**ОТЧЕТ**

по лабораторной работе №7

по курсу «Web-программирование»

на тему «Frontend на ReactJS»

Выполнили:

Студенты группы 16ВВ1

Кукушкин А. Ю.

Десятов Н. О.

Принял:

к.т.н., доцент Дубравин А.В.

Пенза 2020

Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

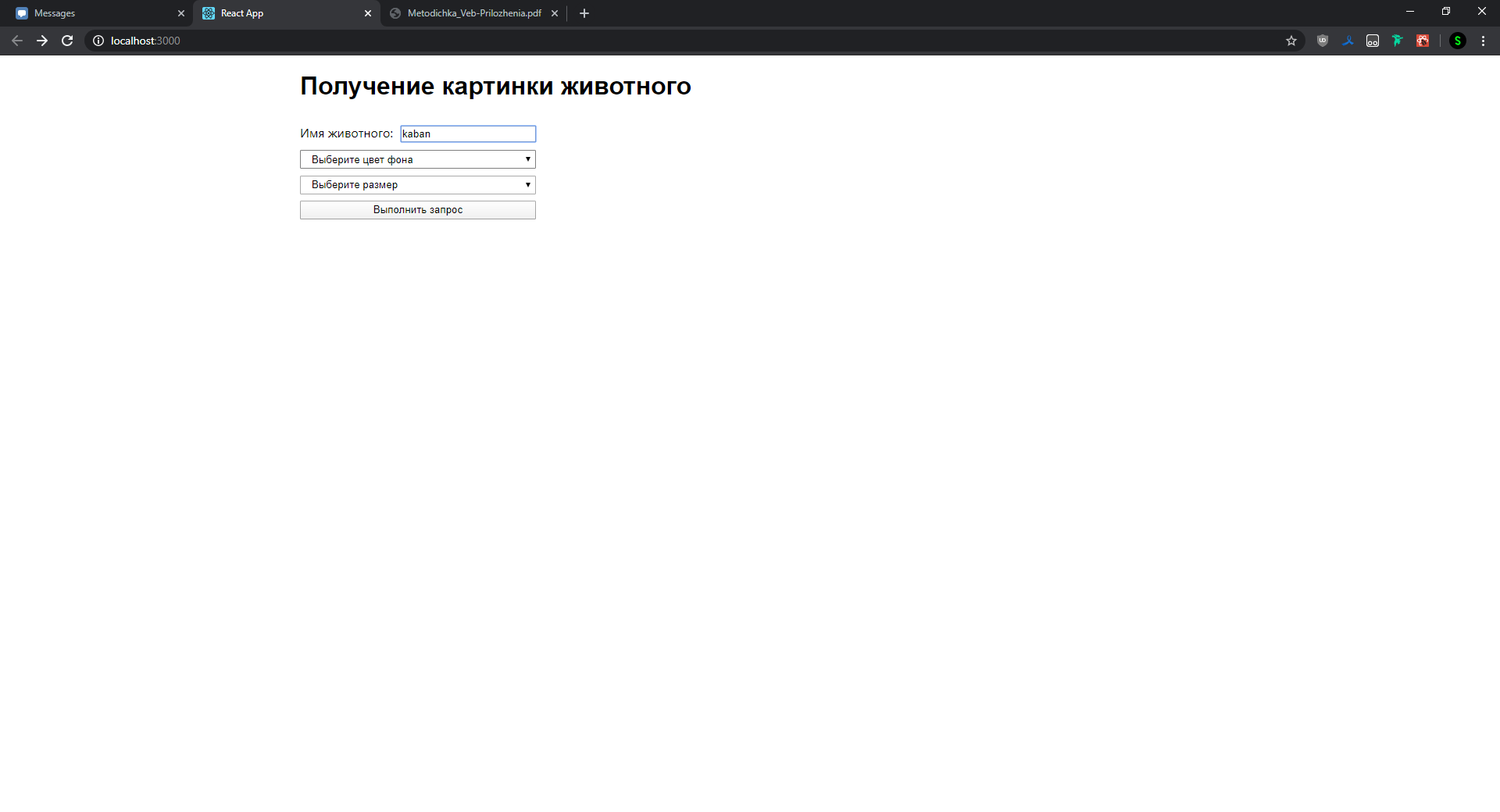
### Лабораторное задание.

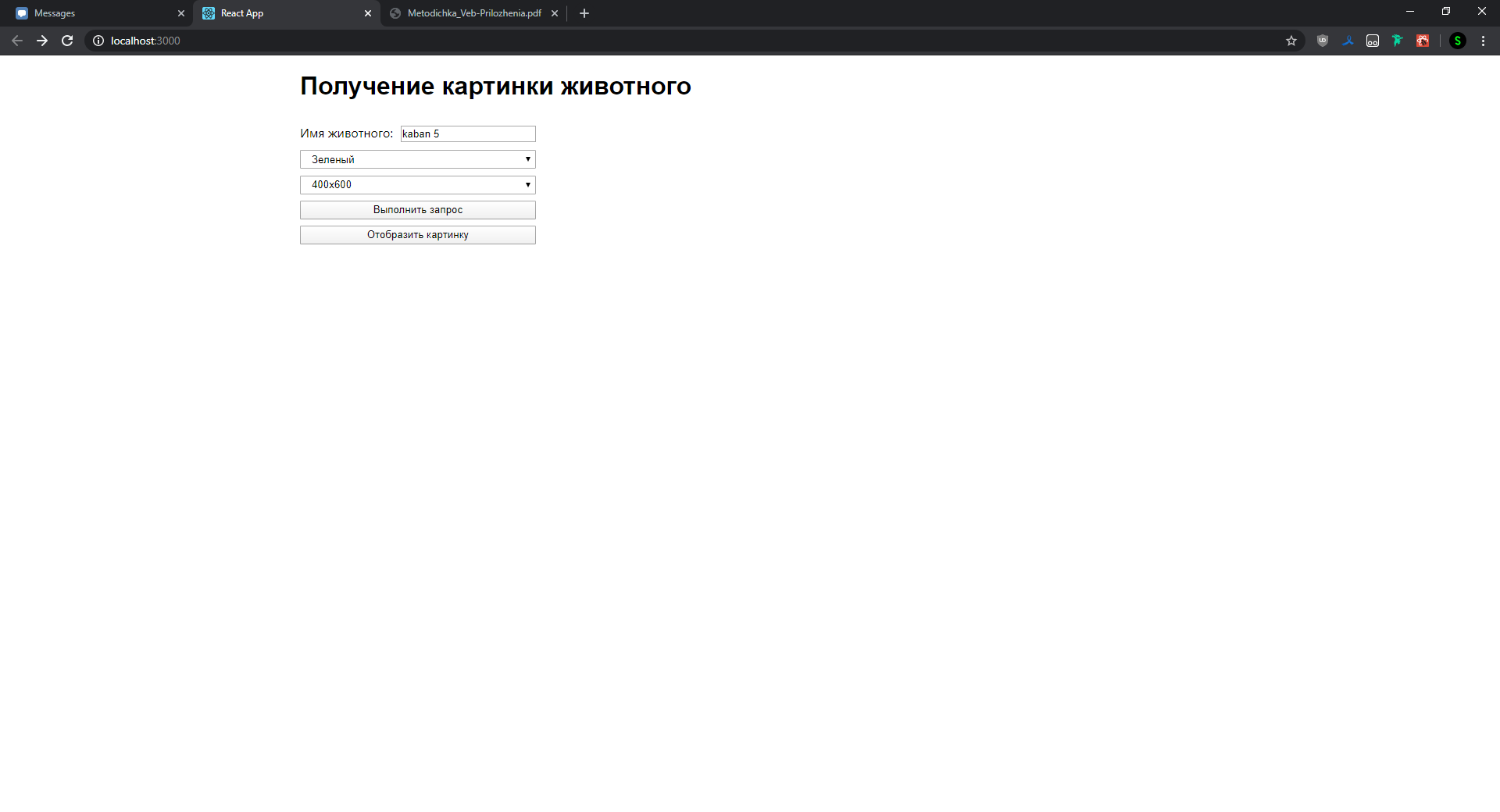
Необходимо для лабораторной работы №5 реализовать клиентское приложение с использованием технологи ReactJS. Необходимо в интерфейсе предусмотреть элементы управления для всех параметров, которые можно передать на сервер через REST API. Нужно не забыть, что часть параметров может быть опциональной. Так же нужно реализовать кнопку, которая должна активировать запрос к серверу и по результатам ответа изменить состояние элементов управления в соответствии с полученными значениями, а также заполнить поле с именем изображения и вместе с этим необходимо выполнить второй запрос к серверу, загрузить изображение по его имени и отобразить на странице.

### Результат работы.

Для корректной работы необходимо запустить сервер с API которого мы будем работать и написанное нами react приложение.

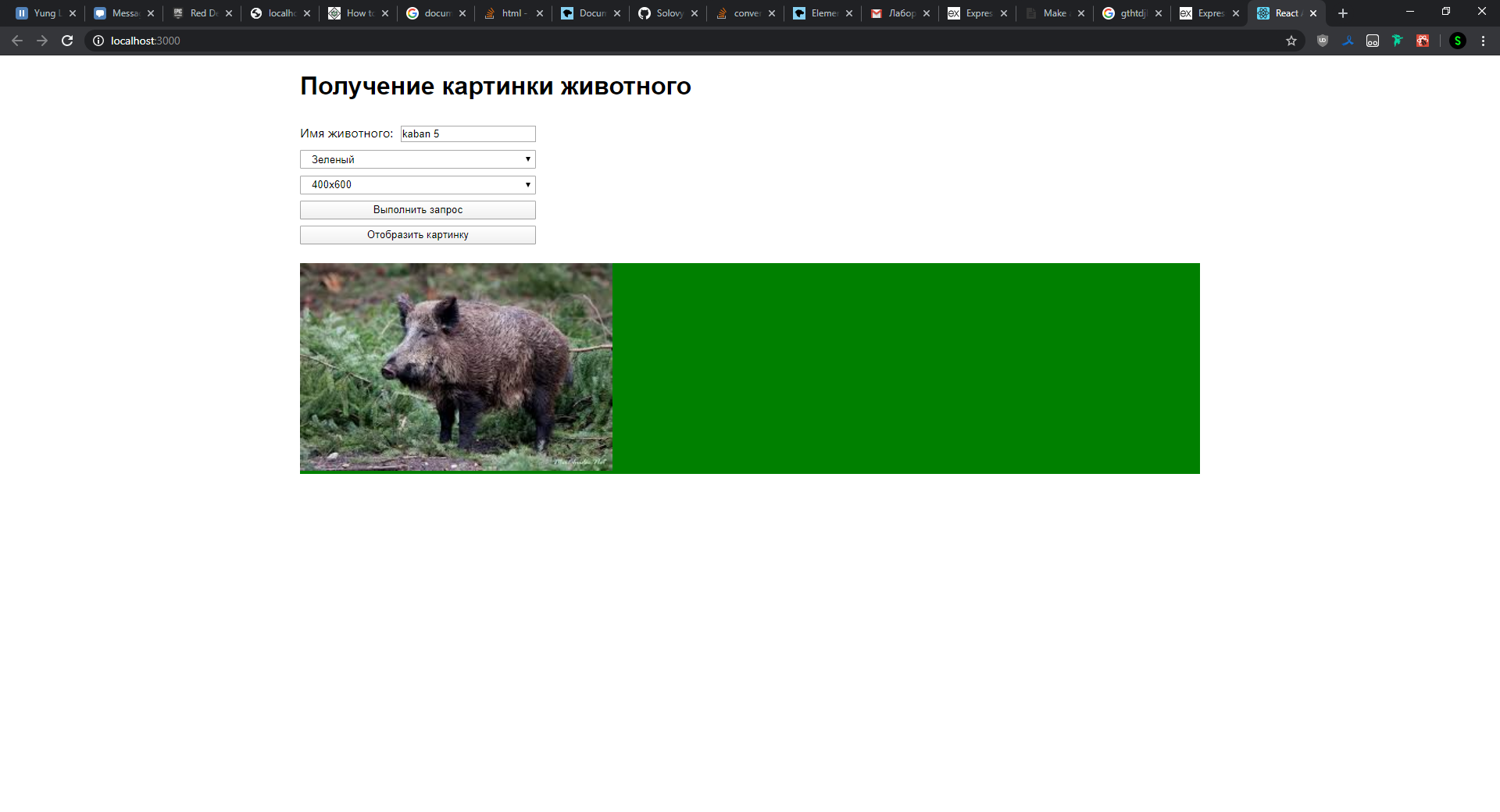
Необязательно заполнять все поля интерфейса, достаточно просто ввести имя животного в поле (можно без индекса, он заполнится тоже автоматически).





**Рисунок 1. Ввод имени и получение ответа.**

В это время произошел первый запрос к серверу, при успешном ответе появляется вторая кнопка, на основе данных выполняется второй запрос к серверу. В ответе приходит html код, упакованный в json объект, который и отображается внизу формы.



**Рисунок 2. Интерфейс после второго запроса.**

Мы видим, что сервер вернул, что и требовалось, 5-ое изображение кабана размером 400x600 на зеленом фоне.

### Выводы

В ходе данной лабораторной работы получили ценные практические навыки написания frontend при помощи React JS, закрепили навыки работы с REST API.

### Листинг

#### React приложение - App.js

import React from 'react';

import AmazingForm from './components/amazingForm';

const styles = {

  App: {

    backgroundColor: '#fff',

    margin: '0 20%',

  },

}

function App() {

  return (

    <div style={styles.App} className="App">

      <h1 style={{fontFamily: 'Arial'}}>Получение картинки животного</h1>

      <AmazingForm />

    </div>

  );

}

export default App;

#### React приложение - amazingForm.js

import React from 'react';

import PhotoPreview from './photoPreview';

export default class AmazingForm extends React.Component {

    constructor(props) {

        super(props);

        this.state = {

            name: '',

            color: 'Выберите цвет',

            size: 'Выберите размер',

            formIsFull: false,

        };

    }

    RecieveAnswer(event) {

        event.preventDefault();

        let reqBody = {

            colorSelector: this.state.color,

            text: this.state.name,

            sizeSelector: this.state.size,

        }

        fetch('http://localhost:8888/getPicture', {

            method: 'POST',

            headers: {

                'Content-Type': 'application/json'

            },

            body: JSON.stringify(reqBody)

        })

            .then(response => response.json())

            .then(result => {

                console.log(result);

                if (result.name === 'not found') {

                    this.setState({ name: result.name });

                    return;

                }

                this.setState({name: result.name.join(' ')})

                if (result.color === '#FF0000') { this.setState({ color: 'red' }) }

                if (result.color === '#00FF00') { this.setState({ color: 'green' }) }

                if (result.color === '#FF8000') { this.setState({ color: 'orange' }) }

                if (result.size.width === '400') { this.setState({ size: '400х600' }) }

                if (result.size.width === '800') { this.setState({ size: '800х600' }) }

                if (result.size.width === '1280') { this.setState({ size: '1280х960' }) }

                this.setState({ formIsFull: true });

            });

    }

    render() {

        return (

            <div>

                <form onSubmit={event => this.RecieveAnswer(event)}>

                    <label>

                        Имя животного:

                    <input pattern={'[a-zA-Z]{1,8}\\s?\\d?'} required={true} value={this.state.name} onChange={event => this.setState({ name: event.target.value })} style={{ margin: '10px' }} type='text' />

                    </label>

                    <div>

                        <select value={this.state.color} onChange={event => this.setState({ color: event.target.value })} style={styles.selector}>

                            <option hidden={true} value="default">Выберите цвет фона</option>

                            <option value="red">Красный</option>

                            <option value="green">Зеленый</option>

                            <option value="orange">Оранжевый</option>

                        </select>

                    </div>

                    <div style={{ marginTop: '8px' }}>

                        <select value={this.state.size} onChange={event => this.setState({ size: event.target.value })} style={styles.selector}>

                            <option hidden={true} value="default">Выберите размер</option>

                            <option value="400х600">400х600</option>

                            <option value="800х600">800х600</option>

                            <option value="1280х960">1280х960</option>

                        </select>

                    </div>

                    <input style={styles.button} type="submit" value="Выполнить запрос" />

                </form>

                <PhotoPreview color={this.state.color}

                    id={this.state.name.trim().split(' ')[1]}

                    name={this.state.name.trim().split(' ')[0]}

                    size={this.state.size}

                    visible={this.state.formIsFull} />

            </div>

        )

    }

}

const styles = {

    selector: {

        width: '302px',

        height: '24px',

        paddingLeft: '10px'

    },

    button: {

        width: '302px',

        height: '24px',

        marginTop: '8px'

    }

}

#### React приложение - photoPreview.js

import React from 'react';

export default class PhotoPreview extends React.Component {

    constructor(props) {

        super(props);

        this.state = {

            name: '',

            color: 'Выберите цвет',

            size: 'Выберите размер',

            resultIsRecieved: false,

            recievedPage: 'default'

        };

    }

    recievePicture(event) {

        event.preventDefault();

        fetch('http://localhost:8888/picture/' + this.props.name + '/' + this.props.id + '/' + this.props.color + '/' + this.props.size, {

            method: 'GET',

        })

            .then(response => response.json())

            .then(result => {

                console.log(result);

                this.setState({ recievedPage: result.page });

                this.setState({ resultIsRecieved: true });

            });

    }

    render() {

        if (this.state.resultIsRecieved) {

            return <div dangerouslySetInnerHTML={{\_\_html: this.state.recievedPage}} />

        }

        return (

            <div style={this.props.visible ? null : styles.body}>

                <button onClick={event => this.recievePicture(event)} style={styles.button}>Отобразить картинку</button>

            </div>

        )

    }

}

const styles = {

    body: {

        display: 'none',

    },

    button: {

        width: '302px',

        height: '24px',

        marginTop: '8px'

    }

}