Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №7

по курсу «Web-программирование»

на тему «Frontend на ReactJS»

Выполнили:

студенты группы 16ВВ2

Гогулевский А.Э.

Кузнецов А.С.

Мультяева И.И.

Приняли:

к.т.н., доцент Дубравин А.В.

к.т.н., доцент Карамышева Н.С.

Пенза 2020

**Цель работы.**

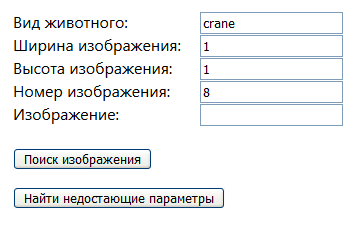
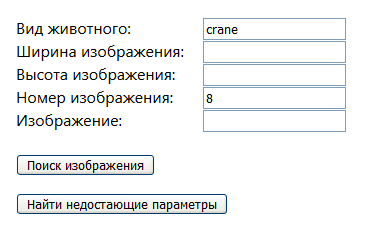
Реализовать Frontend на ReactJS.

**Задание.**

Необходимо для лабораторной работы №5 реализовать клиентское приложение с использованием технологи ReactJS.

|  |  |
| --- | --- |
| **Вариант** | **Животные** |
| 7 | Животные: журавль, жаба, жираф. |

**Результат работы.**

Рисунок 1 – Получение параметров с сервера

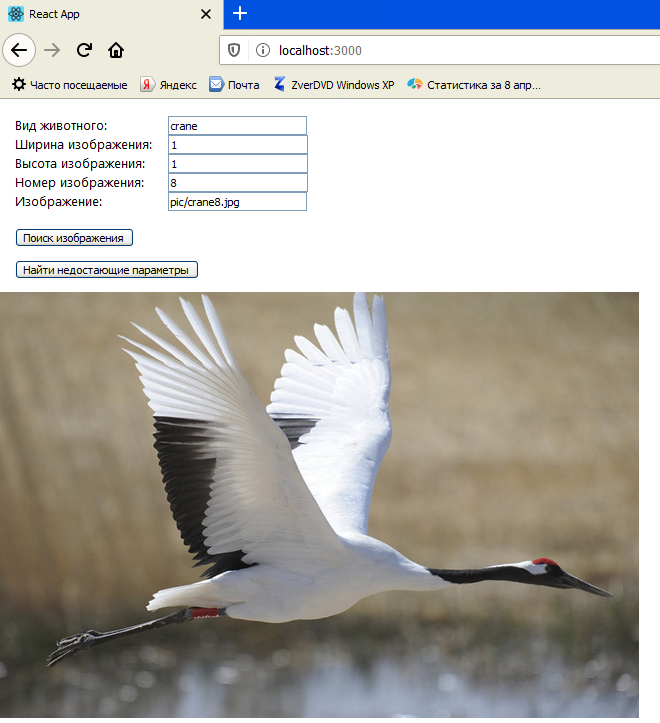


Рисунок 2 – Поиск изображения

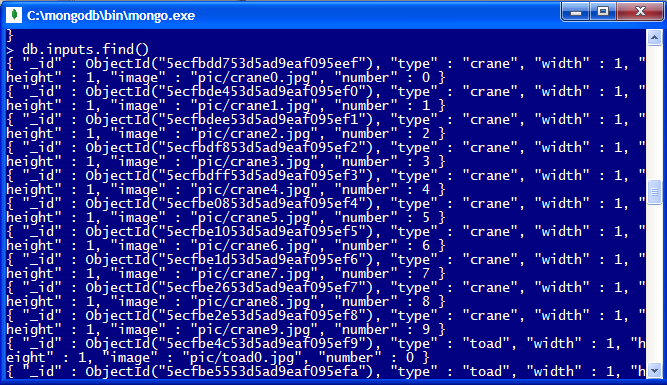


Рисунок 3 – Запись в БД

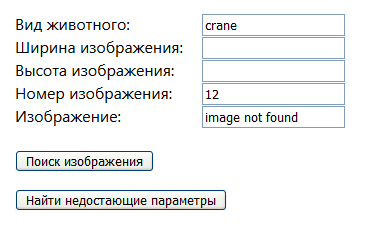


Рисунок 4 – Результат ввода неверных данных

**Листинг сервера**

**index.js**

const express = require("express");

const mongoose = require("mongoose");

const bodyParser = require("body-parser");

const exp = express();

mongoose.connect("mongodb://localhost/inputdb",

{

useNewUrlParser: true,

useUnifiedTopology: true

}).then(() => { console.log("Connected to database.")});

exp.use('/pic', express.static('pic'));

exp.use(bodyParser.json());

exp.use("/api",require("./api"));

app.listen(8888, ()=>{ console.log("Server has started.");});

**api.js**

const express = require("express");

var fs = require("fs");

let jsonbody = null;

const router = express.Router();

const Input = require("./input");

router.get("/input", (req, res)=>{

console.log(jsonbody);

res.send(jsonbody);

});

router.post("/find", (req, res)=>{

console.log(req.body);

var type = req.body.type;

var number = req.body.number

let error = null;

if (number == ""){ number = Math.floor(Math.random() \* 9);}

Input.findOne({type: type, number: number})

.then(input =>

{

if (input != null)

{

res.send({type: input.type, width: input.width, height: input.height, number: input.number});

}

else

{

error = {error : "image not found"};

res.send(error);

}

});

});

router.post("/find", (req, res)=>{

console.log(req.body);

var type = req.body.type;

var width = req.body.width

var height = req.body.height;

var number = req.body.number

let error = null;

if (width == "") { width = 1;}

if (height == ""){ height = 1;}

Input.findOne({type: type, number: number})

.then(input =>

{

if (input != null)

{

Input.findOne({width: width, height: height})

.then(input =>

{

if(input != null)

{

jsonbody = input;

res.redirect("./input");

}

else

{

finderror = {error : "Не найдено изображение животного с указанными параметрами 'ширина' или 'высота'."};

res.send(finderror);

}

});

}

else

{

finderror = {error : "Не найдено изображение животного с указанными параметрами 'вид' или 'номер'."};

res.send(finderror);

}

});

});

module.exports = router;

**input.js**

const mongoose = require("mongoose");

const Schema = mongoose.Schema;

const inputSchema = new Schema({

type: String,

width: Number,

height: Number,

image: String,

value: Number

});

const input = mongoose.model("input",inputSchema);

module.exports = input;

**Листинг клиента**

**App.js**

import React, { useState } from 'react';

function App() {

const [type, setType] = useState('')

const [width, setWidth] = useState('')

const [height, setHeight] = useState('')

const [image, setImage] = useState('')

const [img, setImg] = useState('')

const [number, setNumber] = useState('')

function startSearch(event) {

event.preventDefault()

if (type.trim()) {

var input\_type = null;

var input\_number = null;

input\_type = type;

input\_number = number;

let inputsJSON = {

type: input\_type,

number: input\_number,

};

console.log(inputsJSON);

fetch('/api/find', {

method: 'POST',

headers: {

'Content-Type': 'application/json'

},

body: JSON.stringify(inputsJSON)

})

.then(response => {

if(!response.ok) {

console.log(response);

throw new Error(response.statusText)

}

return response;

})

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

.then(result => {

console.log(result)

if (result.error !== "image not found")

{

setType(result.type)

setWidth(result.width)

setHeight(result.height)

setImage('')

}

else

setImage('nothing in database')

});

}

}

function sendReq(event) {

event.preventDefault()

if (species.trim()) {

var in\_type = null;

var in\_width = null;

var in\_height = null;

var in\_number = null;

var in\_type = type;

var in\_width = width;

var in\_height = height;

var in\_number = number;

let inputsJSON = {

type: in\_type,

width: in\_width,

height: in\_height,

number: in\_number

};

console.log(inputsJSON)

fetch('/api/input', {

method: 'POST',

headers: {

'Content-Type': 'application/json'

},

body: JSON.stringify(animalsJSON)

})

.then(response => {

if(!response.ok) {

throw new Error(response.statusText)

}

return response;

})

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

.then(result => {

console.log(result)

if (result.image != null)

{

setImage(result.image)

setImg(result.image)

}

else

setImage('image not found')

});

}

}

return (

<div className="wrapper">

<label>

<br></br>

<a style={{marginLeft: 20}}>Вид животного: </a>

<input

style={{marginLeft: 68}}

value={species}

onChange={event => setSpecies(event.target.value)}

/>

<br></br>

<a style={{marginLeft: 20}}>Ширина изображения: </a>

<input

style={{marginLeft: 15}}

value={width}

onChange={event => setWidth(event.target.value)}

/>

<br></br>

<a style={{marginLeft: 20}}>Высота изображения: </a>

<input

style={{marginLeft: 23}}

value={height}

onChange={event => setHeight(event.target.value)}

/>

<br></br>

<a style={{marginLeft: 20}}>Номер изображения: </a>

<input

style={{marginLeft: 26}}

value={value}

onChange={event => setValue(event.target.value)}

/>

<br></br>

<a style={{marginLeft: 20}}>Изображение: </a>

<input

style={{marginLeft: 77}}

value={image}

onChange={event => setImage(event.target.value)}

/>

<br></br>

<br></br>

<input

type="submit"

style={{marginLeft: 20}}

value="Поиск изображения"

onClick={sendReq}

/>

<p></p>

<input

type="submit"

style={{marginLeft: 20}}

value="Найти недостающие параметры"

onClick={startSearch}

/><p></p>

<img

src={img}

onChange={event => setImg(event.target.value)}

/>

</label>

</div>

);

}

export default App;

**Вывод**

В ходе выполнения лабораторной работы мы научились реализовывать Frontend на ReactJS.