|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

|  |  |
| --- | --- |
| ФАКУЛЬТЕТ | РАДИОТЕХНИЧЕСКИЙ |

|  |  |
| --- | --- |
| КАФЕДРА | СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ |

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №8**

Студент Столярова Ольга Денисовна .

*фамилия, имя, отчество*

Группа РТ5-51Б .

Название предприятия МГТУ им. Н. Э. Баумана ю

Студент Столярова О.Д.

Преподаватель Гапанюк Ю.Е.

*2021 г.*

**Цель работы**

Bзучение возможностей создания пользовательского интерфейса в веб-приложениях с использованием библиотеки React.

**Задание**

На основе [методических указаний](https://github.com/iu5team/iu5web-fall-2021/blob/main/tutorials/lab8/lab8_tutorial.md) разработайте React-приложение. Для создания приложения необходимо решить следующие задачи:

1. Создать стартовый React-проект. Удалить неиспользуемый код. Организовать директории для страниц, компонентов, утилит и работы с сетью.
2. Организовать роутниг в веб-приложении.
3. Разработать базовые страницы, на которых будут отображаться сущности из выбранной вами предметной области:
   1. Стартовая страница.
   2. Страница просмотра списка объектов.
   3. Страница просмотра конкретного объекта.
4. Вынести переиспользуемые компоненты в отдельные файлы:
   1. Для навигации по приложению можно добавить header.
   2. Для отображения дополнительной информации (данные о студенте и предметной области) можно использовать footer.
   3. Источники ввода-вывода (поля ввода (inputs)/формы/текстовые блоки).
   4. Переиспользуемые таблицы/гриды.
5. Добавить асинхронные запросы в разработанный API, чтобы страница получала данные с сервера.
6. Если в Вашем проекте реализована сложная логика работы с состоянием приложения, то рекомендуется добавить пользовательские хуки.
7. Страницы приложения должны хорошо отображаться как на больших, так и на маленьких экранах.

**Код программы:**

**Файл src/App.js:**

import { BrowserRouter, Route, Switch } from "react-router-dom";

import HomePage from './pages/HomePage'

import ListPage from "./pages/ListPage";

import ObjectPage from "./pages/ObjectPage";

function App() {

  return (

      <BrowserRouter basename="/" >

          <Switch>

            <Route exact path="/" component={HomePage}></Route>

            <Route exact path="/list" component={ListPage}></Route>

            <Route exact path="/list/object" component={ObjectPage}></Route>

          </Switch>

      </BrowserRouter>

  );

}

export default App;

**Файл src/components/HeaderComp.js:**

import { Link } from "react-router-dom";

function HeaderComp() {

    return (

        <div>

            <Link to='/'>Главная</Link>

            <hr></hr>

        </div>

    );

}

export default HeaderComp;

**Файл src/components/FooterComp.js:**

function FooterComp() {

    return (

        <div style={{bottom: '20px', position: 'absolute'}}>

            <hr></hr>

            <i>Группа:</i> РТ5-51Б

            <br></br>

            <i>Студентка:</i> Столярова Ольга Денисовна

        </div>

    );

}

export default FooterComp;

**Файл src/modules/GetObjects.js:**

const getObjects = async () =>{

    const res = await fetch('http://127.0.0.1:8000/stocks/', {method: "GET"})

        .then((response) => {

            return response.json();

        }).catch(()=>{

            return {resultCount:0, results:[]}

        })

    return res

}

export default getObjects

**Файл src/Pages/HomePage.js:**

import { Link } from "react-router-dom";

import HeaderComp from "../components/HeaderComp";

import FooterComp from "../components/FooterComp";

function HomePage() {

    return (

        <div>

            <HeaderComp></HeaderComp>

            <h2>Главная страница</h2>

            <p>Данное веб-приложение использует библиотеку React для создания пользовательского интерфейса.</p>

            <p>Приложение также взаимодействует с базой данных.</p>

            <Link to="/list"><button>Список объектов БД</button></Link>

            <FooterComp></FooterComp>

        </div>

    );

}

export default HomePage;

**Файл src/Pages/ListPage.js:**

import { useState, useEffect } from "react";

import { Link } from "react-router-dom";

import HeaderComp from "../components/HeaderComp";

import FooterComp from "../components/FooterComp";

import getObjects from "../modules/GetObjects.js";

function ListPage() {

    const [creatList, setCreatList] = useState([])

    const [creatNames, setCreatNames] = useState([])

    const handleObjectsList = async () => {

        const names = []

        const rases = await getObjects()

        for (let ras of rases) {

            names.push(ras['name']);

        }

        setRasesList(rases)

        setRasesNames(names)

    }

    useEffect(()=>{

        handleObjectsList()

    }, [])

    return (

        <div>

            <HeaderComp></HeaderComp>

            <h2>Список объектов Creat</h2>

            <ul>

                {creatNames.map((name)=>{

                    return (

                        <li key={name}>

                            <Link to={{pathname: "/list/object", data: rasesList.find(o => o.name == name)}}>{name}</Link>

                        </li>

                    )

                })}

            </ul>

            <FooterComp></FooterComp>

        </div>

    );

}

export default ListPage

**Файл src/Pages/ObjectPage.js:**

import { Link } from "react-router-dom";

import HeaderComp from "../components/HeaderComp";

import FooterComp from "../components/FooterComp";

function ObjectPage(params) {

    const rasesList = params.location.data

    return (

        <div>

            <HeaderComp></HeaderComp>

            <h2>{rasesList.name}</h2>

            <h4>Описание:</h4>

            <p>{rasesList.descript}</p>

            <Link to="/list"><button>Назад</button></Link>

            <FooterComp></FooterComp>

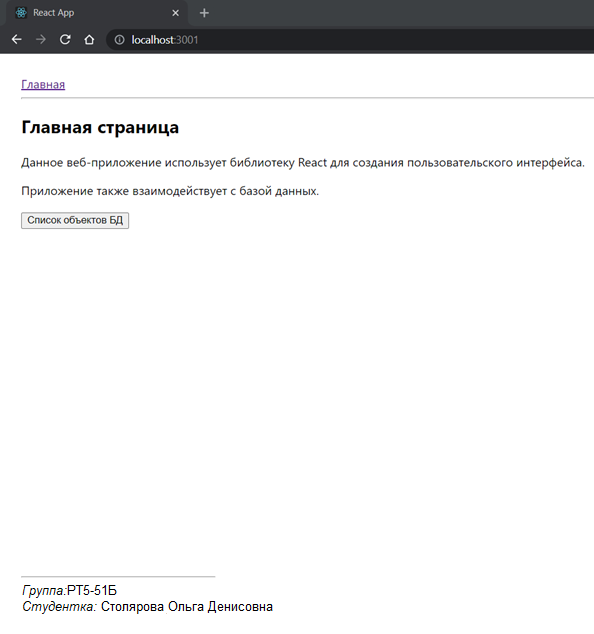
        </div>

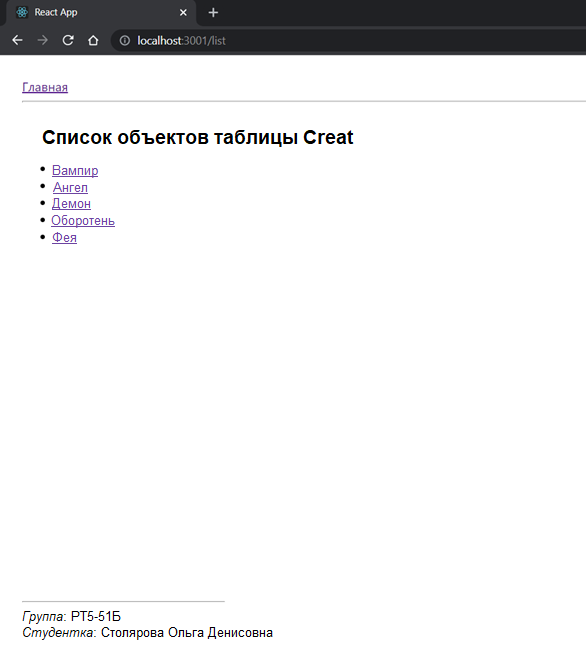
    );

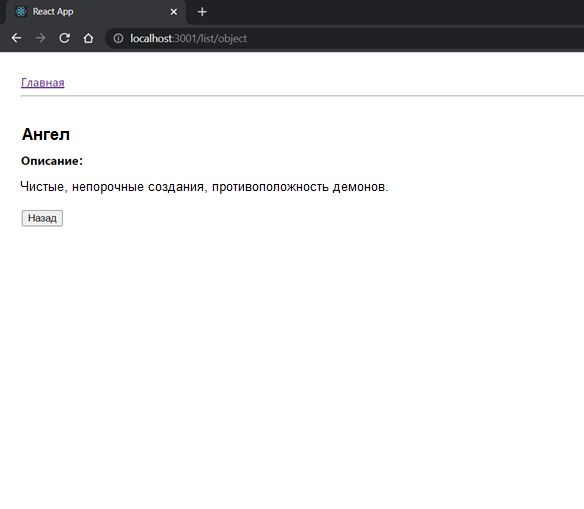
}

export default ObjectPage;

**Пример выполнения:**







Ссылка на GitHub

https://github.com/OlyaSto/Olyabmstu