Министерство образования Республики Беларусь

Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

Факультет информационных технологий и управления

Кафедра информационных технологий автоматизированных систем

ОТЧЕТ

По лабораторной работе №1

«**ОСНОВЫ *HTML* И *CSS***»

Выполнил: ст. гр. 220603

Д. М. Василевич

Проверил: А. Л. Гончаревич

Минск 2025

СОДЕРЖАНИЕ

[Введение 3](#_Toc189856832)

[1 Постановка задачи 4](#_Toc189856833)

[2 Теоретическая часть 5](#_Toc189856834)

[3 Ход работы 7](#_Toc189856835)

[Заключение 8](#_Toc189856836)

ВВЕДЕНИЕ

В современной веб-разработке знание основ *HTML* и *CSS* является ключевым навыком для создания сайтов и веб-приложений. *HTML* (*HyperText* *Markup* *Language*) отвечает за структуру и содержимое веб-страницы, включая заголовки, текст, изображения, таблицы и ссылки. *CSS* (*Cascading* *Style* *Sheets*), в свою очередь, управляет внешним видом и стилем этих элементов, позволяя изменять цвет, размер, расположение и другие визуальные аспекты.

Данная лабораторная работа посвящена изучению основ *HTML* и *CSS*. В рамках задания студенты познакомятся с базовыми элементами *HTML*, научатся структурировать веб-страницы и добавлять текстовые и графические элементы. Также будет изучено применение *CSS* для стилизации страниц, включая работу с цветами, шрифтами, отступами и выравниванием.

Целью данной лабораторной работы является освоение базовых инструментов создания веб-страниц и формирование навыков их стилизации. Выполнение заданий позволит понять, как связаны структура и оформление веб-документов, а также заложит фундамент для изучения более сложных аспектов веб-разработки.

1 ПОСТАНОВКА ЗАДАЧИ

Необходимо разработать многостраничный сайт, демонстрирующий практическое применение *HTML* и *CSS*, а также использование системы контроля версий *Git*.

В процессе работы необходимо сформировать концепцию сайта, определить его тематику, структуру, стилизацию и основную идею проекта. Сайт должен включать главную страницу и несколько дополнительных, таких как «Главная», «Меню» и «Корзина» и т.д. Это обеспечит логичное распределение контента и удобную навигацию. Особое внимание уделяется продумыванию структуры и оформлению: визуальное представление сайта должно быть реализовано с помощью *CSS*, включающего выбор цветовой гаммы, шрифтов, отступов и выравнивания элементов для создания гармоничного и удобного интерфейса.

После этапа разработки сайт необходимо протестировать на различных браузерах и устройствах для проверки корректности работы всех ссылок, функциональности элементов и адаптивности дизайна.

Помимо этого, обязательным является использование системы контроля версий *Git* для отслеживания изменений, организации работы с ветками и ведения коммитов, что позволит обеспечить прозрачность и структурированность процесса разработки. После завершения работы предоставить веб-сайт для демонстрации проделанной работы.

2 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

*HTML* (*HyperText* *Markup* *Language*) и *CSS* (*Cascading* *Style* *Sheets*) являются основными технологиями для создания веб-страниц. *HTML* отвечает за структуру веб-документа, определяя, какие элементы будут на странице: заголовки, абзацы, изображения, таблицы и ссылки. *CSS* используется для оформления этих элементов, позволяя задавать цвета, шрифты, размеры и другие стилистические характеристики. *HTML*-документ состоит из иерархии тегов, каждый из которых определяет определённый элемент страницы.

Теги обрамляются угловыми скобками и, как правило, имеют открывающую и закрывающую часть. Например, <*h*1> используется для заголовков первого уровня, а <*p*> — для абзацев. Ключевыми элементами *HTML* являются элементы текста, такие как <*h*1>, <*h*2>, <*p*>, <*span*>, списки, например <*ol*> для упорядоченных списков и <*ul*> для неупорядоченных, ссылки <*a*>, изображения <*img*>, а также таблицы, создаваемые с помощью <*table*>, <*tr*>, <*td*>. Для построения корректного *HTML*-документа важно соблюдать структуру, начинающуюся с декларации <!*DOCTYPE* *html*>, после которой следуют теги <*html*>, <*head*> и <*body*>.

*CSS* позволяет стилизовать элементы *HTML*. Стили могут быть встроенными через атрибут *style*, внутренними в разделе <*style*> внутри *HTML*-документа или внешними в отдельном файле с расширением .*css*. Основные свойства *CSS* включают цвета и фоны (*color*, *background*-*color*, *background*-*image*), шрифты и текст (*font*-*family*, *font*-*size*, *font*-*weight*, *text*-*align*), границы и отступы (*border*, *margin*, *padding*), а также размеры и положение элементов (*width*, *height*, *display*, *position*). *CSS*-правила состоят из селектора и набора свойств.

Для обеспечения удобного отображения на устройствах с разными размерами экранов используется адаптивный дизайн. Это достигается с помощью медиазапросов.

В качестве удобного инструмента для совместной разработки выбран *Git*. *Git* — это система контроля версий, которая помогает отслеживать изменения в коде и эффективно управлять проектом. Основные команды *Git* включают инициализацию репозитория, добавления файлов в индекс, сохранения изменений, отправка изменений на удалённый репозиторий, получение изменений из удалённого репозитория и управление ветками. Использование *Git* позволяет вести историю изменений и, в случае надобности, быстро откатываться к предыдущим версиям проекта.

*HTML* и *CSS* формируют базовые инструменты для создания веб-страниц, а *Git* является удобным инструментом для управления проектами.

3 ХОД РАБОТЫ

ЗАКЛЮЧЕНИЕ

В ходе выполнения лабораторной работы разработан веб-сайт по предложенной тематике «Сайт для доставки еды».

Сайт содержит следующие веб-страницы:

– главная страница;

– страница «Меню»;

– страница «Специальные предложения»;

– страница «Как заказать»;

– страница «Контакты»;

– страница «Корзина»;

Итогом создания веб-страниц стало получение базовых навыков владения *HTML*. Изучены основы стилизации с помощью *CSS*, включая работу с цветами, шрифтами, отступами. Применены техники гибкой и адаптивной верстки интерфейса под различные устройства пользователя.

Веб-сайт располагается на удаленном репозитории *GitHub*. Сервис использовался как среда, в которой каждый из разработчиков, имея собственную ветку для разработки, мог вносить изменения, не меняя работу существующего продукта. Исходом являлось слияние наработок разработчика с главной веткой веб-сайта для дальнейшего выпуска продукта.