МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

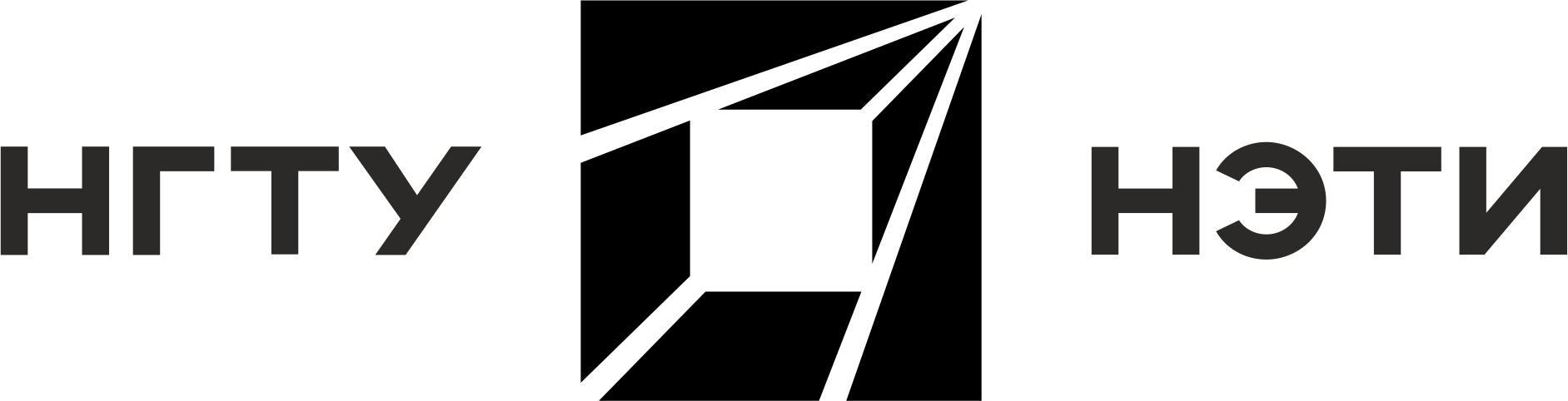
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра систем сбора и обработки данных



**ЛАБОРАТОРНАЯ РАБОТА №4**

по дисциплине: Сетевые информационные технологии

на тему: Технология WEB

Вариант №3

Факультет: ФПМИ

Группа: ПММ-21

Выполнили: Сухих А.С., Черненко Д.А.

Проверил: к.т.н., доцент Кобылянский В.Г.

Дата выполнения: 06.12.22

Отметка о защите:

Новосибирск 2022

**Цель работы:** Изучить основы Web-технологии, типы Web-документов и архитектуру программного обеспечения Web-браузеров.

**Ход работы:**

**1. Разработка сайта в виде простой HTML-страницы для доступа к отчетам.**

Был создан файл с HTML-разметкой, содержащей группу, имена членов бригады, а также таблица с отчетами по лабораторным работам. Разметка имеет вид:

<!DOCTYPE html>

<html>

<head>

<meta charset="RU">

<title>Сетевые технологии</title>

<div>

<h3>ПММ-21</h3>

Андрей Сухих<br>

Данила Черненко

</div>

</head>

<body>

<table>

<caption style="margin: 10px;">Лабораторные работы</caption>

<tr>

<th>Имя файла</th>

<th>Дата загрузки</th>

</tr>

<td><a href=".\lab1.pdf">Лабораторная 1</a></td>

<td>20.10.2022</td>

</tr>

</tr>

<td><a href=".\lab2.pdf">Лабораторная 2</a></td>

<td>30.10.2022</td>

</tr>

</tr>

<td><a href=".\lab3.pdf">Лабораторная 3</a></td>

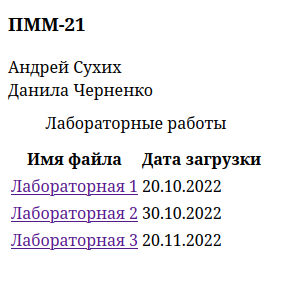
<td>20.11.2022</td>

</tr>

</table>

</body>

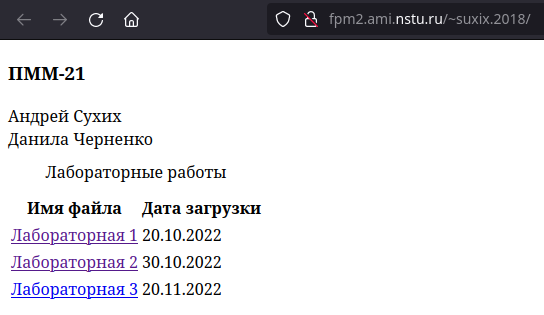
Файл был открыт в браузере. Результат показан на рисунке 1.1.

Рисунок 1.1 — отображение HTML-разметки сайта в браузере

**2. Загрузка HTML-страницы на Web-сервер НГТУ.**

Разработанная в п.1 страница была загружена с помощью Filezilla на сервер [*http://fpm2.ami.nstu.ru/*](http://fpm2.ami.nstu.ru/ю)

После конфигурации сервера страница стала доступна по адресу *http://fpm2.ami.nstu.ru/~suxix.2018/*

Рисунок 2.1. HTML-страница, загруженная на сервер

**3. Разработка веб-сервера.**

Сервер был разработан на языке C++ с использованием сокетов. Создаётся класс WebServer, в котором конфигурируется прослушивающий сокет на порту 4567, принимающий новые подключения. Данный сокет связывается со всеми доступными сетевыми адресами, таким образом он становится доступен как на loopback-интерфейсе, так и на интерфейсе локальной сети.

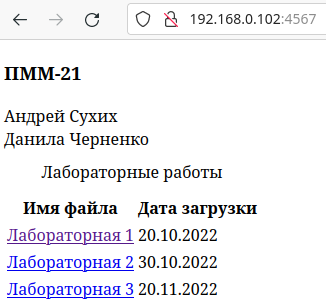
В бесконечном цикле принимаются новые подключения от клиентов, для каждого из которых заводится отдельный поток. Этот поток связывается с обработчиком запросов handleRequest(), где происходит считывание запроса с помощью функции recvAll(), далее парсинг запроса функцией parseRequest() и, наконец, отправка ответа функцией sendRequest().

Сам сервер предназначен для просмотра HTML-разметки, разработанной в п.1. При поступлении любого запроса проверяется какой метод был выполнен. Если был указан любой метод кроме GET, то возвращает ошибка 405 Method Not Allowed. Далее производится проверка наличия запрашиваемого ресурса на диске. Если запрос был выполнен к корневому разделу, то считается, что запрашивался файл index.html. Если запрашиваемого ресурса нет, то возвращается ошибка 404. Также проверяется расширение запрашиваемого ресурса. Среди доступных есть html и pdf, в случае несоответствия возвращается ошибка 400 Bad Request.

В случае прохождения всех проверок формируется ответ в соответствии с протоколом HTTP. Ответ направляется в сокет, и по завершению отправки сокет закрывается. В случае возникновения любой непредвиденной ошибки на стороне сервера возвращается ошибка 500 Internal Server Error.

**4. Просмотр страницы на разработанном веб-сервере**

Сервер и клиент были запущены на различных хостах локальной сети. IP сервера — 192.168.0.102. IP клиента — 192.168.0.101. С браузера клиента был выполнен запрос к коренному элементу сервера. В браузере была отображена HTML-страница, разработанная в п.1. Результат показан на рисунке 4.1.

Рисунок 4.1 — отображение страницы на клиенте при запросе браузера к серверу

5**. Анализ запроса к серверу moodle.ami.nstu.ru с помощью режима разработчика в браузере.**

С сервера moodle был скачан файл «Календарный план». С помощью режима разработчика можно увидеть полный текст запроса:

GET /pluginfile.php/5366/mod\_resource/content/9/%D0%9A%D0%B0%D0%BB%D0%B5%D0%BD%D0%B4%D0%B0%D1%80%D0%BD%D1%8B%D0%B9%20%D0%BF%D0%BB%D0%B0%D0%BD\_%D0%A1%D0%98%D0%A2\_2022\_new.docx HTTP/1.1

Host: moodle.ami.nstu.ru

User-Agent: Mozilla/5.0 (X11; Linux x86\_64; rv:107.0) Gecko/20100101 Firefox/107.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,\*/\*;q=0.8

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

Referer: http://moodle.ami.nstu.ru/course/view.php?id=24

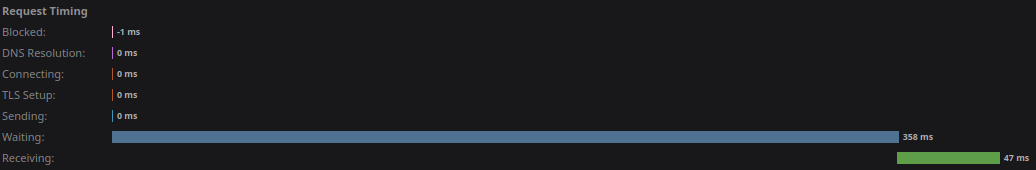
Connection: keep-alive

Cookie: <cookies>

Upgrade-Insecure-Requests: 1

Заголовок Cookie был сокращен, так как содержит в себе много информации.

Также были определены временные затраты на выполнение этапов запроса, они показаны на рисунке 5.1

Рисунок 5.1 — затраты времени на выполнение GET запроса на скачивание файла

Как можно заметить, наибольшие временные затраты связаны с ожиданием ответа от сервера (358 мс) и получением ответа (47 мс).

**Вывод:** В ходе лабораторной работы нашей бригадой были изучены основы Web-технологии, создана простая HTML-страница, а также разработан веб-сервер на языке C++ с использованием сокетов.