

**Московский государственный технический  
университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»  
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Разработка интернет приложений»

Отчет по лабораторной работе №7  
«Разработка пользовательского интерфейса для работы с REST API»

Выполнил:  
студентка группы ИУ5-52Б  
Федосеева Елизавета

Проверил:  
преподаватель каф. ИУ5  
Гапанюк Ю.Е.

Москва, 2021 г.

## Описание задания:

На основе методических указаний разработайте пользовательский интерфейс для работы с REST API. Используйте REST API, разработанный Вами в предыдущей лабораторной работе.

### Файл `server/index.js`:

```
'use strict';

const express = require('express');
const path = require('path');
const app = express();
const htmlPath = path.resolve(__dirname, '..',
'src/index.html')
const staticPath = path.resolve(__dirname, '..', 'src')
const port = 3000;

app.use('/', express.static(staticPath));

app.all('*', (req, res) => {
  res.sendFile(htmlPath);
});

app.listen(port, () => {
  console.log(`Server listening http://localhost:${port}`);
});
```

### Файл `src/components/back-button/BackButtonComponent.js`:

```
export class BackButtonComponent {
  constructor(parent) {
    this.parent = parent
  }

  getHTML() {
```

```

        return '<button id="back-button" class="btn btn-
primary" type="button">Назад</button>'
    }

    addListeners(listener) {
        document.getElementById("back-
button").addEventListener('click', listener)
    }

    render(listener) {
        const html = this.getHTML()
        this.parent.insertAdjacentHTML('beforeend', html)
        this.addListeners(listener)
    }
}

```

### **Файл src/components/lang /LangCardComponent.js:**

```

export class LangCardComponent {
    constructor(parent) {
        this.parent = parent
    }

    getHTML(data) {
        return (
            `
                <div class='card text-white text-center bg-dark'>
                    <div class='card-body'>
                        <h4 class='card-
title'>${data.language_name}</h4>
                        <!--<p class='card-
text'>${data.descript}</p>-->
                        <!--<p class='card-
text'>${data.creation_year}</p>-->
            `
        )
    }
}

```

```

        <button class='btn btn-primary',
id='click-card-${data.id_language}', data-
id='${data.id_language}'>Подробнее</button>
        </div>
    </div>
    `
    )
}

addListeners(data, listener) {
    document.getElementById(`click-card-
${data.id_language}`).addEventListener('click', listener)
}

render(data, listener) {
    const html = this.getHTML(data)
    this.parent.insertAdjacentHTML('beforeend', html)
    this.addListeners(data, listener)
}
}

```

### **Файл src/components/lang /LangComponent.js:**

```

export class LangComponent {
    constructor(parent) {
        this.parent = parent
    }

    getHTML(data) {
        return (
            `
            <div class='card text-white bg-dark' style='width:
300px'>
                <div class="card-header">
                    Язык программирования
                </div>
            `
        )
    }
}

```

```

        <div class='card-body'>
            <h4 class='card-
title'>${data.language_name}</h4>
            <p class='card-text'>${data.descript}</p>
            <p class='card-text'>Год создания:
${data.creation_year}</p>
        </div>
    </div>
    `
    )
}

render(data) {
    const html = this.getHTML(data)
    this.parent.insertAdjacentHTML('beforeend', html)
}
}

```

### **Файл src/modules/ajax.js:**

```

class Ajax {
    async get(url) {
        const response = await fetch(url, {method: "GET"});
        const responseData = await response.json();
        return {status: response.status, data: responseData};
    }
}

export const ajax = new Ajax();

```

### **Файл src/modules/urls.js:**

```

class Urls {
    constructor() {
        this.url = 'http://localhost:8000/api/';
    }
}

```

```

    langs() {
        return `${this.url}lang/`
    }

    lang(id) {
        return `${this.url}lang/${id}/`
    }
}

export const urls = new Urls()

```

### **Файл src/pages/lang/LangPage.js:**

```

import { LangComponent } from
"../../components/lang/LangComponent.js"
import { BackButtonComponent } from "../../components/back-
button/BackButtonComponent.js"
import { MainPage } from "../../main/MainPage.js"
import { ajax } from "../../modules/ajax.js"
import { urls } from "../../modules/urls.js"

export class LangPage {
    constructor(parent, id) {
        this.parent = parent
        this.id = id
    }

    async getData() {
        return ajax.get(urls.lang(this.id))
    }

    page() {
        return document.getElementById('stock-page')
    }
}

```

```

getHTML() {
    return '<div id="stock-page"></div>'
}

clickBack() {
    const mainPage = new MainPage(this.parent)
    mainPage.render()
}

async render() {
    this.parent.innerHTML = ''
    const html = this.getHTML()
    this.parent.insertAdjacentHTML('beforeend', html)

    const back_button = new
BackButtonComponent(this.page())
    back_button.render(this.clickBack.bind(this))

    const data = await this.getData()
    const lang_card = new LangComponent(this.page())
    lang_card.render(data.data)
}
}

```

### **Файл src/pages/main/MainPage.js:**

```

import { LangCardComponent } from
"../../components/lang/LangCardComponent.js"
import { ajax } from "../../modules/ajax.js"
import { urls } from "../../modules/urls.js"
import { LangPage } from "../../pages/lang/LangPage.js"

export class MainPage {
    constructor(parent) {
        this.parent = parent
    }
}

```

```

    async getData() {
        return ajax.get(urls.langs())
    }

    page() {
        return document.getElementById('main-page')
    }

    getHTML() {
        return '<div id="main-page" class="d-flex flex-wrap"><div/>'
    }

    clickCard(e) {
        const cardId = e.target.dataset.id
        const langPage = new LangPage(this.parent, cardId)
        langPage.render()
    }

    async render() {
        this.parent.innerHTML = ''
        const html = this.getHTML()
        this.parent.insertAdjacentHTML('beforeend', html)

        const data = await this.getData()
        data.data.forEach((item) => {
            const lang_card = new
LangCardComponent(this.page())
            lang_card.render(item, this.clickCard.bind(this))
        })
    }
}

```



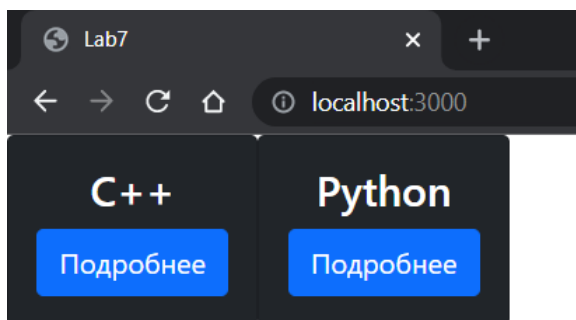
### Файл src/index.html:

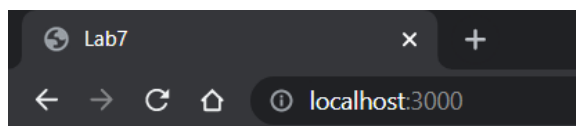
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Lab7</title>
  <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bo
otstrap.min.css">
</head>
<body>
  <div id="root"></div>
  <script src="main.js" type="module"></script>
  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/boot
strap.bundle.min.js"></script>
</body>
</html>
```

### Файл src/main.js:

```
import {MainPage} from "../pages/main/MainPage.js";
const root = document.getElementById('root');
const mainPage = new MainPage(root)
mainPage.render()
```

### Пользовательский интерфейс:





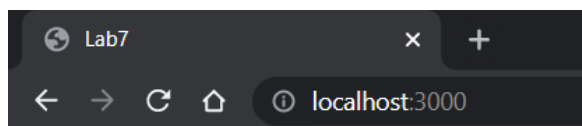
Назад

Язык программирования

**C++**

C++ is the best programming language!

Год создания: 1983



Назад

Язык программирования

**Python**

Python is often used for machine learning.

Год создания: 2021