

## Ejercicio 2(tarea 2)

### - Implementar el Servicio de CalificacionesDto

```
1 package mx.unam.dgtic.alumnos_rest.servicios.v2.interfaces;
2
3 import mx.unam.dgtic.alumnos_rest.dtos.CalificacionDto;
4
5 import java.util.List;
6
7 public interface CalificacionDtoService { 6 usages 1 implementation
8     List<CalificacionDto> findAll(); 1 usage 1 implementation
9     CalificacionDto findById(Long id); 1 implementation
10    CalificacionDto save(CalificacionDto calificacionDto); 1 usage 1 implementation
11    CalificacionDto update(Long id, CalificacionDto calificacionDto); 1 usage 1 implementation
12    CalificacionDto delete(Long id); 1 usage 1 implementation
13    List<CalificacionDto> getPagina(int page, int size, String direction, String sort);
14 }
```

### - Rest Controller completo para Calificaciones Dto

```

4  import mx.unam.dgtic.alumnos_rest.servicios.v2.interfaces.CalificacionDtoService;
5  import org.springframework.beans.factory.annotation.Autowired;
6  import org.springframework.http.ResponseEntity;
7  import org.springframework.web.bind.annotation.*;
8
9  import java.util.List;
10
11  @RestController  no usages
12  @RequestMapping("/api/v2/calificaciones")
13  public class CalificacionDtoRestController {
14
15      @Autowired  5 usages
16      private CalificacionDtoService service;
17
18      @GetMapping("/")  no usages
19      public ResponseEntity<List<CalificacionDto>> getAll() {
20          return ResponseEntity.ok(service.findAll());
21      }
22
23      @GetMapping("/{id}")  no usages
24      public ResponseEntity<CalificacionDto> getById(@PathVariable Long id) {
25          return ResponseEntity.ok(service.findById(id));
26      }
27
28      @PostMapping("/")  no usages
29      public ResponseEntity<CalificacionDto> save(@RequestBody CalificacionDto dto) {
30          return ResponseEntity.ok(service.save(dto));
31      }
32
33      @PutMapping("/{id}")  no usages
34      public ResponseEntity<CalificacionDto> update(@PathVariable Long id, @RequestBody CalificacionDto dto) {
35          return ResponseEntity.ok(service.update(id, dto));
36      }
37
38      @DeleteMapping("/{id}")  no usages
39      public ResponseEntity<CalificacionDto> delete(@PathVariable Long id) {
40          return ResponseEntity.ok(service.delete(id));
41      }
42  }

```

- Implementar paginado para CalificacionesDto.

```

1 package mx.unam.dgtic.alumnos_rest.controladores.v2;
2
3 import mx.unam.dgtic.alumnos_rest.dtos.CalificacionDto;
4 import mx.unam.dgtic.alumnos_rest.servicios.v2.interfaces.CalificacionDtoService;
5 import org.springframework.beans.factory.annotation.Autowired;
6 import org.springframework.http.ResponseEntity;
7 import org.springframework.web.bind.annotation.*;
8
9 import java.util.List;
10
11 @RestController no usages
12 @RequestMapping("/api/v2/paginable/calificaciones")
13 public class CalificacionDtoRestControllerPageable {
14
15     @Autowired 1 usage
16     private CalificacionDtoService service;
17
18     @GetMapping("") no usages
19     public ResponseEntity<List<CalificacionDto>> getPagina(
20         @RequestParam(defaultValue = "1") int page,
21         @RequestParam(defaultValue = "5") int size,
22         @RequestParam(defaultValue = "asc") String direction,
23         @RequestParam(defaultValue = "id") String sort
24     ) {
25         return ResponseEntity.ok(service.getPagina(page, size, direction, sort));
26     }
27 }
28

```

## Postman:

CalificacionRestController

## Metodo GET:


GET ▼ http://localhost:8080/api/v1/cals/

Params Authorization Headers (8) **Body ●** Pre-requests

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw

```
1  {}
2
3  {
4    "nombre": "Carlos",
5    "apellido": "Hernández",
6    "email": "carlos.hdez@unam.mx"
7  }
```

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON ▼ 

```
1  [
2    {
3      "curso": "Programación",
4      "nota": 10.0,
5      "id": 3
6    },
7    {
8      "curso": "Inglés",
9      "nota": 7.8,
10     "id": 4
11   },
12   {
13     "curso": "Biología",
14     "nota": 8.3,
15     "id": 5
16   }
17 ]
```

**Metodo GET{id}:**

GET

▼

http://localhost:8080/api/v1/cals/3

ParamsAuthorizationHeaders (8)Body ●Pre-request Script

● none

● form-data

● x-www-form-urlencoded

● raw

● bin

1

2

3

4

5

6

1

2

3

4

5

6

1

2

3

4

5

6

1

2

3

4

5

6

BodyCookiesHeaders (5)Test Results

Pretty

Raw

Preview

Visualize

JSON ▼

≡

1

2

3

4

5

1

2

3

4

5

**Metodo POST:**

HTTP <http://localhost:8080/api/v1/cals/>

**POST** <http://localhost:8080/api/v1/cals/>

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary **JSON** ☐

```
1 {
2   "curso": "Arte",
3   "nota": 8.3
4 }
```

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON ☐

```
1 {
2   "curso": "Arte",
3   "nota": 8.3,
4   "id": 10
5 }
```

**Metodo PUT:**

HTTP **http://localhost:8080/api/v1/cals/3**

**PUT** **http://localhost:8080/api/v1/cals/3**

Params Authorization Headers (8) **Body** Pre-req

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw

```
1 {
2   .....
3   ..... "curso": "NUEVO CURSO",
4   ..... "nota": 0.0
5   .....
6 }
```

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON **▼**

```
1 {
2   "curso": "NUEVO CURSO",
3   "nota": 0.0,
4   "id": 3
5 }
```

**Metodo DELETE:**


HTTP **DELETE** ⌵ http://localhost:8080/api/v1/cals/3

Params Authorization Headers (8) **Body** ● Pre-request Script Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary **JSON** ⌵

```
1 {  
2   .....  
3   ..... "curso": "NUEVO CURSO",  
4   ..... "nota": 0.0  
5   .....  
6   ..... }
```

**Body** Cookies Headers (5) Test Results

Pretty Raw Preview Visualize **JSON** ⌵ 

```
1 {  
2   "curso": "NUEVO CURSO",  
3   "nota": 0.0,  
4   "id": 3  
5 }
```

**PAGINACION:**





http://localhost:8080/api/v2/paginable/calificaciones

GET

http://localhost:8080/api/v2/paginable/calificaciones

Params

Authorization

Headers (8)

Body ●

Pre-request Script

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

JSON



```
1 [
2   {
3     "id": 1,
4     "curso": "Matemáticas",
5     "nota": 9.2,
6     "alumnoId": 1
7   },
8   {
9     "id": 2,
10    "curso": "Historia",
11    "nota": 8.5,
12    "alumnoId": 1
13  },
14  {
15    "id": 3,
16    "curso": "Biología",
17    "nota": 7.8,
18    "alumnoId": 2
19  },
20  {
21    "id": 4,
22    "curso": "Física",
23    "nota": 9.0,
24    "alumnoId": 2
25  },
26  {
27    "id": 5,
28    "curso": "Química",
29    "nota": 6.7,
30    "alumnoId": 3
31  }
32 ]
```

## METODO GET:

GET

http://localhost:8080/api/v2/calificaciones/

Params

Authorization

Headers (8)

Body ●

Pre-request Script

Tests

Settings

## Query Params

Key

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

JSON



```
1  [
2    {
3      "id": 1,
4      "curso": "Matemáticas",
5      "nota": 9.2,
6      "alumnoId": 1
7    },
8    {
9      "id": 2,
10     "curso": "Historia",
11     "nota": 8.5,
12     "alumnoId": 1
13   },
14   {
15     "id": 3,
16     "curso": "Biología",
17     "nota": 7.8,
18     "alumnoId": 2
19   },
20   {
21     "id": 4,
22     "curso": "Física",
23     "nota": 9.0,
24     "alumnoId": 2
25   },
26   {
27     "id": 5,
28     "curso": "Química",
29     "nota": 6.7,
30     "alumnoId": 3
31   }
32 ]
```

## METODO GET{id}:

HTTP <http://localhost:8080/api/v1/cals/>

**GET** <http://localhost:8080/api/v2/calificaciones/2>

Params Authorization Headers (8) Body ● Pre-request Script

Query Params

| Key |
|-----|
|-----|

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

```
1 {  
2   "id": 2,  
3   "curso": "Historia",  
4   "nota": 8.5,  
5   "alumnoId": 1  
6 }
```

## METODO POST:

http://localhost:8080/api/v1/cais/

**POST** http://localhost:8080/api/v2/calificaciones/

Params Authorization Headers (8) **Body** Pre-request Script Test

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary


```
1 {
2   "curso": "Arte",
3   "nota": 8.3,
4   "alumnoId": 5
5 }
```

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 10,
3   "curso": "Arte",
4   "nota": 8.3,
5   "alumnoId": 5
6 }
```

**METODO PUT:**

 http://localhost:8080/api/v1/cals/

PUT

http://localhost:8080/api/v2/calificaciones/10

Params

Authorization

Headers (8)

Body

Pre-request Script

Test Results

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary [JSON](#)

```
1 {
2   "curso": "DISEÑO",
3   "nota": 0.0,
4   "alumnoId": 5
5 }
```

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview


Visualize

JSON



```
1 {
2   "id": 10,
3   "curso": "DISEÑO",
4   "nota": 0.0,
5   "alumnoId": 5
6 }
```

**METODO DELETE:**

 <http://localhost:8080/api/v1/cals/>

**DELETE** ▾

<http://localhost:8080/api/v2/calificaciones/10>

Params Authorization Headers (8) Body ● Pre-request Script

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary

```
1  {}
2  {
3    "curso": "DISEÑO",
4    "nota": 0.0,
5    "alumnoId": 5
6  }
```

Body Cookies Headers (5) Test Results

Pretty

Raw

Preview

Visualize

JSON ▾



```
1  {}
2  {
3    "id": 10,
4    "curso": "DISEÑO",
5    "nota": 0.0,
6    "alumnoId": 5
7  }
```

 <http://localhost:8080/api/v1/cals/>

GET



<http://localhost:8080/api/v2/calificaciones/>

Params

Authorization

Headers (8)

Body ●

Pre-request Script

Tests

Se

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary **JSON** ▼

```
1  {}
2  .... "curso": "DISEÑO",
3  .... "nota": 0.0,
4  .... "alumnoId": 5
5  {}
```

Body Cookies Headers (5) Test Results

Pretty

Raw

Preview

Visualize

JSON ▼



```
30  "alumnoId": 3
31  },
32  {
33    "id": 6,
34    "curso": "Literatura",
35    "nota": 8.0,
36    "alumnoId": 4
37  },
38  {
39    "id": 7,
40    "curso": "Matemáticas",
41    "nota": 9.5,
42    "alumnoId": 5
43  },
44  {
45    "id": 8,
46    "curso": "Inglés",
47    "nota": 7.9,
48    "alumnoId": 3
49  },
50  {
51    "id": 9,
52    "curso": "Computación",
53    "nota": 10.0,
54    "alumnoId": 4
55  }
56  }
```

