Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

Выполнили:

cтуденты группы 16ВВ2  
Боровов М.С.  
Белов В.А.

Приняли:

к.т.н. Дубравин А.В.  
Кармышева Н.С.

## Отчёт

## по дисциплине «Web-программирование»

## на тему «Frontend на ReactJS»

Лабораторная работа № 7

Пенза 2020

**Цель работы**

Реализовать Frontend на ReactJS.

**Задание**

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

|  |  |
| --- | --- |
| **Вариант** | **Животные** |
| 4 | Гиена, газель, геккон |

**Листинг сервера**

**index.js**

const express = require("express");

const mongoose = require("mongoose");

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

const app = express();

mongoose.connect("mongodb://localhost/beast-db",

{

    useNewUrlParser: true,

    useUnifiedTopology: true

});

app.use('/img', express.static('img'));

app.use(bodyParser.json());

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

app.listen(8888, ()=>{

    console.log("Server started");

});

**api.js**

const express = require("express");

var fs = require("fs");

const router = express.Router();

const Beast = require("./beast");

let jsonbody = null;

router.get("/beast", (req, res)=>{

    console.log(jsonbody);

    res.send(jsonbody);

});

router.post("/find", (req, res)=>{

    console.log("POST is reached");

    console.log(req.body);

    var name = req.body.name;

    var value = req.body.value;

    if (value === '')

    {

        value = Math.floor(Math.random() \* 9);

    }

    Beast.findOne({name: name, value: value})

        .then(beast => {

            if (beast != null)

            {

                console.log("");

                console.log(beast);

                res.send({name: beast.name, value: beast.value, id: beast.ID, food: beast.food, class: beast.class});

            }

            else

            {

                imgerror = {error: "Image not found"};

                res.send(imgerror);

            }

        })

});

router.post("/beast", (req, res)=>{

    var name = req.body.name;

    var food = req.body.food

    var bclass = req.body.class;

    var ID = req.body.ID;

    var number = req.body.number

    let finderror = null;

    if (food == "") {

        food = "predator";

    }

    if (bclass == ""){

        bclass = "mammal";

    }

    if (ID == ""){

        finderror = {error: "ID Error", name: name, number: number };

        res.send(finderror);

    }

    Beast.findOne({name: name, number: number})

    .then(beast => {

        if (beast != null)

        {

            Beast.findOne({food: food})

            .then(beast => {

                if (beast != null)

                {

                    Beast.findOne({class: bclass})

                    .then(beast => {

                        if (beast != null)

                        {

                            Beast.findOne({ID: ID})

                            .then(beast => {

                                if (beast != null)

                                {

                                    jsonbody = beast;

                                    res.redirect("./beast");

                                }

                                else

                                {

                                    finderror = {error: "ID Error", name: name, number: number };

                                    res.send(finderror);

                                }

                            })

                        }

                        else

                        {

                            finderror={error:"Class Error",name:name,number:number};

                            res.send(finderror);

                        }

                    })

                }

                else

                {

                    finderror={error:"Food Error",name:name,number:number };

                    res.send(finderror);

                }

            })

        }

        else

        {

            finderror = {error: "Name Error", name: name, number: number };

            res.send(finderror);

        }

    });

});

module.exports = router;

**beast.js**

const mongoose = require("mongoose");

const Schema = mongoose.Schema;

const beastSchema = new Schema({

    name: String,

    food: String,

    class: String,

    ID: String,

    image: String,

    value: Number

});

const beast = mongoose.model("beast",beastSchema);

module.exports = beast;

**Листинг клиента**

**App.js**

import React, { useState } from 'react';

function App() {

  const [name, setName] = useState('')

  const [value, setValue] = useState('')

  const [food, setFood] = useState('')

  const [aclass, setClass] = useState('')

  const [ID, setID] = useState('')

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

  const [img, setImg] = useState('')

  /// Функция получения всех параметров

  function startSearch(event) {

    event.preventDefault()

    if (name.trim()) {

      var beastname = null;

      var beastvalue = null;

      beastname = name;

      beastvalue = value;

      let animanlsJSON = {

        name: beastname,

        value: beastvalue,

      };

      console.log(animanlsJSON);

      fetch('/api/find', {

        method: 'POST',

        headers: {

          'Content-Type': 'application/json'

        },

        body: JSON.stringify(animanlsJSON)

        })

        .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")

              {

                console.log("RESULT:");

                console.log(result);

                setValue(result.value)

                setFood(result.food)

                setClass(result.class)

                setID(result.id)

                setImage('')

              }

              else

              setImage('Invalid record')

            });

    }

  }

  /// Функция получения картинки

  function sendReq(event) {

    event.preventDefault()

    if (name.trim()) {

      var beastname = null;

      var beastvalue = null;

      var imgfood = null;

      var imgclass = null;

      var imgID = null;

      beastname = name;

      beastvalue = value;

      imgfood = food;

      imgclass = aclass;

      imgID = ID;

      let beastsJSON = {

        name: beastname,

        value: beastvalue,

        food: imgfood,

        class: imgclass,

        ID: imgID

      };

      console.log("GET IMAGE:")

      console.log(beastsJSON)

      fetch('/api/beast', {

        method: 'POST',

        headers: {

          'Content-Type': 'application/json'

        },

        body: JSON.stringify(beastsJSON)

        })

        .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("img/" + result.image)

              }

              else

              {

                setImage('Image not found');

              }

            });

    }

  }

  return (

    <div className="wrapper">

      <label>

          <a>Название</a>

          <a style={{marginLeft: 115}}>Значение</a>

          <br></br>

          <input

            value={name}

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

          />

          <input

            style={{marginLeft: 15}}

            value={value}

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

          /><p></p>

          <a>Способ питания</a>

          <a style={{marginLeft: 67}}>Класс</a>

          <br></br>

          <input

            value={food}

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

          ></input>

          <input

            style={{marginLeft: 15}}

            value={aclass}

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

          ></input><p></p>

          <a>ID</a>

          <a style = {{marginLeft: 170}}>Изображение</a>

          <br></br>

          <input

            value={ID}

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

          ></input>

          <input style = {{marginLeft: 15}}

            value={image}

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

          ></input><p></p>

          <input

            type="submit"

            value="Поиск"

            onClick={sendReq}

          />

          <input style = {{marginLeft: 15}}

            type="submit"

            value="Получить параметры"

            onClick={startSearch}

          /><p></p>

        <img

          src={img}

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

        />

      </label>

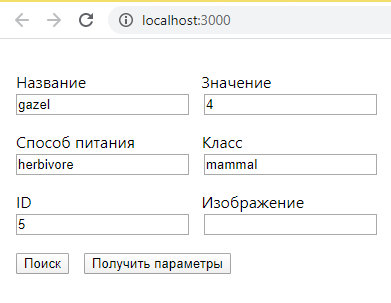
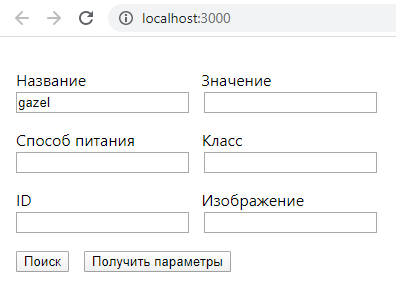
    </div>

  );

}

export default App;

**Результат работы**

Рисунок 1 – Получение параметров с сервера

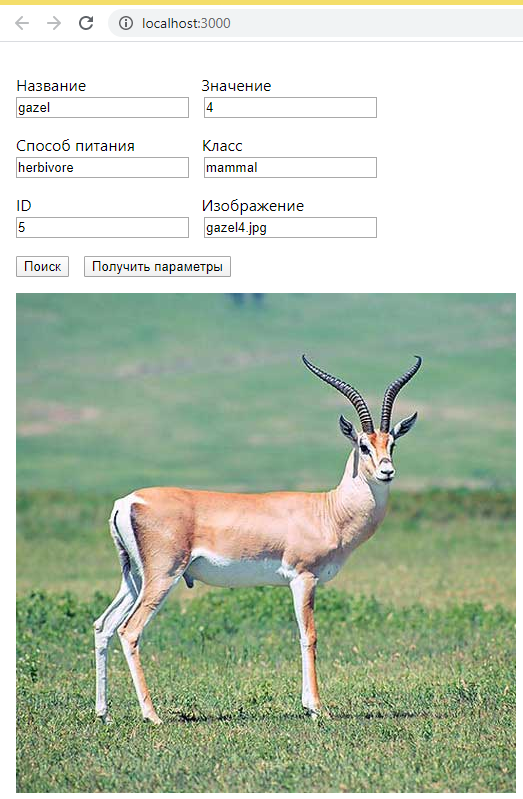


Рисунок 2 – Поиск изображения

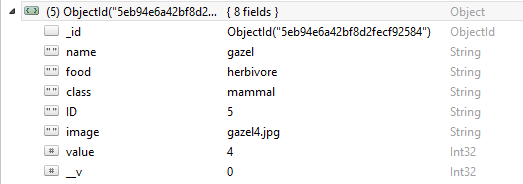


Рисунок 3 – Запись в БД

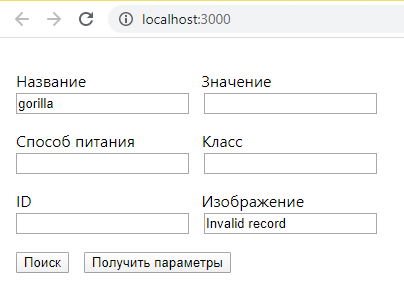


Рисунок 4 – Результат ввода неверных данных

**Вывод.**

В ходе выполнения лабораторной работы мы научились реализовывать технологи Frontend при помощи ReactJS.