

Московский государственный технический университет им. Н.Э. Баумана
Факультет «Информатика и системы управления»
Кафедра «Автоматизированные системы обработки информации и
управления»



Отчет
Лабораторная работа № 8
По курсу «Разработка интернет-приложений»
«React»

ИСПОЛНИТЕЛЬ:

Матиенко Андрей

Группа ИУ5-51

"__"_____ 2019 г.

ПРЕПОДАВАТЕЛЬ:

Гапанюк Ю.Е.

"__"_____ 2019 г.

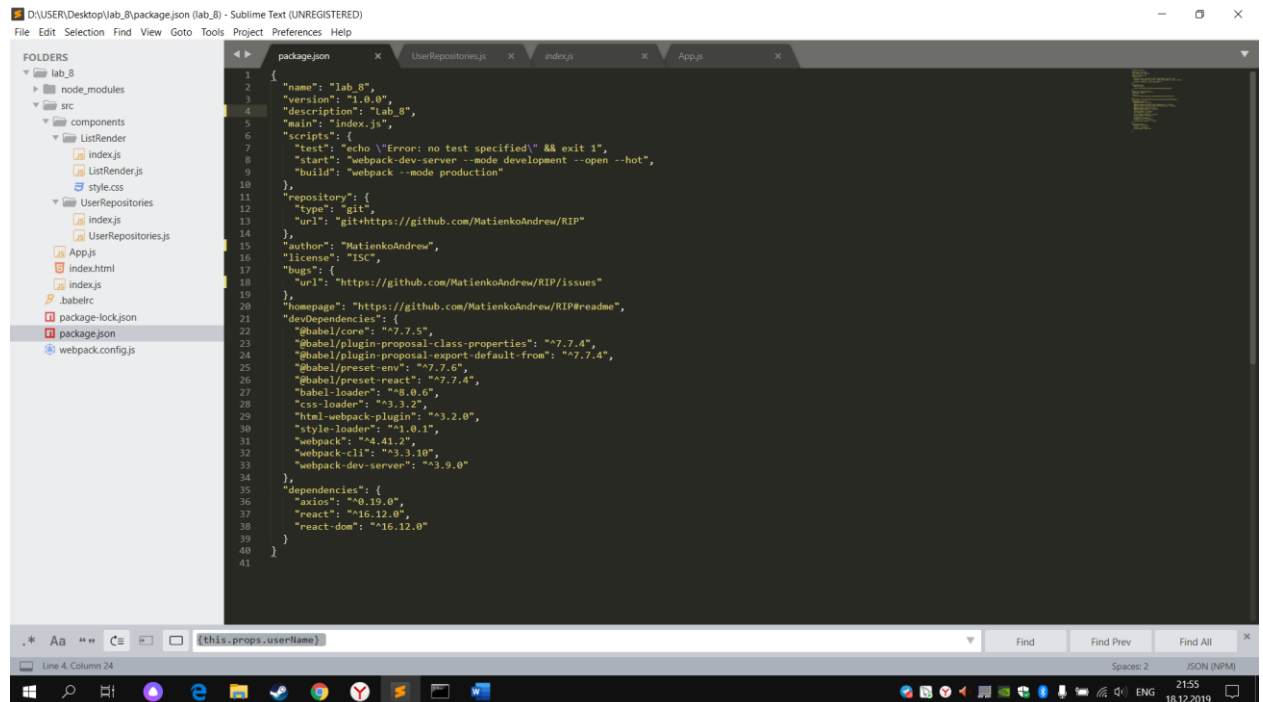
Москва 2019

Задание и порядок выполнения

В этой лабораторной работе вы создадите React-приложение. Вам нужно будет создать одну страницу со списком репозиториях на гитхабе.

Порядок выполнения работы:

1. Настроенный конфиг webpack и babel



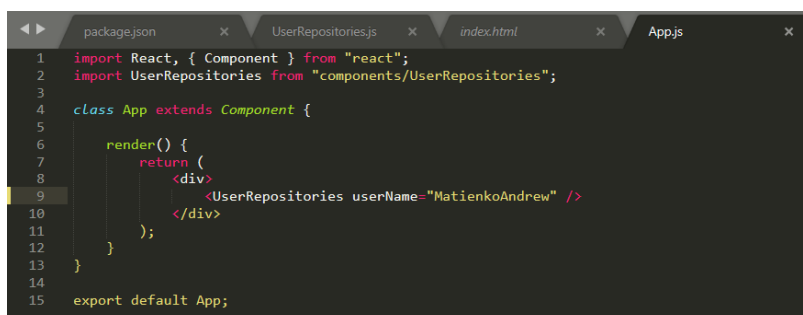
2. React-приложение с одной страницей.

Для запуска ввести команду в кс: *npm start*



MatienkoAndrew repositories:

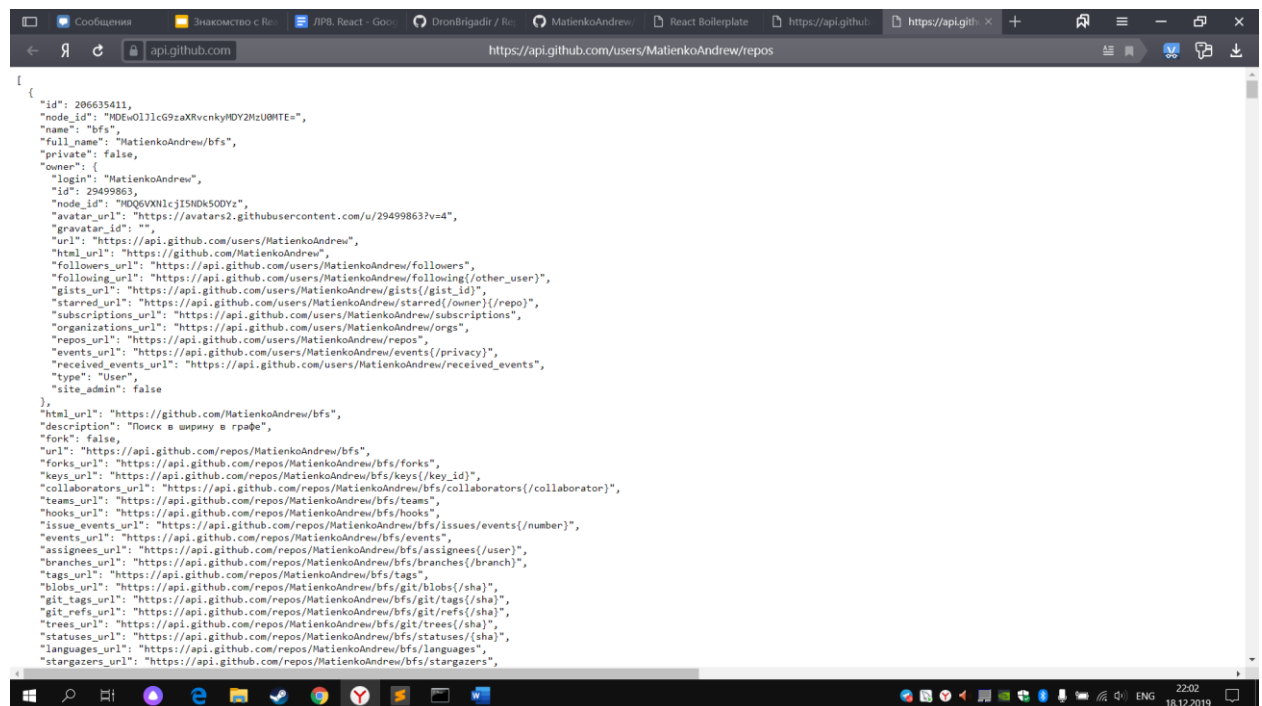
- bfs
- Common
- Day42
- Exam
- filler
- fillit
- ft_printf
- get_next_line
- iu5web-fall-2019
- labs
- lem_in
- Libft
- Networks-and-telecommunications
- push_swap
- reverse_num
- RIP
- yii



3. На странице должны отображаться репозитории с github. Репозитории должны загружаться с помощью axios через открытый api гитхаба.

<https://api.github.com/users/MatienkoAndrew/repos>

```
package.json x UserRepositories.js x index.js x App.js
1 import React, { Component } from "react";
2 import axios from 'axios';
3 import ListRender from "components/ListRender";
4
5 class UserRepositories extends Component {
6
7   state = {
8     repNames: []
9   }
10
11   componentDidMount() {
12     let url = "https://api.github.com/users/" + this.props.userName + "/repos";
13     axios.get(url).then(response => {
14       let repNames = response.data.map(item => item.name);
15       this.setState({repNames});
16     })
17   }
18
19   render() {
20
21     return (
22       <div>
23         <h1>{this.props.userName} repositories:</h1>
24         <ListRender data={this.state.repNames} />
25       </div>
26     );
27   }
28 }
29
30 export default UserRepositories;
```

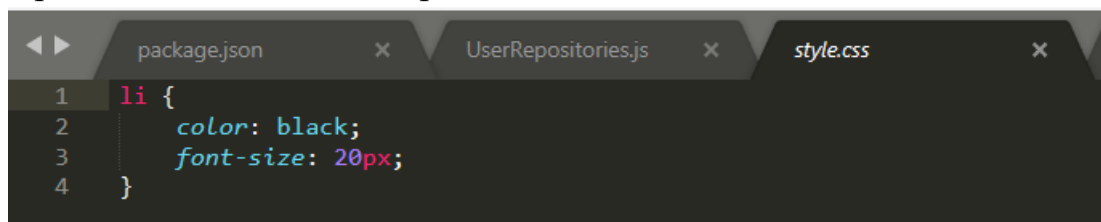


4. Приложение должно успешно собираться вебпаком



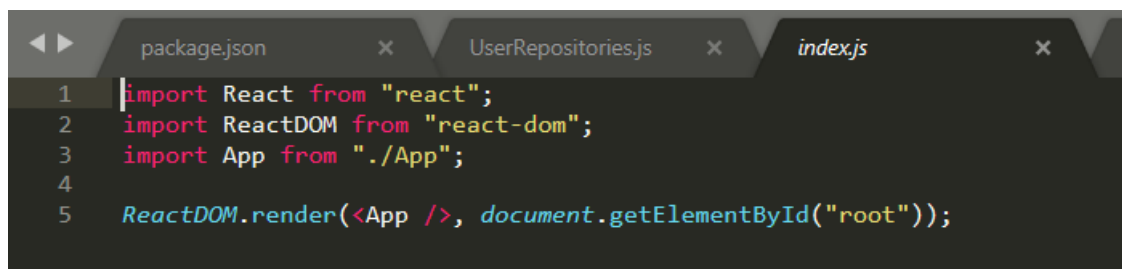
```
1
2 const path = require("path");
3 const HtmlWebpackPlugin = require("html-webpack-plugin");
4
5 module.exports = {
6   entry: "./src/index.js",
7   output: {
8     path: path.join(__dirname, "/dist"),
9     filename: "index-bundle.js"
10  },
11  module: {
12    rules: [
13      {
14        test: /\.js$/,
15        exclude: /node_modules/,
16        use: ["babel-loader"]
17      },
18      {
19        test: /\.css$/,
20        use: ["style-loader", "css-loader"]
21      }
22    ]
23  },
24  plugins: [
25    new HtmlWebpackPlugin({
26      template: "./src/index.html"
27    })
28  ],
29  resolve: {
30    alias: {
31      components: path.resolve(__dirname, 'src/components/')
32    }
33  }
34};
```

5. Приложение должно содержать css



```
1 li {
2   color: black;
3   font-size: 20px;
4 }
```

6. Приложение в dev режиме должно обновляться без перезагрузки страницы с помощью webpack-dev-server'a



```
1 import React from "react";
2 import ReactDOM from "react-dom";
3 import App from "./App";
4
5 ReactDOM.render(<App />, document.getElementById("root"));
```

7. index.html должен добавляться с помощью HtmlWebpackPlugin



The image shows a code editor with four tabs: package.json, UserRepositories.js, index.html, and App.js. The index.html tab is active, displaying the following HTML code:

```
1
2 <!DOCTYPE html>
3 <html lang="en">
4
5 <head>
6   <meta charset="UTF-8">
7   <meta name="viewport" content="width=device-width, initial-scale=1.0">
8   <meta http-equiv="X-UA-Compatible" content="ie=edge">
9   <title>React Boilerplate</title>
10 </head>
11
12 <body>
13   <div id="root">
14
15   </div>
16 </body>
17
18 </html>
```

Выводы:

Научились работе в React