Частное профессиональное образовательное учреждение

«БАШКИРСКИЙ ЭКОНОМИКО-ЮРИДИЧЕСКИЙ КОЛЛЕДЖ» (БЭК)

Профессия: 09.02.07 Информационные системы и программирование

**ИНДИВИДУАЛЬНЫЙ ПРОЕКТ**

на тему

**«Исследование способов создания сайта и сопровождения его в сети интернет»**

по учебной дисциплине

«ПД 03. Информатика»

Студент: Вагизов Данис Равкатович

( ФИО полностью)

Курс 1 группа ИСП 23-19

Руководитель: Серикова Анастасия Викторовна

( ФИО полностью)

*\_\_\_\_\_\_\_* «*\_\_\_»*\_\_\_\_\_\_\_\_\_\_\_\_20\_\_г.

подпись (дата)

Оренбург, 20 \_ г.

**СОДЕРЖАНИЕ**

|  |  |  |
| --- | --- | --- |
|  | [ВВЕДЕНИЕ](#ВВЕДЕНИЕ)**…………………………………………………………** | **3** |
| **1** | Роль веб-сайтов и взаимодействие пользователя с веб-сайтом**……………………………………………………………….** | **5** |
| **2** | Структура веб-сайтов**………………………………………………** | **6** |
| **3** | Создание сайта**……………………………………………………..** | **7** |
| **4** | Выбор фреймворка для создания сайта**…………………………..** | **8** |
| **5** | Как создаются сайты с помощью фреймворка**…………………..** | **9** |
| **6** | **Ведение сайта в интернете………………………………………...** | **10** |
|  | [ЗАКЛЮЧЕНИЕ](#ЗАКЛЮЧЕНИЕ)**…………………………………………………….** | **12** |
|  | [СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ](#СПИСОК)**………………….** | **13** |

**ВВЕДЕНИЕ**

В современном мире, где информационные технологии играют ключевую роль во всех сферах жизни, веб-сайты стали неотъемлемой частью нашего общества. Они служат важными инструментами для обмена информацией, предоставления услуг, продвижения бизнеса и многого другого. В связи с этим, способность создавать и поддерживать веб-сайты стала ценным навыком, который требуется во многих профессиях.

Тема данного проекта, «Исследование способов создания сайтов и сопровождение его в сети интернет», является актуальной и значимой, поскольку она затрагивает важные аспекты информатики и веб-разработки. В рамках этого проекта мы исследуем различные методы и технологии, используемые для создания веб-сайтов, а также рассмотрим вопросы их поддержки и обслуживания в интернете.

Цель данного проекта - провести глубокое исследование различных подходов к созданию веб-сайтов, включая использование HTML/CSS, CMS (таких как WordPress, Joomla), и различных фреймворков (таких как Django, Flask, Ruby on Rails). Мы также изучим вопросы развертывания и поддержки веб-сайтов в интернете, включая вопросы безопасности, оптимизации и обновления контента.

В ходе работы над проектом мы надеемся получить ценные знания и практические навыки в области веб-разработки, которые будут полезны не только для учебных целей, но и для будущей профессиональной деятельности. Этот проект также поможет нам лучше понять важность и сложность работы веб-разработчика, а также оценить вклад информатики в развитие современного общества.

В дополнение к уже указанному, стоит отметить, что веб-разработка является динамично развивающейся областью. Новые технологии и подходы появляются постоянно, и важно быть в курсе этих изменений. В рамках данного проекта мы также постараемся уделить внимание актуальным трендам и инновациям в области веб-разработки.

Кроме того, мы рассмотрим вопросы доступности и удобства использования веб-сайтов. Создание веб-сайта - это не только технический процесс, но и процесс дизайна, который должен учитывать потребности и ожидания пользователей. Мы исследуем принципы создания удобных и доступных веб-сайтов и рассмотрим, как эти принципы можно применить на практике.

Таким образом, данный проект представляет собой комплексное исследование в области веб-разработки, которое поможет нам не только разобраться в технических аспектах создания веб-сайтов, но и лучше понять социальные, этические и дизайнерские аспекты этой работы.

1. **Роль веб-сайтов и взаимодействие пользователя с веб-сайтом**

Веб-сайты играют ключевую роль в информационных технологиях, выполняя множество функций. Они служат платформами для обмена информацией, предоставления услуг, продвижения бизнеса, образования и многого другого. Веб-сайты могут быть использованы для создания онлайн-магазинов, блогов, новостных порталов, образовательных ресурсов, социальных сетей и многого другого.

Веб-сайты также могут включать в себя скрипты на языке JavaScript, который используется для добавления интерактивности на веб-страницы. Кроме того, веб-сайты могут использовать различные серверные технологии и базы данных для обработки данных и предоставления динамического контента.

Когда пользователь заходит на веб-сайт, он видит веб-страницу, которая отображается в его веб-браузере. Эта веб-страница может включать в себя текст, изображения, видео, аудио и другие элементы. Пользователь может взаимодействовать с веб-сайтом, кликая по ссылкам и кнопкам, заполняя формы и выполняя другие действия.

Важно отметить, что внешний вид веб-сайта и его функциональность могут варьироваться в зависимости от используемого веб-браузера, устройства и настроек пользователя. Веб-разработчики должны учитывать эти факторы при создании веб-сайтов, чтобы обеспечить хороший пользовательский опыт для всех пользователей.

Веб-сайты играют важную роль в организации информационно-технологического сопровождения различных сфер деятельности. Они служат средством информирования и взаимодействия с различными субъектами, например, в образовательном процессе современного вуза. Веб-сайты могут служить платформой для публикации информации о курсах, программе обучения, научных исследованиях, мероприятиях, новостях и других материалах. Это позволяет пользователям быстро и легко получать доступ к необходимой информации.

1. **Структура веб-сайтов**

Веб-сайт состоит из множества компонентов. Основные из них - это веб-страницы, которые создаются с использованием языков разметки, таких как HTML и CSS. HTML (HyperText Markup Language) используется для создания структуры веб-страницы, в то время как CSS (Cascading Style Sheets) используется для определения стиля элементов на веб-странице.

Веб-сайты состоят из множества компонентов. Основные из них - это хедер и меню, визуальная часть, контентная часть и футер. Хедер, располагающийся в самом верху страницы, обычно включает в себя логотип и верхнее меню, которое помогает посетителям с навигацией по сайту. Визуальная часть следует сразу после хедера и обычно содержит изображение, серию изображений или видео. Контентная часть следует сразу после обложки и содержит более подробную информацию о владельце сайта, бизнесе или организации. Футер, располагающийся в самой нижней части сайта, обычно содержит уведомления об авторских правах или контактную информацию.

Взаимодействие с пользователем играет ключевую роль в современном мире веб-разработки и дизайна. Это процесс обмена информацией между пользователем и веб-сайтом или приложением, позволяющий пользователям взаимодействовать с контентом и выполнять различные действия. Пользователь может взаимодействовать с веб-сайтом, кликая по ссылкам и кнопкам, заполняя формы и выполняя другие действия. Важно отметить, что внешний вид веб-сайта и его функциональность могут варьироваться в зависимости от используемого веб-браузера, устройства и настроек пользователя. Веб-разработчики должны учитывать эти факторы при создании веб-сайтов, чтобы обеспечить хороший пользовательский опыт для всех пользователей.

1. **Создание сайта**

Существует несколько подходов к созданию веб-сайтов, каждый из которых имеет свои преимущества и недостатки.

Использование HTML/CSS: Это базовый подход, который требует написания кода вручную. HTML используется для создания структуры веб-страницы, а CSS - для определения ее внешнего вида. Этот подход дает большую гибкость и контроль над внешним видом и функциональностью сайта, но требует знаний языков программирования.

Использование CMS (Система Управления Контентом): CMS - это программное обеспечение, которое позволяет создавать и управлять веб-сайтом без необходимости написания кода. Существуют различные CMS, такие как WordPress, Joomla, Drupal, которые предлагают широкий спектр функций и возможностей. CMS обычно предлагают готовые шаблоны дизайна, инструменты для управления контентом и возможности расширения функциональности с помощью плагинов.

Использование конструкторов сайтов: Конструкторы сайтов - это онлайн-сервисы, которые предлагают простой способ создать веб-сайт с помощью графического интерфейса. Они предлагают готовые шаблоны дизайна, инструменты для добавления и редактирования контента, а также возможности для настройки внешнего вида и функциональности сайта.

HTML (HyperText Markup Language) и CSS (Cascading Style Sheets) - это два основных языка, используемых для создания веб-сайтов.

HTML используется для создания структуры веб-страницы, включая размещение текста, изображений, ссылок и других элементов. CSS, с другой стороны, используется для определения внешнего вида этих элементов, включая цвет, размер шрифта, расположение и многое другое.

CMS (Content Management System) - это программное обеспечение, которое позволяет создавать и управлять веб-сайтом без необходимости написания кода. С помощью CMS можно легко добавлять и редактировать контент, изменять внешний вид сайта, добавлять новые функции и многое другое.

Существуют различные CMS, каждая из которых предлагает свой набор функций и возможностей. Некоторые из наиболее популярных CMS включают WordPress, Joomla и Drupal. Каждая из этих систем имеет свои преимущества и недостатки, и выбор подходящей системы зависит от конкретных потребностей и требований проекта.

1. **Выбор фреймворка для создания сайта**

Фреймворк - это набор инструментов и библиотек, которые облегчают разработку веб-сайтов, предоставляя структуру и набор стандартных функций. Вот несколько популярных фреймворков:

1. **Django**: Django - это высокоуровневый фреймворк для разработки веб-приложений на Python. Он следует принципу “Batteries included”, что означает, что он предоставляет все необходимые инструменты для разработки веб-приложения прямо из коробки. Django идеально подходит для разработки сложных веб-приложений, таких как системы управления контентом (CMS) или социальные сети.
2. **Flask**: Flask - это еще один фреймворк на Python, но он гораздо более легкий и гибкий, чем Django. Flask следует принципу “micro”, что означает, что он предоставляет только базовую функциональность, позволяя разработчикам выбирать дополнительные инструменты и библиотеки по мере необходимости. Flask идеально подходит для разработки небольших веб-приложений или микросервисов.
3. **Ruby on Rails**: Ruby on Rails, часто просто Rails, - это фреймворк для разработки веб-приложений на Ruby. Rails следует принципу “Convention over Configuration”, что означает, что он предпочитает соглашениям по умолчанию, чтобы уменьшить количество решений, которые должен принять разработчик. Rails идеально подходит для быстрой разработки веб-приложений, благодаря своей интеграции с базами данных и поддержке автоматизированного тестирования.

Выбор фреймворка зависит от многих факторов, включая требования к проекту, опыт команды и предпочтения разработчиков. Важно помнить, что нет “лучшего” фреймворка - каждый из них имеет свои сильные и слабые стороны, и выбор должен основываться на конкретных потребностях проекта.

1. **Как создаются сайты с помощью фреймворка**

Рассмотрим, как создаются веб-сайты с использованием фреймворков.

**1. Django**

Django следует архитектурному шаблону Model-View-Template (MVT). Вот пример базового представления Django:

from django.http import HttpResponse

from django.shortcuts import render

def home(request):

return HttpResponse("Привет, мир!"

В этом примере home - это функция представления, которая принимает объект запроса и возвращает объект ответа. Это самый базовый пример представления в Django.

**2. Flask**

Flask - это микрофреймворк, который не предоставляет столько готовых решений, как Django, но предлагает больше гибкости. Вот пример базового приложения Flask:

from flask import Flask

app = Flask(\_\_name\_\_)

@app.route('/')

def home():

return "Привет, мир!"

В этом примере home - это функция представления, которая связана с URL-адресом '/' с помощью декоратора @app.route. Когда пользователь посещает этот URL-адрес, Flask вызывает функцию home и возвращает ее результат.

**3. Ruby on Rails**

Ruby on Rails, или просто Rails, использует шаблон Model-View-Controller (MVC). Вот пример контроллера Rails:

class WelcomeController < ApplicationController

def index

render plain: "Привет, мир!"

end

end

В этом примере index - это действие контроллера, которое вызывается, когда пользователь посещает соответствующий URL-адрес. Это действие затем возвращает текст “Привет, мир!”.

Это лишь базовые примеры того, как можно создать веб-сайт с использованием этих фреймворков. В реальном проекте вам, вероятно, придется работать с базами данных, шаблонами, формами, аутентификацией и многим другим. Каждый из этих фреймворков предлагает инструменты и библиотеки для работы с этими аспектами разработки веб-сайтов.

1. **Ведение сайта в интернете**

Ведение сайта в интернете - это многоаспектный процесс, который включает в себя несколько ключевых элементов.

Публикация сайта: После того, как вы закончите писать код и организовывать файлы, которые составляют ваш веб-сайт, вам нужно расположить все это в Интернете, чтобы люди могли найти ваш сайт. Для этого вам потребуется хостинг и доменное имя. Хостинг - это арендованное файловое пространство на веб-сервере хостинговой компании, где вы размещаете ваши файлы веб-сайта. Доменное имя - это уникальный адрес, по которому люди могут найти ваш веб-сайт.

SEO (Search Engine Optimization): SEO - это процесс оптимизации вашего веб-сайта, чтобы улучшить его видимость в результатах поиска. Это включает в себя использование ключевых слов, создание качественного контента, получение обратных ссылок и многое другое.

SMM (Social Media Marketing): SMM - это использование социальных сетей для продвижения вашего веб-сайта. Это может включать в себя создание контента, который можно поделиться в социальных сетях, взаимодействие с вашей аудиторией и многое другое.

Контекстная реклама: Контекстная реклама - это форма таргетированной рекламы, которая отображается пользователям на основе их интересов или поведения в интернете.

Управление контентом: Управление контентом на вашем веб-сайте включает в себя создание, редактирование и обновление контента. Это может включать в себя написание и публикацию новых статей, обновление информации о продуктах или услугах, добавление новых изображений и многое другое.

Мониторинг и аналитика: Мониторинг и аналитика включают в себя отслеживание и анализ данных о том, как пользователи взаимодействуют с вашим веб-сайтом. Это может помочь вам понять, какие части вашего веб-сайта работают хорошо, а какие требуют улучшения.

Важно отметить, что успешное ведение веб-сайта требует постоянного усилия и внимания. Вам нужно регулярно обновлять свой контент, следить за последними трендами в SEO и SMM, анализировать данные о посещаемости вашего сайта и вносить необходимые изменения.

**ЗАКЛЮЧЕНИЕ**

В ходе выполнения данного проекта мы провели глубокое исследование в области создания и поддержки веб-сайтов. Мы рассмотрели различные подходы к созданию веб-сайтов, включая использование HTML/CSS, CMS и конструкторов сайтов. Мы также изучили важность веб-сайтов в информационных технологиях и рассмотрели, как они используются для обмена информацией, предоставления услуг, продвижения бизнеса и многого другого.

Кроме того, мы обсудили вопросы ведения веб-сайта в интернете, включая публикацию сайта, SEO, SMM, контекстную рекламу, управление контентом и мониторинг и аналитику. Мы увидели, что успешное ведение веб-сайта требует постоянного усилия и внимания, и важно быть в курсе последних трендов и изменений в этой области.

В результате нашего исследования мы получили ценные знания и практические навыки в области веб-разработки, которые будут полезны не только для учебных целей, но и для будущей профессиональной деятельности. Мы надеемся, что результаты нашего исследования будут полезны и интересны не только нам, но и всем, кто интересуется веб-разработкой и информатикой в целом.

В заключение хотелось бы отметить, что веб-разработка - это динамично развивающаяся область, и всегда есть что-то новое и интересное, что можно узнать и исследовать. Мы с нетерпением ждем возможности продолжить изучение этой увлекательной темы в будущих проектах.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

I.Периодические издания

1. Ганиева А.Д. Роль веб-сайтов в учебной деятельности вуза./ Ганиева А.Д.// ЖУРНАЛ «НАУЧНЫЙ ЛИДЕР».-2023.-№8.- С. 19-20.
2. Никитина О.О. Роль веб-сайта как главного коммуникационного канала компании в современном контексте./ Никитина О.О.//Научно-исследовательский журнал «Экономические исследование и разработки».-2022.

II.Электронные ресурсы:

1. Взаимодействие пользователя с сайтом: пользовательские задание и ведущие методы-что это такое? [Электронный ресурс].- Режим доступа: <https://askusers.ru/blog/pravila/profili-zadach-v-ux/> (Дата обращения 26.04.2024).
2. Виды сайтов, их структура и типа: полная классификация с примерами- Market-makers.- [Электронный ресурс].- Режим доступа: <https://market-makers.org/виды-сайтов/> (Дата обращения 20.04.2024).
3. Всемирная сеть (веб) и веб-стандарты-Изучение веб-разработки. [Электронный ресурс].Режим доступа:https://developer.mozilla.org/ru/docs/Learn/Getting\_started\_with\_the\_web/The\_web\_and\_web\_standards (Дата обращения 25.04.2024).
4. Интернет для людей, а не для прибыли.- [Электронный ресурс].- Режим доступа: <https://www.mozilla.org/ru/> (Дата обращения 21.04.2024).
5. Как создать сайт: 14 простых шагов с картинками.- [Электронный ресурс].- Режим доступа: <https://www.pixpa.com/ru/blog/how-to-design-a-website> (Дата обращения 27.04.2024).
6. Как создать свой сайт в 2024: полное пошаговое руководство. -[Электронный ресурс].-  Режим доступа:<https://www.websiteplanet.com/ru/blog/как-создать-свой-сайт/> (Дата обращения 19.04.2024).
7. Основные подходы к созданию сайта.- [Электронный ресурс].- Режим доступа: <https://infourok.ru/osnovnye-podhody-k-sozdaniyu-sajta-4823316.html> (Дата обращения 24.04.2024).
8. Основы способа взаимодействия с пользователем: что это? [Электронный ресурс].- Режим доступа: <https://zvenst.ru/osnovnye-principy-vzaimodeistviya-s-polzovatelem-v-sistemax/> (Дата обращения 28.04.2024).
9. Основы CSS - Изучение веб-разработки | MDN.- [Электронный ресурс].- Режим доступа:<https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/CSS_basics> (Дата обращения 15.04.2024).
10. Правильная структура сайта: классические составляющие и фишки.- [Электронный ресурс].- Режим доступа:https://envybox.io/blog/struktura-sajta/ (Дата обращения 18.04.2024).
11. Система управления контентом.- [Электронный ресурс].- Режим доступа: <https://en.wikipedia.org/wiki/Content_management_system> (Дата обращения 20.04.2024).
12. Система управления содержимым.- [Электронный ресурс].- Режим доступа: <https://ru.wikipedia.org/wiki/Система_управления_содержимым> (Дата обращения 21.04.2024).
13. Современные подходы к созданию веб-сайтов (начало) - Персональный блог.- [Электронный ресурс].- Режим доступа: <https://igorosa.com/sovremennye-podxody-k-sozdaniyu-veb-sajtov-nachalo/> (Дата обращения 30.04.2024).
14. Структура документа и веб-сайта. [Электронный ресурс].- Режим доступа:https://developer.mozilla.org/ru/docs/Learn/HTML/Introduction\_to\_HTML/Document\_and\_website\_structure (Дата обращения 29.04.2024).
15. Структура сайта: полное руководство.- [Электронный ресурс].- Режим доступа: <https://ru.wix.com/blog/2021/06/iz-chego-sostoit-sayt> (Дата обращения 27.04.2024).
16. Что такое CMS (система управления контентом): назначение и список лучших движков.- [Электронный ресурс].- Режим доступа: <https://timeweb.com/ru/community/articles/chto-takoe-cms> (Дата обращения 24.04.2024).
17. Ширин Д.А. Роль веб-сайта в организации информационно-технологического сопровождения субъектов образовательного процесса современного вуза. [Электронный ресурс] //[Вестник Новгородского государственного университета им. Ярослава Мудрого](https://cyberleninka.ru/journal/n/vestnik-novgorodskogo-gosudarstvennogo-universiteta-im-yaroslava-mudrogo).- Режим доступа: https://cyberleninka.ru/article/n/rol-veb-sayta-v-organizatsii-informatsionno-tehnologicheskogo-soprovozhdeniya-subektov-obrazovatelnogo-protsessa-sovremennogo-vuza (Дата обращения 30.04.2024).
18. 14 советов, как улучшить взаимодействие с пользователем на веб-сайте.- [Электронный ресурс].- Режим доступа: <https://k-graphiste.com/ru/Блог/Пользовательский-опыт/> (Дата обращения 23.04.2024).
19. 7 способов оптимизировать взаимодействие с пользователем на вашем сайте.- [Электронный ресурс].- Режим доступа: <https://coffee-web.ru/blog/7-ways-to-optimize-user-experience-on-your-website/> (Дата обращения 29.04.2024).
20. HTML Styles CSS.- [Электронный ресурс].- Режим доступа: <https://www.w3schools.com/html/html_css.asp> (Дата обращения 22.04.2024).