



МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение высшего  
образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

ОТЧЕТ  
по лабораторной работе №3

Вариант 4  
по дисциплине  
«Web технологии»

РУКОВОДИТЕЛЬ:

\_\_\_\_\_  
(подпись)

\_\_\_\_\_  
Скорынин С.С.  
(фамилия, и.,о.)

СТУДЕНТ:

\_\_\_\_\_  
(подпись)

\_\_\_\_\_  
Сухоруков В.А.  
(фамилия, и.,о.)

\_\_\_\_\_  
19-В-2  
(шифр группы)

Работа защищена «\_\_» \_\_\_\_\_  
С оценкой \_\_\_\_\_

Нижний Новгород 2021

## Постановка задачи

Реализовать серверное и клиентское приложения для демонстрации запросов:

- ❖ Get
- ❖ Post
- ❖ Put
- ❖ Delete

## Используемые языки и технологии

- ❖ Язык разметки HTML-Данный язык необходим для описания иерархии объектов.
- ❖ Язык таблиц стилей CSS. - Данный язык необходим для описания внешнего вида страницы в целом и для объектов в отдельности.
- ❖ ASP.net Core MVC – многофункциональная платформа для создания веб-приложений и API-интерфейсов с помощью структуры проектирования MVC.
- ❖ React - JavaScript-библиотека для разработки пользовательских интерфейсов.
- ❖ Postgres SQL

## Описание работы

Серверное приложение основано на паттерне MVC. Концепция паттерна MVC предполагает разделение приложения на три компонента:

- ❖ Модель (model) – описывает используемые в приложении данные, а также логику, которая связана непосредственно с данными.
- ❖ Представление (view) – отвечает за визуальную часть или пользовательский интерфейс, через который пользователь взаимодействует с приложением.
- ❖ Контроллер (controller) – представляет центральный компонент MVC, который обеспечивает связь между пользователем и приложением, представлением и хранилищем данных.

## Модель

В папке Models модели Picture и PictureContext. Picture представляет модель картинки, которая содержит пять свойств: Id, Название, Количество лайков, Количество дизлайков, Оценка. Модель PictureContext нужна для взаимодействия с базой данных.

```

1 namespace MySite.Models
2 {
3     Ссылка: 18
4     public class Picture
5     {
6         Ссылка: 15
7         public int Id { get; set; }
8         Ссылка: 5
9         public string Name { get; set; }
10        Ссылка: 5
11        public int Likes { get; set; }
12        Ссылка: 5
13        public int Dislikes { get; set; }
14
15        Ссылка: 4
16        public int Grade { get; set; }
17    }
18 }

```

```

1 using Microsoft.EntityFrameworkCore;
2
3 namespace MySite.Models
4 {
5     public class PictureContext : DbContext
6     {
7         public DbSet<Picture> Pictures { get; set; }
8
9         public PictureContext(DbContextOptions<PictureContext> options)
10             : base(options)
11         {
12             Database.EnsureCreated();
13         }
14     }
15 }

```

## Представление

В папке Views находится папка Home, которая содержит одно пустое представление – Index.

## Контроллер

В папке Controllers содержится два контроллера: HomeController, который содержит метод для передачи данных из БД в представление и PictureController, который содержит функции для реакции на запросы.

```

5 namespace MySite.Controllers
6 {
7     ссылка: 1
8     public class HomeController : Controller
9     {
10         Ссылка: 0
11         PictureContext db;
12         public HomeController(PictureContext context)
13         {
14             db = context;
15         }
16         Ссылка: 0
17         public IActionResult Index()
18         {
19             return View(db.Pictures.ToList());
20         }
21     }
22 }

```

```

9 namespace MySite.Controllers
10 {
11     [ApiController]
12     [Route("[controller]")]
13     ссылка: 1
14     public class PictureController : Controller
15     {
16         Ссылка: 0
17         PictureContext db;
18         public PictureController(PictureContext context)
19         {
20             db = context;
21         }
22         Ссылка: 0
23         [HttpGet]
24         public IEnumerable<Picture> Get()
25         {
26             return db.Pictures.ToList();
27         }
28         [HttpPost]
29         Ссылка: 0
30         public void PostPicture(Picture product)
31         {
32             db.Pictures.Add(product);
33             db.SaveChanges();
34         }
35         [HttpPut]
36         Ссылка: 0
37         public void PutPicture(Picture product)
38         {
39             db.Pictures.Update(product);
40             db.SaveChanges();
41         }
42     }
43 }

```

Клиентское приложение содержит файлы Home.js, Add.js, Update.js, в которых содержатся методы postPicture, DeletePicture, PutPicture и разметка страниц.

```
const DeletePicture = (e) => {
  const id = e.target.elements.id.value;
  const requestOptions = {
    method: 'DELETE',
    headers: { 'Content-Type': 'application/json' },
  };
  fetch(`http://localhost:17499/Picture/${id}`, requestOptions);
}

return (
  <div>
    <h3>Рейтинг картинок</h3>
    <table className="table table-bordered">
      <thead>
        <tr className="table-dark">
          <td>Имя файла</td>
          <td>Количество лайков</td>
          <td>Количество дизлайков</td>
          <td>Оценка картинки</td>
          <td></td>
          <td></td>
        </tr>
      </thead>
      <tbody>
        {pictures.map((picture) => (
          <tr>
            <td>{picture.name}</td>
            <td>{picture.likes}</td>
            <td>{picture.dislikes}</td>
            <td>{picture.grade}</td>
            <td><Link className="btn btn-default" to={{ pathname: '/Update', data: picture }}>Оценить</Link>
            <td><form onSubmit={DeletePicture}>
              <input type="hidden" value={picture.id} name="id" />
              <button className="btn btn-default">Удалить</button>
            </form>
          </td>
        </tr>
        </tbody>
      </table>
      <Link className="btn btn-default" to={{ pathname: '/Add' }}>Добавить</Link>
    </div>
  );
```

```
const postPicture = () => {
  const requestOptions = {
    method: 'POST',
    headers: { 'Content-Type': 'application/json' },
    body: JSON.stringify({ "Name": Name, "Likes": parseInt(Likes), "Dislikes": parseInt(Dislikes), "Grade": parseInt(Grade) });
  };
  fetch("http://localhost:17499/Picture", requestOptions)
    .then(response => response.json());
}

return (
  <form>
    <p>Введите имя файла, который нужно добавить</p>
    <div className="form-group">
      <label htmlFor="Name" className="col-md-2 control-label">Название:</label>
      <div className="col-md-4">
        <input type="text" name="Name" className="form-control" onChange={(e) => setName(e.target.value)} />
      </div>
    </div>
    <div className="form-group">
      <div className="col-md-offset-2 col-md-10">
        <Link type="submit" onClick={postPicture} className="btn btn-default" to={{ pathname: '/' }}>Добавить</Link>
      </div>
    </div>
  </form>
);
```

```

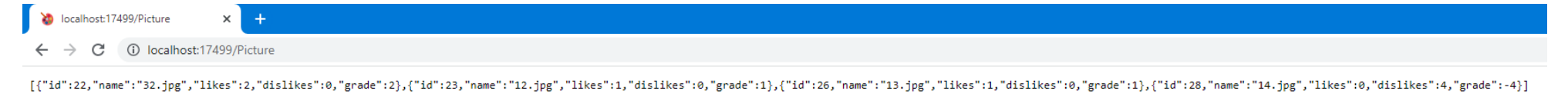
const PutPicture = () => {
  const requestOptions = {
    method: 'PUT',
    headers: { 'Content-Type': 'application/json' },
    body: JSON.stringify({ "Id": id, "Name": name, "Likes": parseInt(likes), "Dislikes": parseInt(dislikes), "Grade": grade });
  };
  fetch(`http://localhost:17499/Picture`, requestOptions)
    .then(response => response.json());
}

return (
  <form>
    <p>Для оценки увеличить значение на 1 в одном из полей: "Лайки" или "Дизлайки".</p>
    <p>Увеличьте или уменьшите значение поля "Оценка" на 1 в соответствии с вашим выбором.</p>
    <div className="form-group">
      <label htmlFor="Description" className="col-md-2 control-label">Лайки:</label>
      <div className="col-md-4">
        <input type="text" name="Likes" className="form-control" value={likes} onChange={(e) => setLikes(e.target.value)} />
      </div>
    </div>
    <div className="form-group">
      <label className="col-md-2 control-label">Дизлайки:</label>
      <div className="col-md-4">
        <input type="text" name="Dislikes" className="form-control" value={dislikes} onChange={(e) => setDislikes(e.target.value)} />
      </div>
    </div>
    <div className="form-group">
      <label className="col-md-2 control-label">Оценка:</label>
      <div className="col-md-4">
        <input type="text" name="Dislikes" className="form-control" value={grade} onChange={(e) => setGrade(e.target.value)} />
      </div>
    </div>
    <div className="form-group">
      <div className="col-md-offset-2 col-md-10">
        <Link type="submit" onClick={PutPicture} className="btn btn-default" to={{ pathname: '/' }}>Оценить</Link>
      </div>
    </div>
  </form>
);
}

```

Проблемы не найдены. Стр: 57 Симв: 16 Пробелы CRLF

# Результат работы



Project1

Главная

## Рейтинг картинок

Имя файла	Количество лайков	Количество дизлайков	Оценка картинки		
32.jpg	2	0	2	Оценить	Удалить
12.jpg	1	0	1	Оценить	Удалить
13.jpg	1	0	1	Оценить	Удалить
14.jpg	0	4	-4	Оценить	Удалить

Добавить

## Рейтинг картинок

Имя файла	Количество лайков	Количество дизлайков	Оценка картинки		
32.jpg	2	0	2	Оценить	Удалить
12.jpg	1	0	1	Оценить	Удалить
13.jpg	1	0	1	Оценить	Удалить
14.jpg	0	4	-4	Оценить	Удалить
25.jpg	0	0	0	Оценить	Удалить

Добавить

DevTools is now available in Russian! [Always match Chrome's language](#) [Switch DevTools to Russian](#) [Don't show again](#)

Elements Console Sources Network Performance Memory Application Security >> 2 6 1

Filter ☐ Invert ☐ Hide data URLs ☒ All Fetch/XHR JS CSS Img Media Font Doc WS Wasm Manifest Other

☐ Has blocked cookies ☐ Blocked Requests ☐ 3rd-party requests

20 ms 40 ms 60 ms 80 ms 100 ms 120 ms 140 ms 160 ms

Name X Headers Payload Preview Response Initiator Timing

☐ Picture

☒ Picture

☐ Picture

General

Request URL: http://localhost:17499/Picture

Request Method: POST

Status Code: 200 OK

Remote Address: [::1]:17499

Referrer Policy: strict-origin-when-cross-origin

Response Headers View source

Access-Control-Allow-Origin: http://localhost:39657

Date: Mon, 27 Dec 2021 12:23:14 GMT

Server: Microsoft-IIS/10.0

Transfer-Encoding: chunked

X-Powered-By: ASP.NET



Project1

localhost:39657/Update

Project1

Главная

Для оценки увеличить значение на 1 в одном из полей: "Лайки" или "Дизлайки".

Увеличьте или уменьшите значение поля "Оценка" на 1 в соответствии с вашим выбором.

Лайки:

3

Дизлайки:

0

Оценка:

3

Оценить

DevTools is now available in Russian! Always match Chrome's language Switch DevTools to Russian Don't show again

Elements Console Sources Network Performance Memory Application Security

Filter

Has blocked cookies Blocked Requests 3rd-party requests

10 ms 20 ms 30 ms 40 ms 50 ms 60 ms 70 ms 80 ms 90 ms 100 ms 110

Recording network activity...  
Perform a request or hit **Ctrl + R** to record the reload.  
[Learn more](#)

Project1

localhost:39657

Project1

Главная

### Рейтинг картинок

Имя файла	Количество лайков	Количество дизлайков	Оценка картинки		
12.jpg	1	0	1	Оценить	Удалить
13.jpg	1	0	1	Оценить	Удалить
25.jpg	0	0	0	Оценить	Удалить
32.jpg	3	0	3	Оценить	Удалить

Добавить

DevTools is now available in Russian! Always match Chrome's language Switch DevTools to Russian Don't show again

Elements Console Sources Network Performance Memory Application Security

Filter

Has blocked cookies Blocked Requests 3rd-party requests

10 ms 20 ms 30 ms 40 ms 50 ms 60 ms 70 ms 80 ms 90 ms 100 ms 110

Name

Picture

Picture

Picture

General

Request URL: http://localhost:17499/Picture  
Request Method: PUT  
Status Code: 200 OK  
Remote Address: [::1]:17499  
Referrer Policy: strict-origin-when-cross-origin

Response Headers

Access-Control-Allow-Origin: http://localhost:39657  
Date: Mon, 27 Dec 2021 12:25:11 GMT  
Server: Microsoft-IIS/10.0  
Transfer-Encoding: chunked  
X-Powered-By: ASP.NET

## **Вывод**

В ходе выполнения лабораторной работы были изучены и освоены основы проектирования веб-сервиса на основе REST.