Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №6

по курсу «Web-программирование»

на тему «Реализация REST API»

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 REST API.

**Задание.**

Необходимо доработать лабораторную работу №5 реализовав в ней поддержку REST API и реализовать два обработчика:

1. Получает запрос с названием животного и дополнительными параметрами и подбирает по этому запросу подходящее изображение из числа имеющихся;

2. По имени и параметрам изображения возвращает картинку.

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

**Порядок выполнения работы**

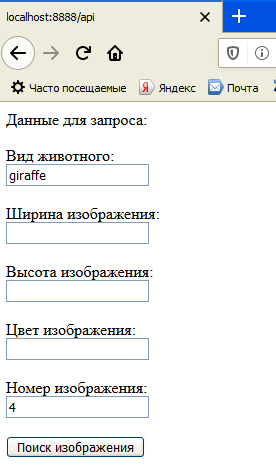


Рисунок 1– API

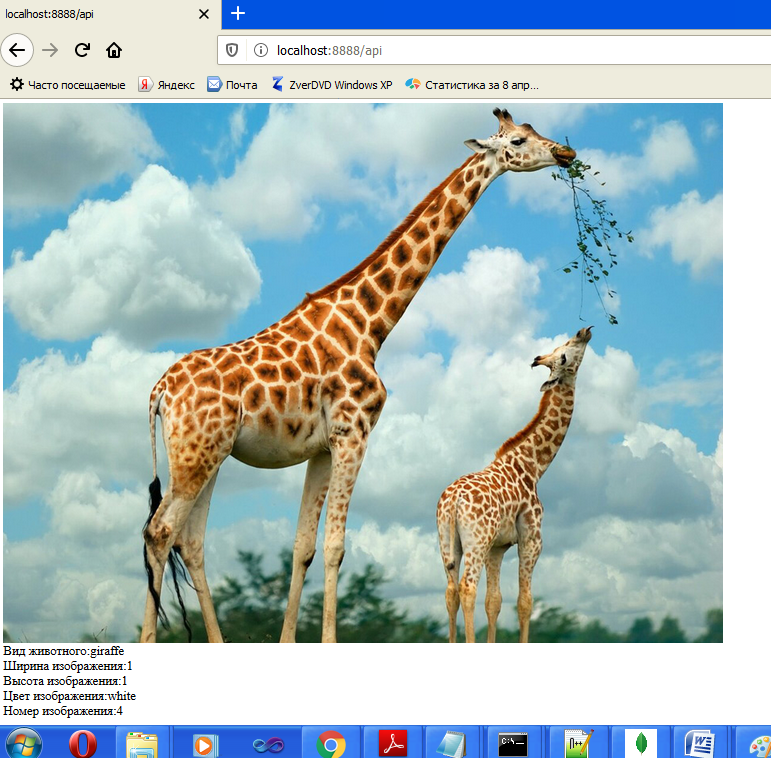


Рисунок 2 - Результат работы

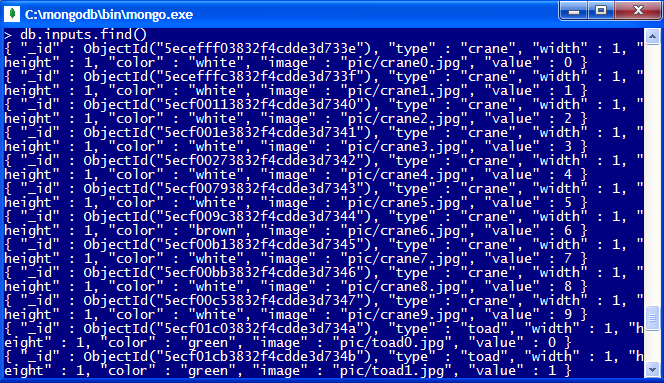


Рисунок3 - База данных

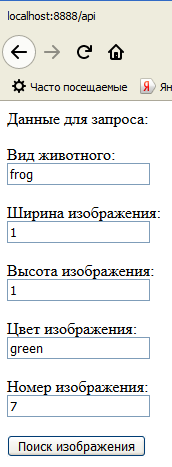


Рисунок 4 - ввод названия с ошибкой



Рисунок 5 - Ошибка

**Листинг.**

**main.html**

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />

</head>

<body>

<a>Данные для запроса:</a>

<br><br>

<a>Вид животного:</a>

<br>

<input id="species"></input>

<br><br>

<a>Ширина изображения:</a>

<br>

<input id="width"></input>

<br><br>

<a>Высота изображения:</a>

<br>

<input id="height"></input>

<br><br>

<a>Цвет изображения:</a>

<br>

<input id="color"></input>

<br><br>

<a>Номер изображения:</a>

<br>

<input id="number"></input>

<br><br>

<input type="submit" value="Поиск изображения" onclick="find()"/>

</body>

<script>

async function find() {

var img\_s= null;

var img\_w = null;

var img\_h = null;

var img\_c = null;

var img\_n = null;

var img\_i = null;

img\_s = document.getElementById("species").value;

img\_w = document.getElementById("width").value;

img\_h = document.getElementById("height").value;

img\_c = document.getElementById("color").value;

img\_n = document.getElementById("number").value;

img\_i = "pic/" + String(img\_s) + String(img\_n) + ".jpg";

let inputsJSON = {

species: img\_s,

width: img\_w,

height: img\_h,

color: img\_c,

image: img\_i,

value: img\_n

};

let responce = await fetch('./api/input',{

method: 'POST',

headers :{

'Content-Type': 'application/json'

},

body: JSON.stringify(inputsJSON)

});

const json = await responce.json();

console.log(json);

// Ошибка

if (json.error != null)

{

console.log(json.error);

var pjson = JSON.stringify(json);

document.write(pjson);

}

else

{

document.write('<img id="reqImg" src ="">');

console.log(json.image);

document.getElementById("reqImg").src = json.image;

delete json.\_id;

delete json.\_\_v;

document.write('<br><a>Вид животного:</a>' + json.species);

document.write('<br><a>Ширина изображения:</a>' + json.width);

document.write('<br><a>Высота изображения:</a>' + json.height);

document.write('<br><a>Цвет изображения:</a>' + json.color);

document.write('<br><a>Номер изображения:</a>' + json.value);

}

}

</script>

</html>

**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("/", (req, res) => {

var page = fs.readFileSync("main.html");

res.writeHead(200, {"Content-Type": "text/html"});

res.write(page);

res.end();

})

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

console.log(jsonbody);

res.send(jsonbody);

});

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

console.log(req.body);

var type = req.body.type;

var width = req.body.width

var height = req.body.height;

var color = req.body.color;

var number = req.body.number

let error = null;

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

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

if (color == "") { color = "white";}

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

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

.then(input =>

{

if (input != null)

{

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

.then(input =>

{

if(input != null)

{

Input.findOne({color: color})

.then(input =>

{

if(input != null)

{

jsonbody = input;

res.redirect("./input");

}

else

{

error = {error : "Не найдено изображение животного с указанным параметром 'цвет'."};

res.send(finderror);

}

});

}

else

{

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

res.send(finderror);

}

});

}

else

{

error = {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,

color: String,

image: String,

value: Number

});

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

module.exports = input;

**Вывод.**

В ходе выполнения лабораторной работы научились реализовывать REST API.