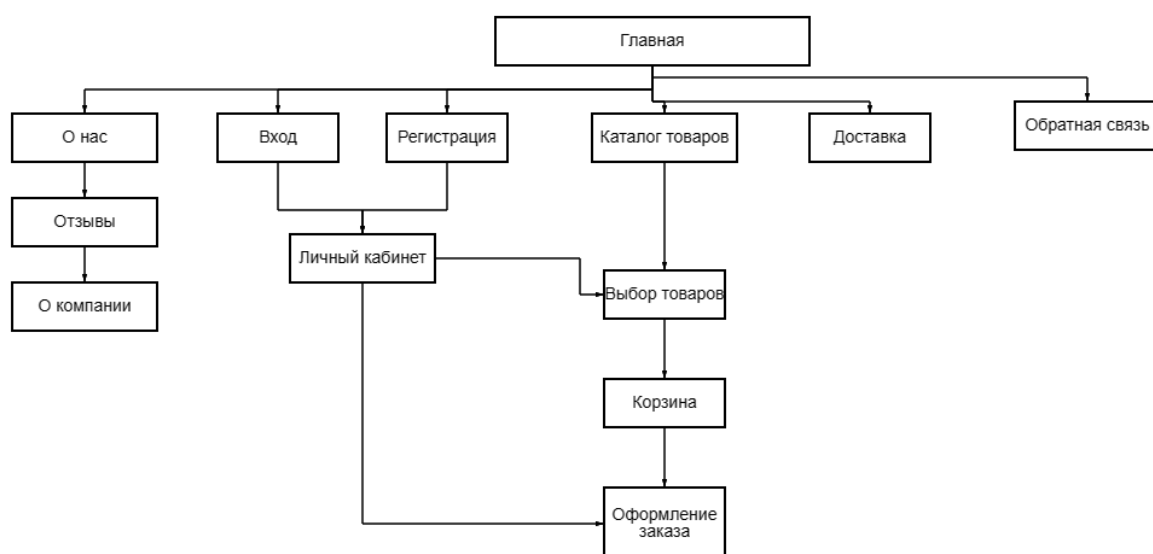


Рисунок 5 – карта сайта

Главная страница WEB – приложения содержит приветствие, общую информацию и навигационные ссылки к основными разделам.

На странице “О нас” описывается основная информация компании, её история, отзывы и контакты.

В личном кабинете пользователя отображается его основная информация – ФИО, возраст, город.

В каталоге товаров отображается ассортимент товаров, возможность выбрать определённый товар и поместить в корзину. В дальнейшем на

странице корзины пользователь имеет возможность приобрести выбранный товар, оформив заказ.

Макет сайта - это проект, который визуально представляет собой композицию веб-страницы или сайта в целом (рисунок 6). Он включает в себя расположение элементов, цветовую гамму, шрифты, изображения и другие визуальные компоненты.

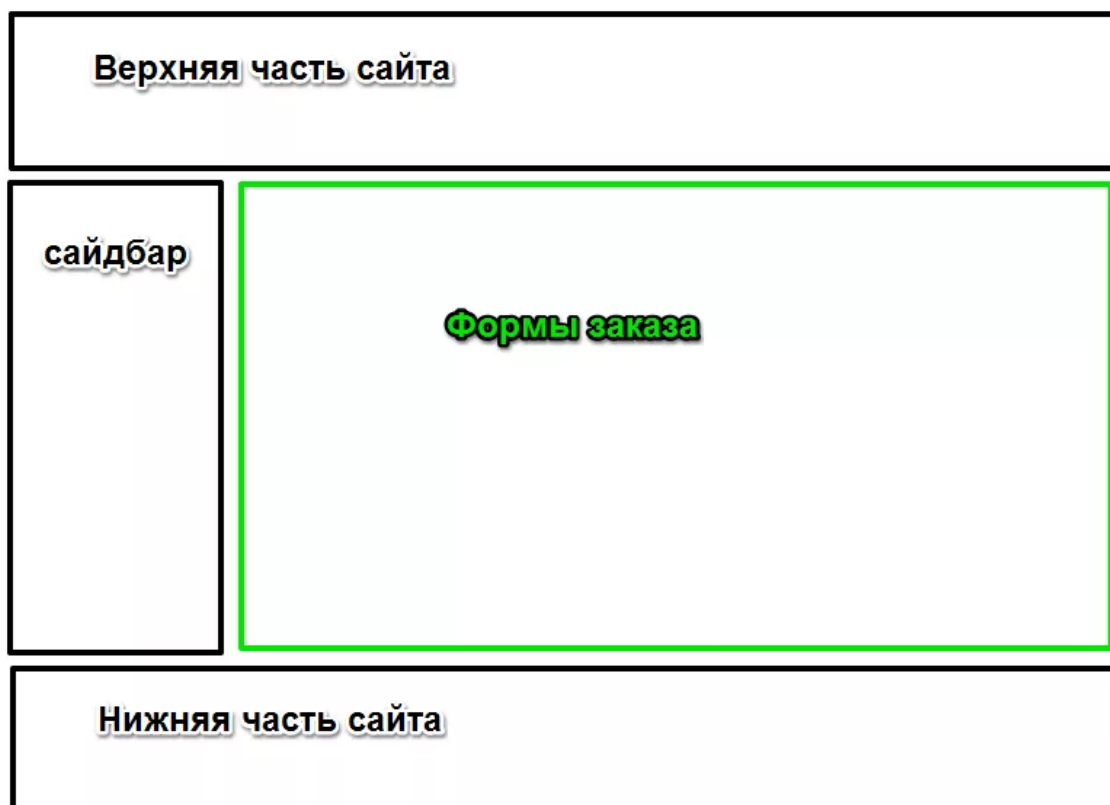


Рисунок 6 – макет сайта

Макет сайта включает в себя следующие аспекты:

1. Макет шапки (header) с логотипом, навигационными элементами, контактной информацией и, возможно, элементами для поиска.
2. Макет контента - расположение основной информации на странице, включая текст, изображения, видео и другие медиа-контенты.
3. Макет нижнего колонтитула (footer) - здесь обычно размещается дополнительная навигация, контактная информация, ссылки на социальные сети и другие важные элементы.

4. Макет всплывающих окон или модальных окон при необходимости.

					ВП.430000.000 ПЗ	Лист
						18
Изм	Лист	№ документа	Подпись	Дата		

5 Создание шаблона дизайна страницы

Создание шаблона дизайна страницы - это процесс разработки визуального макета страницы, который определяет расположение элементов, цветовую палитру, шрифты, графику и другие визуальные аспекты веб-страницы.

Шаблон WEB – приложения для продажи компьютерных комплектующих планируется в жёлтых оттенках. Жёлтый - это яркий и энергичный цвет, который может подчеркнуть определённые аспекты бренда. Предпочтительно используются оттенки жёлтого цвета, которые могут ассоциироваться с инновациями, энергией, радостью и оптимизмом.

На рисунке 7 представлен общий вид стилизованной страницы страниц без наполнения.

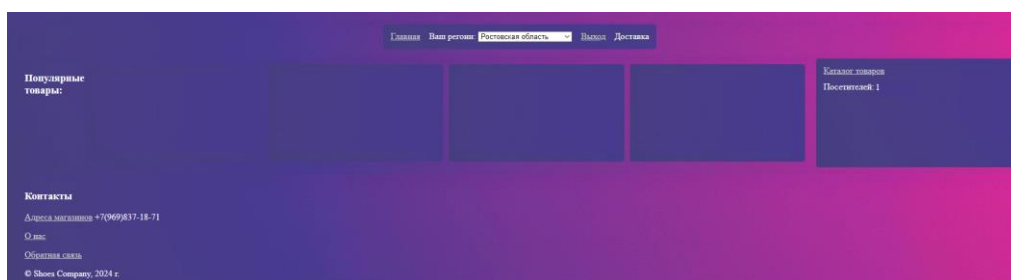


Рисунок 7 – общий вид стилизованной страницы без наполнения

Чтобы добиться контраста и удобства чтения, важно выбрать темные цвета для текста и элементов интерфейса. Жёлтый может быть использован для выделения особых элементов, но для текста используется чёрный цвет, чтобы сохранить удобство чтения.

Жёлтый цвет хорошо сочетается с цветными фотографиями, насыщенными оттенками. Фотографии компьютерных комплектующих на ярких фонах будут выглядеть привлекательно. Помимо этого жёлтый цвет может быть использован для выделения ключевых элементов, таких как кнопки "Купить", "Добавить в корзину" и других вызовов к действию.

Дизайн должен быть привлекательным и удобным для использования не только на десктопе, но и на мобильных устройствах.

6 Формирование контента

При формировании контента сайта по продаже компьютерных комплектующих, важно учитывать специфику аудитории, которая заинтересована в электронике и компьютерах.

Созданы детальные, но удобные для восприятия описания компьютерных комплектующих, таких как процессоры, видеокарты, оперативная память, мониторы и т.д. Важно описать технические характеристики, плюсы и минусы, рекомендации по использованию и поддержку товаров.

Для удобства пользователей важно использовать фотографии высокого качества и видеообзоры продуктов. Визуальный контент помогает покупателям увидеть товары ближе и сделать обоснованный выбор. На рисунке 8 представлен пример изображения, использованных в WEB – приложении.



Рисунок 8 – пример использования высококачественного изображения

Формирование контента для сайта по продаже компьютерных комплектующих должно быть направлено на предоставление полезной и информативной информации, которая поможет покупателям сделать обоснованный выбор и оценить предлагаемую продукцию.

Для SEO-оптимизации были использованы ключевые слова и фразы, связанные с компьютерными комплектующими, чтобы улучшить видимость сайта в поисковых результатах. В частности - "купить видеокарту", "новые процессоры", "обзор оперативной памяти".

					ВП.430000.000 ПЗ	Лист
						22
Изм	Лист	№ документа	Подпись	Дата		

7 Вёрстка сайта

Для запуска и тестирования на компьютере при разработке WEB – приложения использовалось программное обеспечение OpenServer (рисунок 9).

OpenServer - это популярное программное обеспечение, предназначенное для создания локальных веб-серверов на базе операционной системы Windows. OpenServer включает в себя веб-сервер Apache, интерпретатор PHP, СУБД MySQL, phpMyAdmin, а также другие инструменты, необходимые для разработки и тестирования веб-приложений и веб-сайтов.

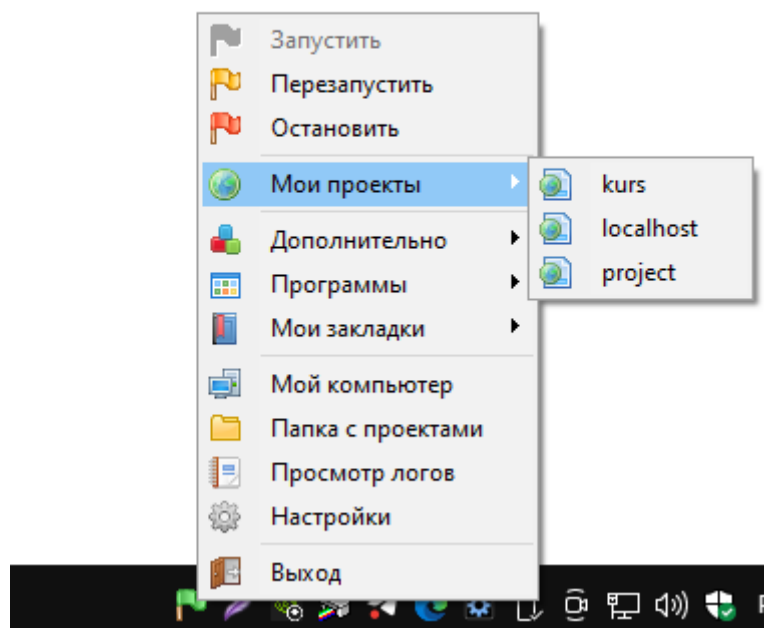


Рисунок 9 – запуск локального сервера при помощи OpenServer

Для запуска WEB – приложения с помощью OpenServer необходимо поместить папку с проектом в папку domain корневой папки OpenServer.

Папка проекта WEB – приложения содержит вид, показанный на рисунке 10.

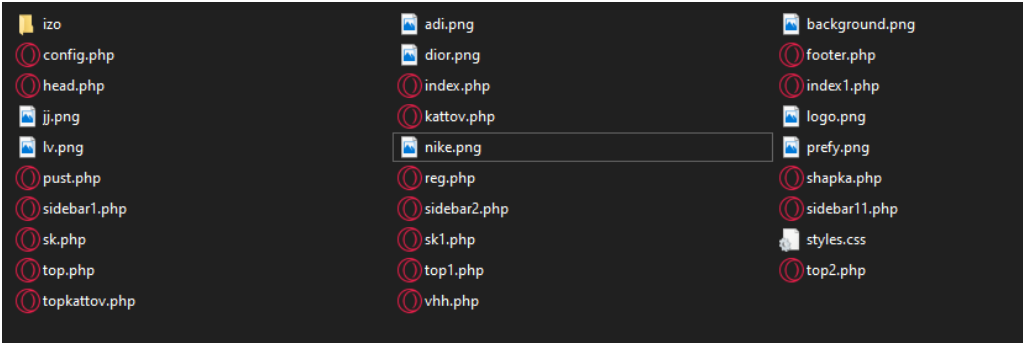


Рисунок 10 – папка проекта

В ней хранятся:

1. Папка izo. Главная папка изображений, содержащая изображения, отображающиеся в WEB – приложении.
2. index.php (рисунок 11). Файл определяющий структуру страницы сайта.

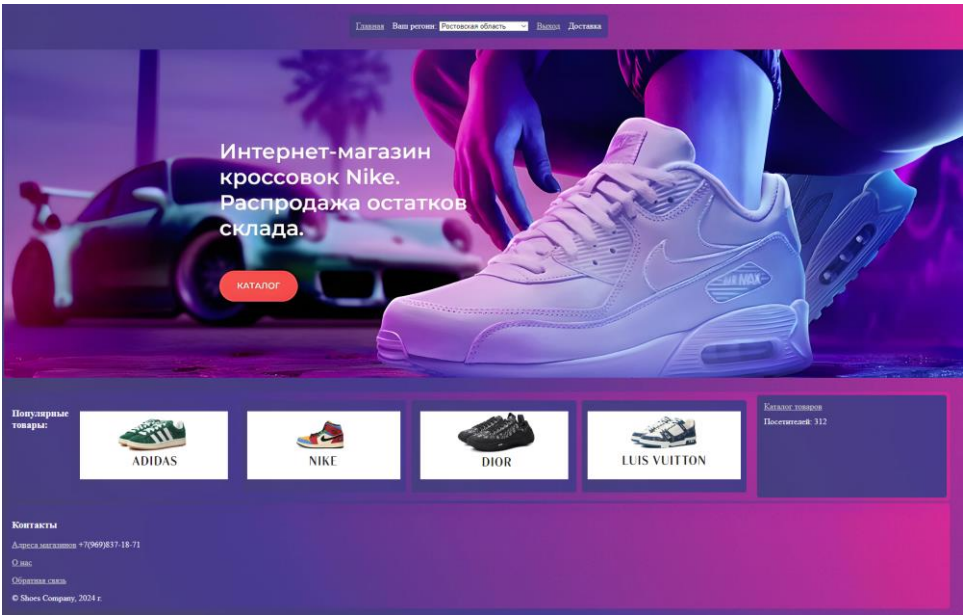


Рисунок 11 – sait.php

В файле строится DOM структура, и подключаются внешние файлы определяющие содержание и вид верхней части, боковой панели, а также нижней части сайта.

3. config.php. Файл, отвечающий за корректное подключение к базе данных и работе с ней.

4. footer.php. Файл, отвечающий за нижнюю часть WEB – приложения, “подвал”.

5. head.php. Файл, отвечающий за подключение скриптов, плагинов и пр. к проекту.

6. shapka.php. Файл, отвечающий за “шапку” сайта.

7. sidebar1.php. Файл, отвечающий за левую информационную панель проекта.

8. sidebar2.php. Файл, отвечающий за правую информационную панель нашего проекта.

9. lc.php. Файл, отвечающий за отображение личного кабинета пользователя на сайте. Содержит данные из базы данных – ФИО, возраст, город.

10. styles.css. Файл, содержащий графический дизайн WEB – приложения.

11. top.php. Файл, отвечающий за навигацию сайта (рисунок 12).

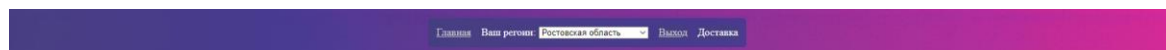


Рисунок 12 – top.php

12. vhh.php. Файл, отвечающий за страницу входа в личный кабинет пользователя.

13. reg.php. Файл, отвечающий за регистрацию пользователя, для дальнейшего входа в личный кабинет.

14. adm.php. Файл, отвечающий за вход и саму админ-панель. Администратор имеет возможность просматривать пользователей и товары. Редактировать, удалять и добавлять информацию о товарах.

8 Тестирование и размещение сайта в сети

Программа тестирования WEB - приложения по продаже компьютерных комплектующих включает в себя ряд шагов и функциональных тестов, чтобы убедиться в его корректной работе и удобстве использования для пользователей.

1. Тестирование интерфейса: проверка наличия всех необходимых разделов и функций, удобство навигации по сайту, корректное отображение информации на различных устройствах (пк, планшет, смартфон).

2. Тестирование функциональности поиска: проверка точности и скорости поиска по различным критериям (название товара, категория, бренд и т.д.).

3. Тестирование процесса добавления товара в корзину и оформления заказа: проверка корректности отображения выбранных товаров, удобства процесса оформления заказа, включая ввод личных данных и выбор способа оплаты.

4. Тестирование обратной связи и поддержки: проверка работы форм обратной связи, скорости ответов на вопросы клиентов и корректности обработки претензий и жалоб.

5. Тестирование безопасности: проверка наличия защищенного соединения, включая SSL-сертификат, а также защиты от взлома и мошенничества.

6. Тестирование производительности: проверка скорости загрузки страниц, отклика сервера и работы сайта при большом количестве пользователей.

После проведения тестирования WEB – приложения можно сделать вывод, что все функции работают корректно.

Процесс размещения сайта на хостинг включает в себя несколько этапов:

1. Выбор хостинг-провайдера: существует множество компаний, предоставляющих услуги хостинга, поэтому важно выбрать надежного провайдера с подходящими для вас тарифными планами и уровнем поддержки.

2. Регистрация на хостинге: необходимо создать учетную запись на выбранном хостинг-провайдере и приобрести подходящий тарифный план с нужными ресурсами (дисковое пространство, трафик, поддержка языков программирования и т. д.).

3. Получение доступа к серверу: после регистрации на хостинге получаем доступ к серверу, где будет размещаться сайт.

4. Загрузка файлов сайта: используя FTP или другие методы, необходимо загрузить файлы вашего сайта (HTML, CSS, JavaScript, изображения и другие ресурсы) на сервер хостинг-провайдера.

5. Настройка доменного имени: если уже есть доменное имя, необходимо настроить его на сервере хостинга, указав соответствующие DNS-записи. Если доменное имя еще не зарегистрировано, нужно зарегистрировать его и настроить на сервере.

6. Настройка базы данных: если сайт использует базу данных (например, для хранения информации о пользователях, контенте и т. д.), необходимо создать и настроить соответствующую базу данных на сервере хостинга.

7. Тестирование сайта: после размещения всех файлов и настройки сервера, следует провести тестирование сайта на сервере хостинга, чтобы убедиться, что все работает корректно.

Изм	Лист	№ документа	Подпись	Дата

WП.430000.000 ПЗ

Лист

27

Заключение

При разработке WEB – приложения были пройдены следующие этапы разработки:

1. Планирование проекта. Разработан план проекта, были определены основные технологии и архитектуры при разработке.
2. Разработка дизайна пользовательского интерфейса. Произведено создание макета пользовательского интерфейса, разработан дизайн страницы.
3. Разработка клиентской части приложения.
4. Разработка серверной часть приложения.
5. Тестирование.

Основные особенности реализации WEB - приложения включают в себя следующие аспекты: дизайн и пользовательский интерфейс, создание структуры базы данных, настройка серверных скриптов, разработка серверной логики, тестирование.

Ограничения созданного WEB – приложения заключается в отсутствии размещения на хостинг, слабо проработанного дизайна. В дальнейшем WEB – приложение можно хорошо модернизировать – гораздо лучше проработать дизайн, добавить некоторые скрипты, разместить сервер на хостинг, а также оптимизировать программный код.

В заключении необходимо сравнить готовый продукт с поставленными задачами, сделать вывод о полноте выполненного задания:

1. Создание веб-приложения с использованием HTML, CSS и JavaScript для клиентской части. – WEB – приложения создано. При создании использовались HTML, CSS и скрипты JavaScript.
2. Разработка серверной части приложения с использованием языка программирования PHP или иного языка. – При разработке серверной части WEB – приложения использовался именно язык PHP, реализованно достаточно функций взаимодействия с серверов.

3. Работа с базой данных MySQL (допускается применение других СУБД) для хранения информации. – Была использованна база данных для работы с данными пользователя, структурированием этих данных и корректным отображением как серверной, так и клиентской части созданного WEB – приложения.

4. Реализация функциональности веб-приложения, включая авторизацию и аутентификацию пользователей, обмен данными с сервером и отображение информации на странице. – Была реализованна авторизация и регистрация, с хранением данных в базе данных. Создан личный кабинет пользователя. Произведенна работа с формами для обмена данными сервера и клиента.

5. Тестирование и отладка приложения для обеспечения его корректной работы. – WEB – приложение было полностью протестированно, ошибок при работе с приложением не обнаружено.

					ВП.430000.000 ПЗ	Лист
						29
Изм	Лист	№ документа	Подпись	Дата		

Перечень используемых информационных ресурсов

1. PHP: Простой учебник. [Электронный ресурс]. URL: <https://www.php.net/manual/ru/tutorial.php> (дата обращения: 21.03.2024)
2. Самоучитель HTML [Электронный ресурс]. URL: <https://htmlbook.ru> (дата обращения: 20.03.2024)
3. Современный учебник JavaScript [Электронный ресурс]. URL: <https://learn.javascript.ru/> (дата обращения: 21.03.2023)
4. Самоучитель CSS [Электронный ресурс]. URL: <https://htmlbook.ru/samcss> (дата обращения: 19.03.2023).

Приложение А

```
<!DOCTYPE html>
<html lang="en">
<?php require_once("config.php");
ob_start();
?>
<link rel="stylesheet" href="styles.css">
<head>
<title>Страница на PHP</title>
    <?php
        require_once $_SERVER['DOCUMENT_ROOT'].'/head.php';
    ?>
</head>
<body link="#ffcc00" vlink="#cecece" alink="#ff0000">
    <div class="page">
        <div class="shapka">
            <?php
                require_once $_SERVER['DOCUMENT_ROOT'].'/top.php';
            ?>
        </div>
    </div>
    <p><a href="sk.php"></p></a>
    <div class="page">
        <div class="content1">
            <h3>Популярные товары: </h3>
            <div class="content1">
                <p></p>
```



```

        </div>
        <div class="content1">
            <p></p>
        </div>
        <div class="content1">
            <p></p>
        </div>
        <div class="content1">
            <p></p>
        </div>
        <br>
    </div>
    <div class="sidebar_left">
        <?php
        require_once $_SERVER['DOCUMENT_ROOT'].'/sidebar1.php';
        ?>
    </div>
</div>
<div class="footer">
    <?php
    require_once $_SERVER['DOCUMENT_ROOT'].'/footer.php';
    ?>
</div>
</body>
</html>

<div class="menu">
<a href="vhh.php">Каталог товаров</a>
</div>

```

```

<div class="counter">
<?php
session_start();
if (!isset($_SESSION['counter']))
{
    echo "Сегодня никто не посещал сайт";
    $_SESSION['counter']=0;
}
else
{
    $_SESSION['counter']++;
    echo "Посетителей: " . $_SESSION['counter'];
}
?>
</div>

<div class="log">
<a href="index.php">Главная</a>
</div>

<div class="log">
<a href="index.php">Главная</a>
</div>

<div class="log">
Ваш регион: <select name="region" id="input-select">
    <option value="Ростовская область">Ростовская область </option>
    <option value="Краснодарский край">Краснодарский край</option>
    <option value="Ставропольский край">Ставропольский край</option>
    <option value="Воронежская область">Воронежская область </option>
    <option value="Московская область">Московская область </option>

```

```

        <option value="Нижегородская область">Нижегородская область
    </option>
    <option value="Новгородская область">Новгородская область </option>
    <option value="Тверская область">Тверская область </option>
    <option value="Москва">Москва</option>
    <option value="Санкт-Петербург">Санкт-Петербург</option>
    <option value="Нижний Новгород">Нижний Новгород</option>
    <option value="Краснодар">Краснодар</option>
    <option value="Ставрополь">Ставрополь</option>
</select>
</div>
<div class="vx">
<a href="vhh.php">Вход</a>
</div>
<div class="reg">
<a href="reg.php">Регистрация</a>
</div>
<div class="contact">
Доставка
</div>
<div class="footer_contact">
<h3>Контакты</h3>
<a href="index.php">Адреса магазинов</a>
+7(969)837-18-71<p>
<a href="index.php">О нас</a><p>
<a href="index.php">Обратная связь</a>
<p></p>© Shoes Company, 2024 г.
</div>
<!DOCTYPE html>

```

```

<html lang="en">

<link rel="stylesheet" href="styles.css">

<head>

<title>Страница на PHP</title>

<p></p>

    <?php
        require_once $_SERVER['DOCUMENT_ROOT'].'/head.php';
    ?>

</head>

<body>

    <div class="shapka">

        <?php
            require_once $_SERVER['DOCUMENT_ROOT'].'/top.php';
        ?>

    </div>

    <div class="content1">

        <p></p>

    </div>

</div>

<div class="footer">

    <?php
        require_once $_SERVER['DOCUMENT_ROOT'].'/footer.php';
    ?>

</div>

</body>

</html>

```

```

<!DOCTYPE html>
<html lang="en">
<?php require_once("config.php");
ob_start();
?>
<link rel="stylesheet" href="styles.css">
<head>
<title>Страница на PHP</title>
    <?php
        require_once $_SERVER['DOCUMENT_ROOT'].'/head.php';
    ?>
</head>
<body>
    <div class="content1">
        <form method="post" action="<?php echo $_SERVER['PHP_SELF'];
?>">
        <label for="login">Логин:</label><br>
        <input type="text" id="login" name="login"><br><br>

        <label for="pass">Пароль:</label><br>
        <input type="password" id="pass" name="pass"><br><br>
        <input type="submit" value="Войти">
    </form>
    <?php
        if (isset($_POST['login'])) {
            $pass = $_POST['pass'];
            $sql = "SELECT * FROM user WHERE pass = $pass";
            $result = $conDB->query($sql);
            if ($result->num_rows > 0) {

```

Изм	Лист	№ документа	Подпись	Дата

ВП.430000.000 ПЗ

Лист

36

```

echo '<a href="index1.php"'>Добро пожаловать</a>';
}
}
?>

<button><a href="index.php">На главную</a></button>

</div>

</body>

</html>

<!DOCTYPE html>

<html lang="en">

<?php require_once("config.php");
ob_start();
?>

<link rel="stylesheet" href="styles.css">

<head>

<title>Страница на PHP</title>

    <?php
        require_once $_SERVER['DOCUMENT_ROOT'].'/head.php';
    ?>

</head>

<body>

    <div class="content1">

        <form method="post" action="<?php echo $_SERVER['PHP_SELF'];
?>">

        <label for="login">Логин:</label><br>

        <input type="text" id="login" name="login"><br><br>

        <label for="pass">Пароль:</label><br>

        <input type="password" id="pass" name="pass"><br><br>

```

Изм	Лист	№ документа	Подпись	Дата

```

<input type="submit" value="Зарегистрироваться">
</form>

<?php
$login = $_POST['login'];
$pass= $_POST['pass'];
$sql = "INSERT INTO user (login, pass) VALUES ('$login', '$pass')";

if ($conDB->query($sql) === TRUE)
?>

<button><a href="vhh.php">Войти</a></button>

</div>

</body>

</html>

<?php
    $nameDB = "sait";
    $nameSERVER = "localhost";
    $nameUSER = "root";
    $passUSER = "root";

    $conDB = mysqli_connect($nameSERVER, $nameUSER, $passUSER,
$nameDB) or die("ERROR".mysqli_error($nameDB));

    mysqli_query($conDB, "SET NAMES utf8");

?>

<!DOCTYPE html>
<html lang="en">
<?php require_once("config.php");
ob_start();

```

```

?>
<link rel="stylesheet" href="styles.css">
<head>
<title>Страница на PHP</title>
    <?php
        require_once $_SERVER['DOCUMENT_ROOT'].'/head.php';
    ?>
</head>
<body link="#ffcc00" vlink="#cecece" alink="#ff0000">
    <div class="page">
        <div class="shapka">
            <?php
                require_once
$_SERVER['DOCUMENT_ROOT'].'/topkattov.php';
            ?>
        </div>
    </div>
    <p></p>
    <div class="content2">
        <h3>Выберите интересующие Вас товары:</h3>
        <form action="kattov.php" method="post">
            <label for="nike">Nike</label>
            <input type="radio" id="nike" name="nike"><br><br>
            <label for="adi">Adidas</label>
            <input type="radio" id="adi" name="adi"><br><br>
            <label for="dior">Dior</label>
            <input type="radio" id="dior" name="dior"><br><br>
            <label for="lv">Luis Vuitton</label>
            <input type="radio" id="lv" name="lv"><br><br>

```



```

<input type="submit" value="Показать">
</form>

<?php
if(isset($_POST['nike'])) {
    $sql = "SELECT * FROM tov WHERE name = 'nike'";
    $result = $conDB->query($sql);
    while($row = $result->fetch_assoc()) {
    if ($result->num_rows > 0) {
        echo '<div>' . $row["nazv"] . '<br>' . $row["izo"] . '<br>' . "Количество " .
        $row["kol-vo"] . '<br>' . '<button>Добавить в корзину</button> <br>' . '</div>' .
        '<br>';}} }
    if(isset($_POST['adi'])) {
        $sql = "SELECT * FROM tov WHERE name = 'adi'";
        $result = $conDB->query($sql);
        while($row = $result->fetch_assoc()) {
        if ($result->num_rows > 0) {
            echo '<div>' . $row["nazv"] . '<br>' . $row["izo"] . '<br>' . "Количество " .
            $row["kol-vo"] . '<br>' . '<button>Добавить в корзину</button> <br>' . '</div>' .
            '<br>';}} }
        if(isset($_POST['dior'])) {
            $sql = "SELECT * FROM tov WHERE name = 'dior'";
            $result = $conDB->query($sql);
            while($row = $result->fetch_assoc()) {
            if ($result->num_rows > 0) {
                echo '<div>' . $row["nazv"] . '<br>' . $row["izo"] . '<br>' . "Количество " .
                $row["kol-vo"] . '<br>' . '<button>Добавить в корзину</button> <br>' . '</div>' .
                '<br>';}} }
            if(isset($_POST['lv'])) {
                $sql = "SELECT * FROM tov WHERE name = 'lv'";

```

					ВП.430000.000 ПЗ	Лист
						40
Изм	Лист	№ документа	Подпись	Дата		

```

$result = $conDB->query($sql);
while($row = $result->fetch_assoc()) {
if ($result->num_rows > 0) {
    echo '<div>' . $row["nazv"] . '<br>' . $row["izo"] . '<br>' . "Количество " .
    $row["kol-vo"] . '<br>' . '<button>Добавить в корзину</button> <br>' . '</div>' .
    '<br>';}}
?>

</div>

</div>

<div class="footer">
    <?php
    require_once $_SERVER['DOCUMENT_ROOT'].'/footer.php';
    ?>
</div>
</body>
</html>

p {
    color: white;
}
h1 {
    color: white;
}
h3 {
    color: white;
}
body {
    background-image: url(background.png);
    color: rgb(0, 0, 0);
}

```

```
padding: 10px;  
}
```

```
div {  
    background-image: url(background.png);  
    color: white;  
    border-radius: 5px;  
    margin: 5px 5px 5px 5px;  
    padding: 3px;  
}
```

```
.sidebar_left {  
    padding: 5px;  
    width: 20%;  
}
```

```
.content{  
    color: white;  
    width: 80%;  
    display: flex;  
    justify-content: center;  
    padding: 5px;  
    text-align: center;  
}
```

```
.content1 {  
    color: white;  
    width: 100%;  
    display: flex;
```

					ВП.430000.000 ПЗ	Лист
						42
Изм	Лист	№ документа	Подпись	Дата		

```

    justify-content: center;
    padding: 5px;
}

```

```

.page {
    display: flex;
    justify-content: center;
    padding: 1px;
}

```

```

.shapka {
    display: flex;
    justify-content:left;
    padding: 5px;
}

```

```

button {
    border-radius: 5px;
}

```

```

href {
    color: white;
}

```

```

.content2{
    color: white;
    width: 100%;
    display: flex;
    justify-content: left;
    padding: 5px;
}

```

					WП.430000.000 ПЗ	Лист
						43
Изм	Лист	№ документа	Подпись	Дата		

```
.okno {  
    background: #8A2BE2;  
    width: 300px;  
    height: 50px;  
    text-align: center;  
    padding: 15px;  
    border: 3px solid purple;  
    border-radius: 10px;  
    color: #8A2BE2;  
}
```

					ВП.430000.000 ПЗ	Лист
						44
Изм	Лист	№ документа	Подпись	Дата		