

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Алтайский государственный технический университет им. И. И. Ползунова»  
Университетский технологический колледж

Отчет защищен с оценкой \_\_\_\_\_

Преподаватель \_\_\_\_\_ *С. В. Умбетов*

«\_\_\_\_\_» \_\_\_\_\_ 2025 г.

Отчёт по практической работе №2  
по теме:  
«Создание HTTP сервера»

Студент группы 1ИСП-21 \_\_\_\_\_ *Д.А. Рукин*  
группа и.о., фамилия

Преподаватель ассистент, к. т. н. \_\_\_\_\_ *С. В. Умбетов*  
должность, ученая степень и.о., фамилия

БАРНАУЛ 2025

## Практическая работа №2

Задание принял:

A handwritten signature in black ink, consisting of a large, stylized 'D' followed by a checkmark-like stroke.

Подпись

Рукин Д.А.

ФИО

## Ход работы

```
const http = require('http')
const fs = require('fs')
const path = require('path')

http
  .createServer((request, response) => {
    let filePath

    if (request.url === '/') {
      filePath = 'pages/index.html'
    } else if (request.url === '/styles/style.css') {
      filePath = 'styles/style.css'
    } else if (request.url === '/scripts/script.js') {
      filePath = 'scripts/script.js'
    } else {
      filePath = 'pages/404.html'
    } if (request.url === '/styles/404.css') {
      filePath = 'styles/404.css'
    } else if (request.url === '/img/cat-404.png') {
      filePath = 'img/cat-404.png'
    }

    const ext = path.extname(filePath)
    let contentType = 'text/html'
    if (ext === '.png') contentType = 'image/png'
    if (ext === '.css') contentType = 'text/css'
    if (ext === '.js') contentType = 'text/javascript'

    fs.readFile(filePath, (err, content) => {
      if (err) {
        if (err.code === 'ENOENT') {
          // Если файл не найден, читаем 404.html
          fs.readFile('pages/404.html', (err, content) => {
            response.writeHead(404, { 'Content-Type': 'text/html' })
            response.end(content, 'utf-8')
          })
        } else {
          response.writeHead(500)
          response.end('Server Error: ' + err.code)
        }
      } else {
        response.writeHead(200, { 'Content-Type': contentType })
        response.end(content, 'utf-8')
      }
    })
  })
  .listen(3000, '127.0.0.1', () => {
    console.log('Server running at http://127.0.0.1:3000/')
  })
```

Рисунок 1 – Код сервера

```
PS C:\Users\denis\OneDrive\Документы\Лабораторные\Практика Умбетов\PracticRep> cd Lr1
PS C:\Users\denis\OneDrive\Документы\Лабораторные\Практика Умбетов\PracticRep\Lr1> node app.js
Server running at http://127.0.0.1:3000/
```

Рисунок 2 – Запуск сервера

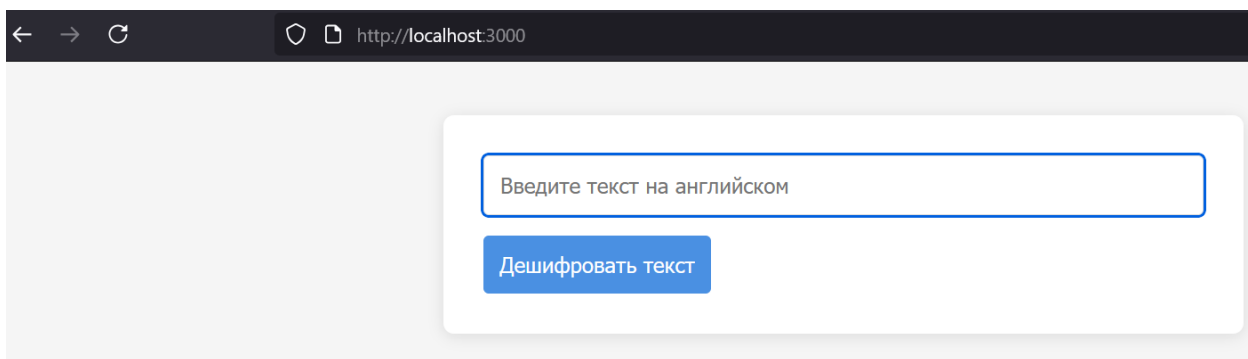


Рисунок 3 – Главная страница (localhost)

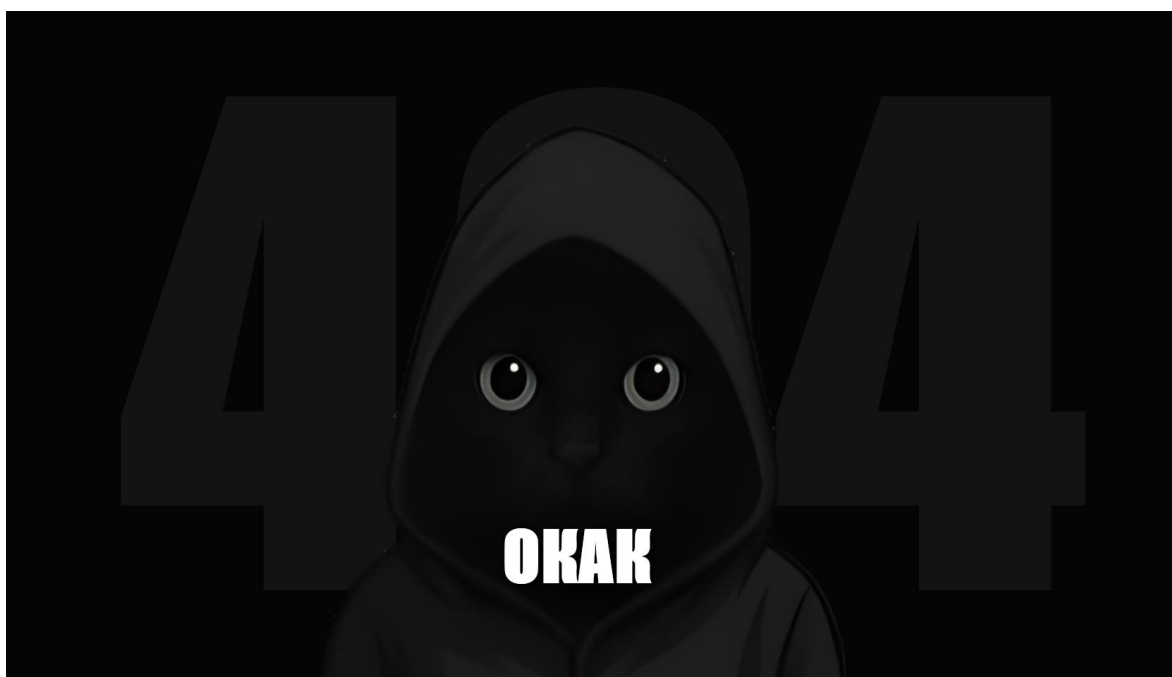


Рисунок 4 – Страница ошибки 404

`http://localhost:3000/asdasdasd`

Рисунок 5 – Пример адреса, который вызовет ошибку 404

```
<!DOCTYPE html>
<html lang="ru">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width" />
    <link rel="stylesheet" href="/styles/404.css">
    <title>Страница не найдена</title>
  </head>
  <body>
    <div class="cat-error">
      <div class="error">
        404
      </div>
      <div class="image-404">
        
        <h1 class="okak">ОКАК</h1>
      </div>
    </div>
  </body>
</html>
```

Рисунок 6 – Файл «404.html»

**Вывод:**

В ходе работы были приобретены навыки построения сервера, изучение функций его рендера и подключение модулей и стилей.