Министерство образования и науки Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЁТ**

по лабораторной работе №4

по курсу "Разработка web-приложений"  
на тему " Маршрутизация запросов"

Выполнили:

Студенты группы 16ВВ2

Воронков Е.В.

Духанов И.В.

Проверили:

Дубравин А.В.  
Карамышева Н.С.

Пенза 2020

**Цель работы**

Выполнить модернизацию созданной ранее html-страницы в соответствии с вариантом. Вариант-2.

**1. Задание:**

1. Реализовать роутер

2. Реализовать маршрутизацию внутри http-сервера

3. Выполнить контрольное задание. Выполнить редактирование кода, реализующего сервер, следующим образом: добавить обработку для трех различных URL запросов к серверу. В обработчиках запросов выводить в консоль сообщение о том, какой обработчик был вызван.

**Выполнение:**

Добавили обработку для трех различных URL запросов к серверу.

Листинг файла index.js:

var server = require("./server");

var router = require("./router");

var requestHandlers = require("./requestHandlers");

var handle = {}

handle["/"] = requestHandlers.start;

handle["/start"] = requestHandlers.start;

handle["/favicon.ico"] = requestHandlers.favicon;

handle["/upload"] = requestHandlers.upload;

handle["/placeHolder"] = requestHandlers.placeHolder;

server.start(router.route, handle);

Листинг файла router.js:

function route(handle, pathname, response) {

console.log("About to route a request for " + pathname);

if (typeof handle[pathname] === 'function') {

handle[pathname](response);

}

else {

console.log("No request handler found for " + pathname);

response.writeHead(200, { 'Content-Type': 'text/html' });

response.end();

}

}

exports.route = route

Листинг файла requestHandlers.js:

const fs = require("fs");

function start(response) {

console.log("Request handler 'start' was called.");

const page = fs.readFileSync('index.html');

response.writeHead(200, { 'Content-Type': 'text/html' });

response.write(page);

response.end();

}

function upload(response) {

console.log("Request handler 'upload' was called.");

const page = fs.readFileSync('upload.html');

response.writeHead(200, { 'Content-Type': 'text/html' });

response.write(page);

response.end();

}

function favicon(response) {

console.log("Request handler 'favicon' was called.");

response.writeHead(404);

response.end();

}

function placeHolder(response) {

console.log("Request handler 'placeHolder' was called.");

const page = fs.readFileSync('placeHolder.html');

response.writeHead(200, { 'Content-Type': 'text/html' });

response.write(page);

response.end();

}

exports.start = start;

exports.upload = upload;

exports.favicon = favicon;

exports.placeHolder = placeHolder;

Листинг файла server.js:

const fs = require("fs");

const http = require('http');

var url = require("url");

var querystring = require("querystring");

function start(route, handle)

{

var server = http.createServer(function onRequest(request, response) {

var pathname = url.parse(request.url).pathname;

console.log("Request for " + pathname + " recieved.");

route(handle, pathname, response);

}

);

server.listen(8888);

console.log("Server started!");

}

exports.start = start;

Листинг файла index.html:

<html>

<head>

<title>Wow</title>

<meta charset="utf-8">

</head>

<body>

Лабораторная работа 4. Маршрутизация запросов<br /><br />

<a href = "/upload">Запросить /upload</a><br />

<a href = "/placeHolder">Запросить /placeHolder</a><br />

<a href = "/notHandled">Запросить /notHandled</a>

</body>

</html>

Листинг файла upload.html:

<html>

<head>

<title>Wow</title>

<meta charset="utf-8">

</head>

<body>

Лабораторная работа 4. Маршрутизация запросов<br /><br />

Был запрошен URL /upload<br /><br />

<a href = "/start">Запросить /start</a><br />

<a href = "/placeHolder">Запросить /placeHolder</a><br />

<a href = "/notHandled">Запросить /notHandled</a>

</body>

</html>

Листинг файла placeHolder.html:

<html>

<head>

<title>Wow</title>

<meta charset="utf-8">

</head>

<body>

Лабораторная работа 4. Маршрутизация запросов<br /><br />

Был запрошен URL /placeHolder<br /><br />

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor<br />

incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis<br />

nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.<br />

Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu<br />

fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in <br />

culpa qui officia deserunt mollit anim id est laborum.<br /><br />

<a href = "/start">Запросить /start</a><br />

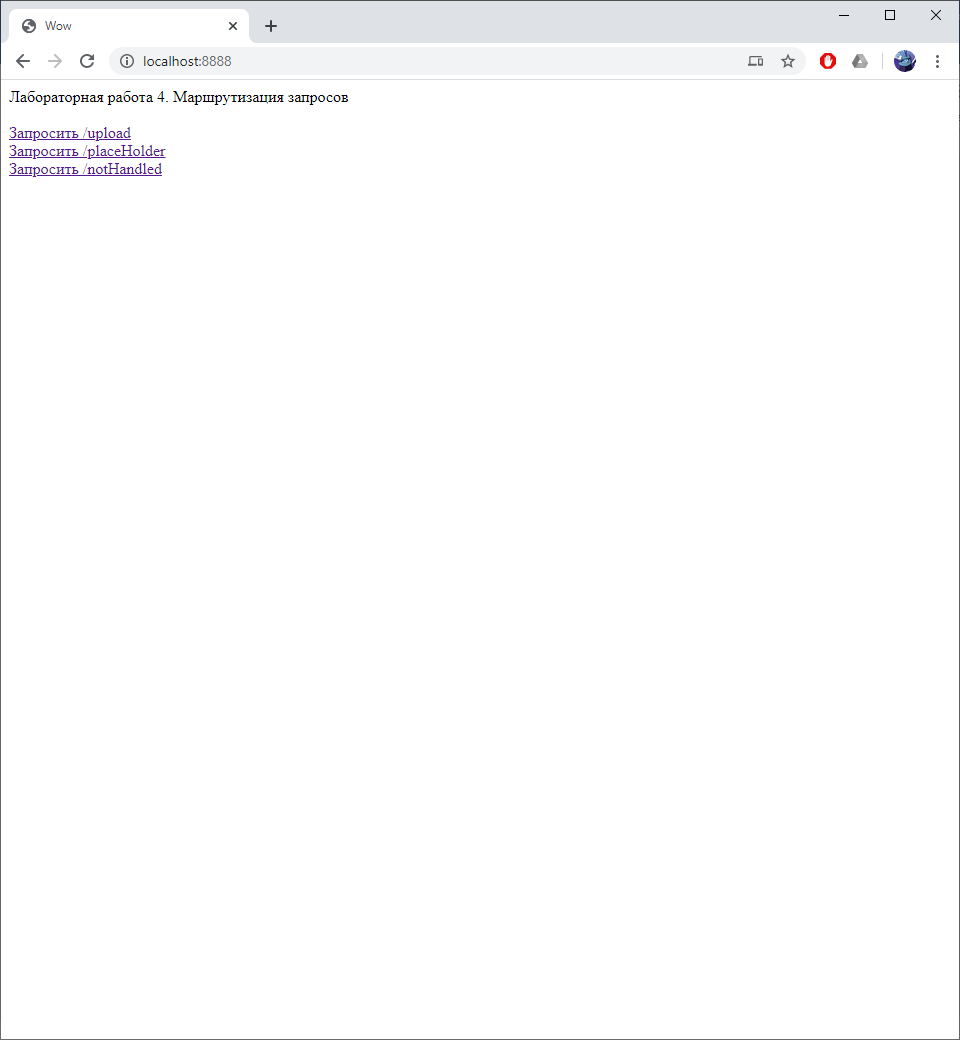
<a href = "/upload">Запросить /upload</a><br />

<a href = "/notHandled">Запросить /notHandled</a>

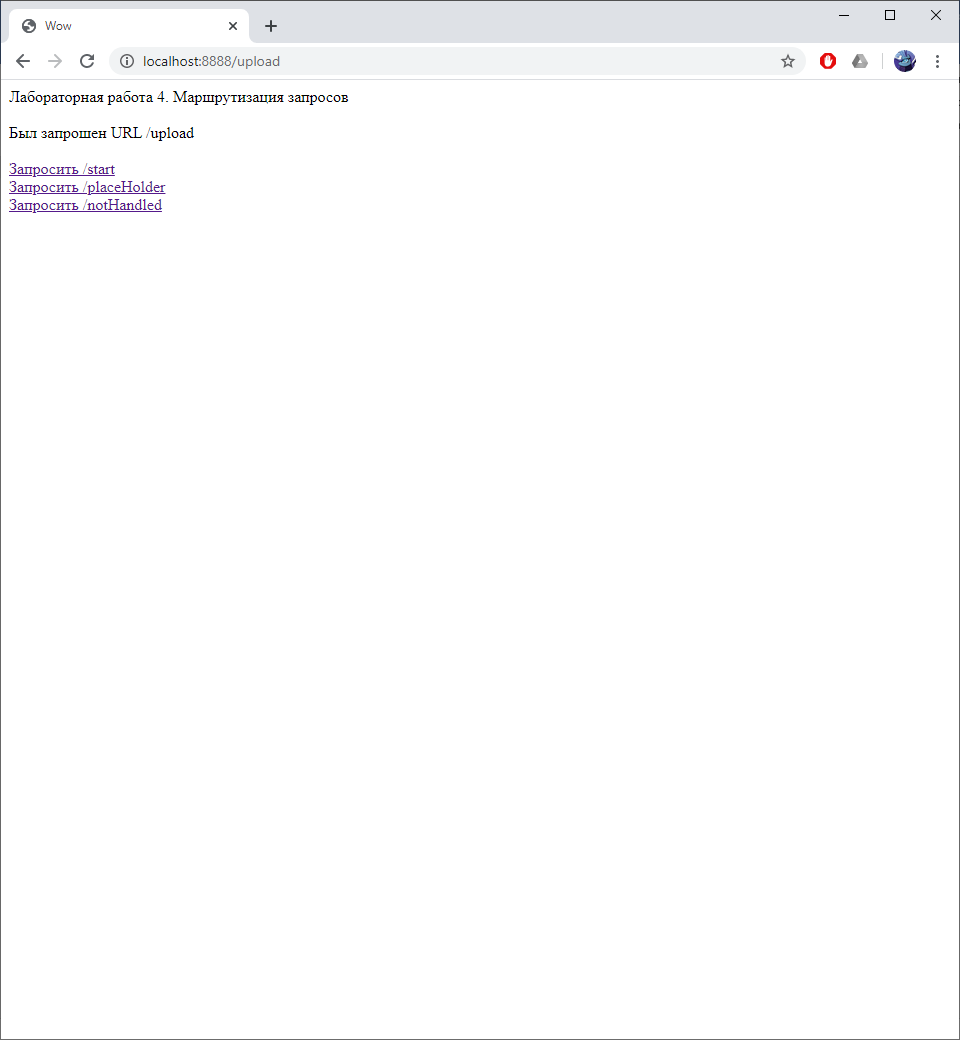
</body>

</html>

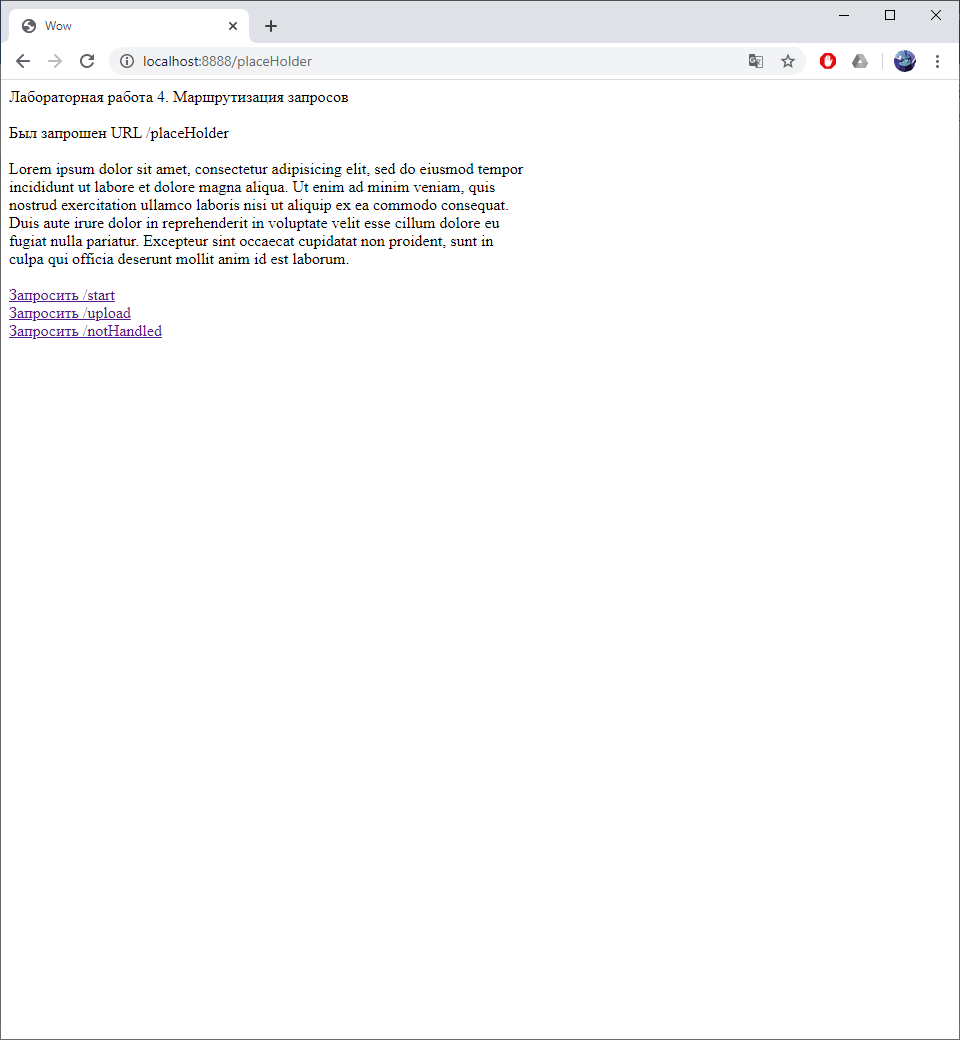
Результаты работы представлены на рисунках 1-3.



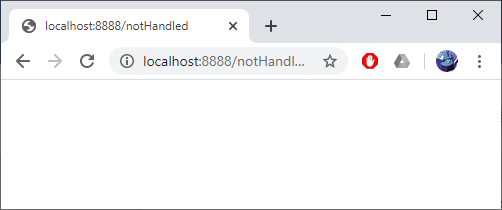
**Рисунок 1 – Запрос /**



**Рисунок 2 – Запрос /upload**

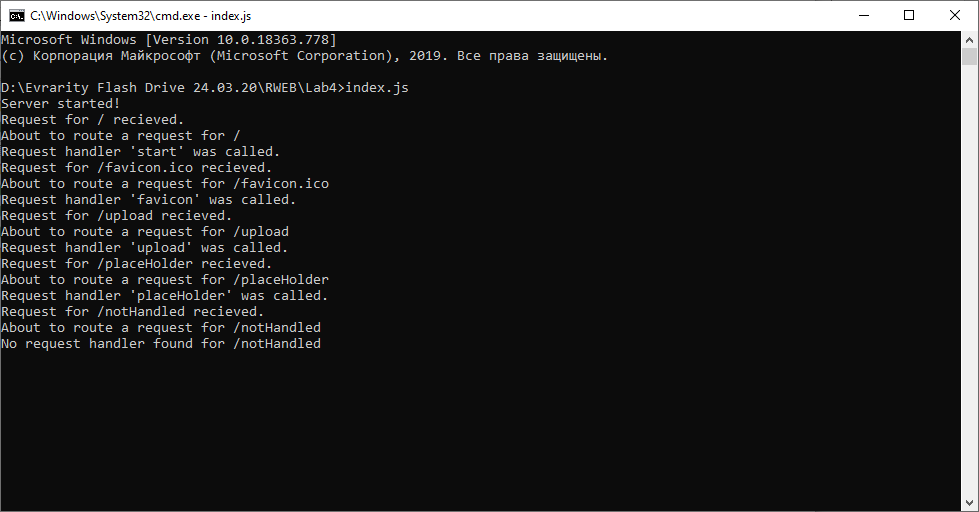


**Рисунок 3 – Запрос /placeholder**



**Рисунок 4 – Запрос /notHandled**

В обработчиках запросов выводится в консоль сообщение о том, какой обработчик был вызван:



**Рисунок 5 – Результат работы**

**Вывод**

В ходе выполнения лабораторной работы была выполнена модернизация созданной ранее html-страницы в соответствии с вариантом.