

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

Отчёт по лабораторной работе №7 по курсу «Разработка интернет  
приложений».

«Разработка пользовательского интерфейса для работы с REST API».

Выполнил:  
Анцифров Н. С.  
студент группы ИУ5-51Б

Проверил:  
Гапанюк Ю. Е.

Подпись и дата:

Подпись и дата:

Москва, 2021 г.

## Содержание

1. Задание лабораторной работы. ....	2
2. Текст программы.....	2
3. Выполнение программы.....	7

### 1. Задание лабораторной работы.

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

### 2. Текст программы.

*lab7/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}`);
});
```

*lab7/src/components/back-button/BackButtonComponent.js*

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

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

  getHTML() {
    return (
      <button id="back-button" class="btn btn-primary"
type="button">Назад</button>
    )
  }
}
```

```

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

```

### *lab7/src/components/button/ButtonComponent.js*

```

export class ButtonComponent {
  constructor(parent) {
    this.parent = parent;
  }

  render() {
    this.parent.insertAdjacentHTML('beforeend', '<button type="button"
class="btn btn-primary">Hello world 4!</button>')
  }
}

```

### *lab7/src/components/stock/StockComponent.js*

```

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

  getHTML(data) {
    return (
      <div class="card mb-3" style="width: 540px;">
        <div class="row g-0">
          <div class="col-md-8">
            <div class="card-body">
              <h5 class="card-title">${data.name}</h5>
              <p class="card-text">${data.description}</p>
              <p class="card-text">${data.rating}</p>
            </div>
          </div>
        </div>
      </div>
    )
  }

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

```

### *lab7/src/components/stock/stock-card/StockCardComponents.js*

```

export class StockCardComponent {
  constructor(parent) {
    this.parent = parent;
  }
  getHTML(data) {
    return (
      <div class="card" data-link="1" style="width: 300px;">

```

```

        <div class="card-body">
            <h5 class="card-title">${data.name}</h5>
            <p class="card-text">${data.description}</p>
            <p class="card-text">${data.rating}</p>
            <button class="btn btn-primary" id="click-card-
${data.pk}" data-id="${data.pk}">Подробнее</button>
        </div>
    </div>
    )
}

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

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

```

*lab7/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();

```

*lab7/src/modules/urls.js*

```

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

    stocks() {
        return `${this.url}movies/`
    }

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

export const urls = new Urls()

```

### lab7/src/pages/main/MainPage.js

```
import {StockCardComponent} from "../../components/stock-card/StockCardComponents.js";
import {StockPage} from "../../stock/StockPage.js";
import {ajax} from "../../modules/ajax.js";
import {urls} from "../../modules/urls.js";

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

  get 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 stockPage = new StockPage(this.parent, cardId)
    stockPage.render()
  }

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

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

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

### lab7/src/pages/stock/StockPage.js

```
import {StockComponent} from "../../components/stock/StockComponent.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 StockPage {
  constructor(parent, id) {
    this.parent = parent;
    this.id = Number(id);
  }
}
```

```

    }

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

    get 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 backButton = new BackButtonComponent(this.page)
        backButton.render(this.clickBack.bind(this))

        const data = await this.getData()
        const stock = new StockComponent(this.page)
        stock.render(data.data)
    }
}

```

*lab7/src/index.html*

```

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

```

lab7/src/main.js

```
import {MainPage} from "../pages/main/MainPage.js";

const root = document.getElementById('root');

const mainPage = new MainPage(root)
mainPage.render()
```

### 3. Выполнение программы.

Запустим сервер из предыдущей лабораторной работы через cmd:

```
C:\Users\firry>cd C:\Users\firry\OneDrive\Документы\МГТУ\5 семестр\РИП\Лаб6\lab6
C:\Users\firry\OneDrive\Документы\МГТУ\5 семестр\РИП\Лаб6\lab6>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
January 15, 2022 - 14:13:32
Django version 4.0, using settings 'lab6.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

На сервере находятся записи:

#### Api Root

The default basic root view for DefaultRouter

GET /

HTTP 200 OK

Allow: GET, HEAD, OPTIONS

Content-Type: application/json

Vary: Accept

```
{
  "movies": "http://127.0.0.1:8000/movies/"
}
```

#### Movie List

API endpoint, который позволяет просматривать и редактировать акции компаний

GET /movies/

HTTP 200 OK

Allow: GET, POST, HEAD, OPTIONS

Content-Type: application/json

Vary: Accept

```
[
  {
    "pk": 4,
    "name": "Назад в будущее (1985)",
    "description": "Подросток Марти МакФлай с помощью машины времени, сооруженной его другом-профессором доком Брауном, попадает из 80-х в далекие 50-е. Там он встречает",
    "rating": "8.6",
    "pic": "https://avatars.mds.yandex.net/get-kinopoisk-image/1599028/73cf2ed0-fd52-47a2-9e26-74104360786a/600x900",
    "is_growing": false,
    "date_modified": "2022-01-15T11:10:13.992418Z"
  },
  {
    "pk": 5,
    "name": "Назад в будущее 2 (1989)",
    "description": "Продолжение фантастической истории о приключениях американского подростка во времени. На этот раз с помощью модернизированной Доком машины времени",
    "rating": "8.3",
    "pic": "https://upload.wikimedia.org/wikipedia/ru/0/07/Back-to-Future-Part-II-529542.jpg",
    "is_growing": false,
    "date_modified": "2022-01-15T11:11:06.050462Z"
  },
  {
    "pk": 6,
    "name": "Назад в будущее 3 (1990)",
    "description": "Несмотря на просьбы Дока не навещать его в XIX веке, Марти вновь хочет вмешаться в ход событий и на машине времени уносится в 1885 год, навстречу Д",
    "rating": "8.2",
    "pic": "https://static.kinoafisha.info/k/movie_posters/canvas/800x1200/upload/movie_posters/3/8/7/5412783/16212525745412783.jpg",
    "is_growing": false,
    "date_modified": "2022-01-15T11:12:11.599601Z"
  }
]
```

Запустим сервер с помощью команды «node server» в WebStorm и проверим его работу:

