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:  студент группы ИУ5-54Б  Кашурин М.В. | Проверил: |
| преподаватель каф. ИУ5 |
| Гапанюк Ю.Е. |

Москва, 2021 г.

**Задание:**

1. На основе [методических указаний](https://github.com/iu5team/iu5web-fall-2021/blob/main/tutorials/lab7/lab7_tutorial.md) разработайте пользовательский интерфейс для работы с REST API. Используйте REST API, разработанный Вами в предыдущей лабораторной работе.

**Текст программы MainPage.js**

import {OsCardsComponents} from "../../components/os- cards/OsCardsComponents.js";

import {OsPage} from "../../os/OsPage.js"; import {ajax} from "../../modules/ajax.js"; import {urls} from "../../modules/urls.js";

export class MainPage { constructor(parent) {

this.parent = parent;

}

clickCard(e) {

const cardId = e.target.dataset.id

const osPage = new OsPage(this.parent, cardId) osPage.render()

}

get page() {

return document.getElementById('main-page')

}

getHTML() {

return (

`

<div id="main-page" class="d-flex flex-wrap"><div/>

`

)

}

getData() {

return ***ajax***.get(***urls***.oss())

}

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

this.parent.insertAdjacentHTML('beforeend', html)

const data = await this.getData() data.data.forEach((item) => {

const OSCard = new OsCardsComponents(this.parent) OSCard.render(item, this.clickCard.bind(this))

})

}

}

**OsPage**

import {OsComponent} from "./OsComponent.js";

import {BackButtonComponent} from "../components/back- button/BackButtonComponent.js";

import {MainPage} from "../pages/main/MainPage.js"; import {***ajax***} from "../modules/ajax.js";

import {***urls***} from "../modules/urls.js";

export class OsPage { data1;

constructor(parent, id) { this.parent = parent this.id = id

}

clickBack() {

const mainPage = new MainPage(this.parent) mainPage.render()

}

async getData() {

return ***ajax***.get(***urls***.os(this.id))

}

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 osComp = new OsComponent(this.page) osComp.render(data.data)

}

get page() {

return ***document***.getElementById('stock-page')

}

getHTML() {

return (

`

<div id="stock-page">

</div>

`

)

}

}

**Urls.js**

class Urls {

constructor() {

this.url = 'http://localhost:8000/';

}

oss() {

return `${this.url}os/`

}

os(id) {

return `${this.url}os/${id}/`

}

}

export const ***urls*** = new Urls()

**OsComponent.js**

export class OsComponent { tt; constructor(parent) {

this.parent = parent

}

getHTML(data) { this.tt=data.os\_name return (

`

<div class="card mb-3" style="width: 540px;">

<div class="row g-0">

<div class="col-md-4">

<img src="${data.src}" class="img-fluid"

alt="картинка">

</div>

<div class="col-md-8">

<div class="card-body">

<h5 class="card-title">${data.os\_name}</h5>

<p class="card-text">${data.os\_descript}</p>

</div>

</div>

</div>

</div>

`

)

}

render(data) {

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

}

}

**OsCardsComponents.js**

export class OsCardsComponents { constructor(parent) {

this.parent = parent;

}

addListeners(data, listener) {

***document***

.getElementById(`click-card-${data.pk}`)

.addEventListener("click", listener)

}

getHTML(data) { return (

`

<div class="card" style="width: 300px;">

<div class="card-body">

<img src="${data.src}" class="img-fluid" alt="картинка">

<h5 class="card-title">${data.os\_name}</h5>

<p class="card-text">${data.os\_descript}</p>

<button class="btn btn-primary" id="click-card-

${data.pk}" data-id="${data.pk}">Нажми на меня</button>

</div>

</div>

`

)

}

render(data, listener) {

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

}

}

**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)

}

}

**Результат выполнения программы:**



