Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет информационных технологий и управления

Кафедра информационных технологий автоматизированных систем

Отчёт  
по лабораторной работе №1

«Работа с *HTML* и *CSS*»

|  |  |
| --- | --- |
| Проверил | Выполнил |
| Гончаревич А. Л. | ст. гр. №820602 |
|  | А. Д. Бондарчук |

Минск 2021

**СОДЕРЖАНИЕ**

Введение 3

1 Цель работы 4

2 Теоретическая часть 5

2.1 Краткие сведения о *HTML* и *CSS* 5

2.2 Описание используемых тегов 5

3 Практическая часть 7

3.1 Описания назначений каждой страницы 7

3.2 Ход работы 7

3.3 Руководство пользователю 7

Заключение 8

Список использованных источников 9

# Введение

Вёрсткой веб-страниц называется создание такого *HTML*-кода, который позволяет размещать элементы веб-страницы (изображения, текст, линии и т.д.) в нужных местах документа и отображать их в окне браузера согласно разработанному макету. При этом следует принимать во внимание ограничения присущие *HTML* и *CSS*, учитывать особенности браузеров и знать приёмы вёрстки, которые дают желаемый результат.

Вёрстка – это процесс творческий и чётких алгоритмов здесь не существует.

Любой сайт – это комплексный продукт. Он должен быть эстетичным и привлекательным в плане дизайна, содержать интересную и полезную информацию. В то же время он должен легко индексироваться поисковиками, быстро загружаться, а для автора быть удобным для редактирования и расширения возможностей. Но это ещё не всё. Сайт должен без явных ошибок отображаться в разных браузерах, подстраиваться под размер окна.

*CSS* расширил возможности *HTML* по работе с текстом на веб-странице, но они ещё уступают развитым программам вёрстки полиграфических материалов. Поэтому приходится знать атрибуты *CSS* относящиеся к вёрстке текста, особенности их применения, существующие ограничения и способы их обхода. А это уже задача вёрстки.

Все подходы к вёрстке имеют как преимущества, так и недостатки. Верстальщик как специалист сам выбирает, какие инструменты использовать, основываясь на взвешивании достоинств и недостатков для конкретного проекта.

Основными инструментами для вёрстки являются фреймы, таблицы и *div*. В стандарте *HTML5* фреймы больше не поддерживаются, а для работы стали доступны «семантические» блоки: *header*, *main*, *nav*, *section*, *article*, *aside*, *footer*. Однако блоки не дают определённых преимуществ, а являются лишь «синтаксическим сахаром» [1].

# Цель работы

Требуется разработать статический сайт по схеме, приведённой на рисунке 1, с использованием *HTML* и *CSS*. Общее количество страниц, составляющих сайт должно быть не менее восьми. При разработке предусмотреть удобную и понятную навигацию по сайту, а также возможность возврата с «конечных» страниц на родительские.

Необходимо включить в содержимое сайта ссылки на внешние ресурсы, таблицу, фреймы, нумерованный список, содержащий дисциплины, изображения, видео.



Рисунок 1 – Структурная схема сайта

# Теоретическая часть

## Краткие сведения о *HTML* и *CSS*

*HTML* – язык гипертекстовой разметки, который используется повсеместно для создания обычных веб-страниц, а также многофункциональных веб-приложений. Браузеры интерпретируют содержимое *HTML*-документа и отображают страницу в соответствии с инструкциями, включёнными в этот файл. *HTML* не является языком программирования, поскольку с его помощью нельзя производить вычислений и обрабатывать данные, его задачей является отображение данных, он позволяет создавать и структурировать разделы, абзацы и ссылки, используя теги и атрибуты.

*CSS* – язык таблицы стилей. Он, как и *HTML*, не является языком программирования. Он позволяет применять стили выборочно к элементам в документе *HTML*. Основной целью создания *CSS* являлось разделение логической структуры веб-страницы от описания внешнего вида этой веб-страницы. Хотя описание внешнего вида страницы допустимо помещать в сам *HTML*-документ, использование внешнего файла таблицы стилей считается хорошим тоном, как и использование внешних файлов с кодом на языке *JavaScript*, используемых в документе.

## Описание используемых тегов

Структура *HTML*- документа состоит из тегов. Во время разработки сайта использовались следующие теги:

– <*html*>, является корневым элементом документа. Все остальные элементы содержаться внутри него;

– <*head*>, содержит техническую информацию о странице: заголовок, метаинформацию, используемую поисковыми системами, ссылки на такие внешние ресурсы, как шрифт, файл таблицы стилей, иконку сайта, отображаемую слева от заголовка;

– <*body*>, содержит всё содержимое документа;

– <*ol*>, <*ul*>, <*li*>, представляют нумерованный список, маркированный список, отдельный элемент списка соответственно. Допустимо создавать вложенные списки, используя в качестве отдельного элемента список;

– <*a*>, является ссылкой. Всё содержимое этого тега является ссылкой, а с помощью атрибута *href* указывается путь к ресурсу;

– <*button*>, представляет кнопку. Его можно использовать в качестве альтернативы тегу <*a*>, но, чтобы обеспечить корректную работу кнопки необходимо задействовать обработчик, описанный с помощью языка *JavaScript*;

– <*p*>, представляет параграф с отступом;

– <*script*>, содержимое этого элемента представляет программный код. Часто код выносят во внешний файл, а затем подключают его в документе с использованием атрибута *src*, оставляя при этом содержимое тега пустым;

– <*link*>, устанавливает связь с внешним документом вроде файла со стилями или со шрифтами. В отличие от тега <*a*>, этот тег размещается всегда внутри контейнера head и не создаёт ссылку;

– <*map*>, служит контейнером для элементов <*area*>, которые определяют активные области для карт-изображений, которые устанавливают невидимые зоны на изображении, являющиеся ссылками на *HTML*-документы;

– <*img*>, предназначен для отображения на странице изображений в графическом формате. Адрес файла с картинкой задаётся с помощью атрибута *src*;

– <*table*>, <*tr*>, <*td*>, представляют таблицу, строку таблицы и колонку таблицы соответственно;

– <*div*>, является универсальным блочным тегом, который может содержать другие элементы;

– <*h*1>, …, <*h*6>, представляют заголовки разных размеров;

– <*section*>, <*footer*>, они не выделяются особым образом, это семантические элементы, то есть элементы со значением, например, первый определяет раздел в документе, а второй обычно содержит такую информацию о разделе, как имя автора, ссылки на соответствующие документы, авторские данные [2];

– <*iframe*>, создаёт плавающий фрейм, который находится внутри обычного документа, он позволяет загружать в область заданных размеров любые другие независимые документы;

– <*span*>, является универсальным строчным тегом;

– <*center*>, позволяет расположить содержимое тега, горизонтально центрировав его. Более предпочтительным способом является использование соответствующих *CSS*-свойств;

– <*b*>, <*strong*>, используются для полужирного выделения текста, второй вариант применяется с целью акцентирования внимания на такое начертание. Например, программы, которые читают и воспроизводят текст с экрана, для слепых и слабовидящих людей будут интонационно выделять этот текст, то же относится и к тегу <*em*>.

# Практическая часть

## Описания назначений каждой страницы

назначение:

объявления - на странице "Объявления" размещаются объявления для группы за последнюю неделю. так как, в основном, объявления довольно краткие, и их не так много,

справа от области с контентом находится несложный паттерн для замощения страницы;

расписание - страница "Расписание" выполнена в цветах и стилях приложения "Расписание БГУИР" для операционной системы андроид.

область контента представляет собой расписание на шесть дней с указанием учебных недель проведения занятия, а также времени начала и конца пары,

номера кабинета и корпуса и фамилии и инициалов преподавателя;

учебно-методический материал - страница "Учебно-методический материал" разделена на две части:

список дисциплин, состоящий из гиперссылок на странички-описания, загружаемых в область справа от списка, в ней также могут находиться гиперссылки на

методические пособия, лабораторные практикумы, руководства;

## Ход работы

Языки: HTML5, CSS, JavaScript (для создания обработчиков событий по нажатию кнопок)

Средства: Visual studio community 2019, notepad++

## Руководство пользователю

Как пользоваться сайтом.

# Заключение

В результате выполнения работы создан статический сайт о группе с использованием *HTML* и *CSS*.

# Список использованных источников

[1] *Htmlbook* [Электронный ресурс]. – Режим доступа : *http://htmlbook.ru/.*

[2] Пилгрим, М. Погружение в *HTML*5 / М. Пилгрим. – СПб. : БХВ-Петербург, 2011. – 304 с.