



Московский государственный технический университет имени Н. Э. Баумана

Факультет «Информатика и системы управления»

Кафедра ИУ5

Отчёт по

Лабораторной работе 8

Выполнил:

Бахрамов Н.А

Группа ИУ5-53Б

Москва

2021

Задание:

На основе [методических указаний](#) 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 роутинг 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.

App.tsx

```
import React from 'react';
import './App.css';
import {Route,Switch} from 'react-router-dom'
import Homepage from './routes/Homepage';
import Content from './routes/SearchContent'
import PageNotFound from './routes/PageNotFound';
import logo from './ico/icons8-github-50.png';

const App: React.FC = () => {
  console.log();
  return (
    <>
      <header className='header'>
        <img src={logo} alt='logo' />
        <span className='title'>GitSearch</span>
      </header>
      <Switch>

        <Route exact path='/example_2/build/' component={Homepage} />
        <Route path='/example_2/build/search' component={Content}/>
        <Route path='/example_2/build/error' component={PageNotFound}/>

      </Switch>
    </>
  );
}

export default App;
```

Routes:

Homepage

```
import React from 'react'
import { useDispatch } from 'react-redux';
import { useHistory, withRouter } from 'react-router';
import { InputActionEnter } from '../store/input/action';
import './Homepage.css'

function HomePage() {
  const his = useHistory();
  const dispatch = useDispatch();

  const dat = (ev: React.KeyboardEvent<HTMLInputElement>) => {
    if( ev.key === 'Enter') {
      let str = document.getElementById('inp') as HTMLInputElement;
      dispatch(InputActionEnter(str.value));
      his.push('/example_2/build/search')
    }
  }

  const dat2 = () => {
    let str = document.getElementById('inp') as HTMLInputElement;
    dispatch(InputActionEnter(str.value));
    his.push('/example_2/build/search')
  }

  return(
    <div className='homepage'>
      <input
        className="inp_class"
        type='text'
        id='inp'
        placeholder='Введителя имя пользователя GitHub...'
        onPress={({ev})=>dat(ev)}
      />
      <button className='home_button' onClick={dat2}>ok</button>
    </div>
  );
}

export default withRouter(HomePage);
```

Content

```
function Content() {
  const state = useTypedSeceltonHook(store => store);
  const dispatch = useDispatch();
  const his = useHistory();

  useEffect(() => {
    dispatch(AppAction(state.inp.data));
  }, [dispatch, state.inp.data]);

  const back = useCallback(()=>{
    his.push('/example_2/build/');
  },[his]);

  if(state.content.err !== '')
    his.push('/example_2/build/error');

  if(state.content.loading) return(
    <div>
      <h1 className="load">Loading...</h1>
    </div>
  );

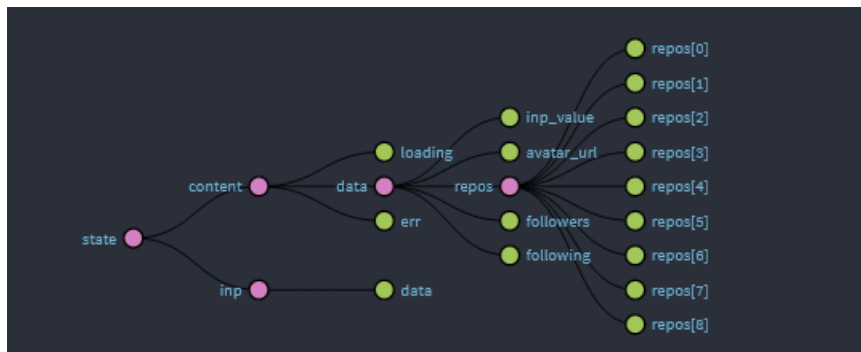
  return(
    <div className="content">
      <div className='prev'>
        <Avatar src={state.content.data.avatar_url}></Avatar>
        <div className='name'>{state.content.data.inp_value}</div>
        <div className='lable'>
          <span>followers: {state.content.data.followers}</span>
          <span>following: {state.content.data.following}</span>
        </div>
      </div>
      <div className='repo'>
        <label>repos: {state.content.data.repos?.length}</label>
        <div className='overflow-repos'>
          {state.content.data.repos?.map(val => (<div className='repo-lab'>{val}</div>))}
        </div>
      </div>
      <button className='back_button' onClick={back}>Назад к поиску</button>
    </div>
  );
}

export default withRouter(Content);
```

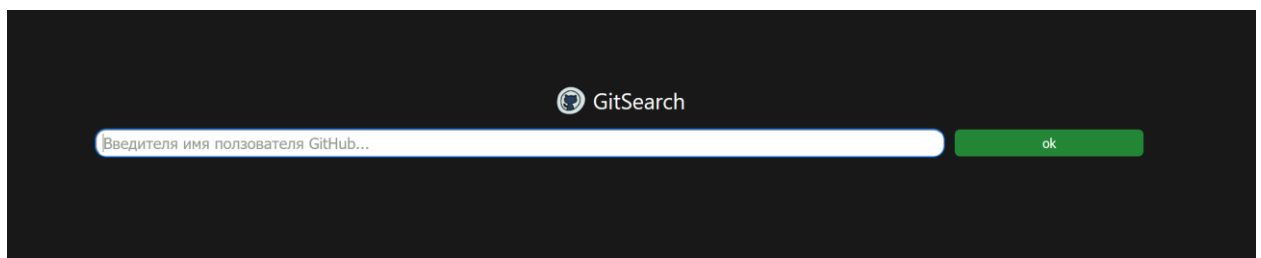
PageNotFound

```
function PageNotFound() {  
  const his = useHistory();  
  const dispatch = useDispatch();  
  
  const back = useCallback(()=>{  
    dispatch(BackAction());  
    his.push('/example_2/build/');  
  },[dispatch,his]);  
  
  return (  
    <div className="PageNotFound">  
      <span>Page not found: 404</span>  
      <button className='pnf_button' onClick={back}>Назад к поиску</button>  
    </div>  
  );  
}
```

Redux



Вид





eNBiOFF

followers: 0

following: 0

repos: 9

art_shop
dz_Sety
EazyTest
for_work
infoblog

[Назад к поиску](#)