

LABORATORIO 7 - GESTIÓN DE SERVICIOS TICS

Estudiante: JEFFERSON JAVIER VILLEGAS ALVARADO

Código: 20203266

Curso: Gestión de Servicios TICS

Sección: H791

Tema: Documentación y pruebas de API REST con Spring Boot

ENDPOINT 1: Registrar estudiante

Método: POST

Ruta: /estudiantes

Descripción: Registra un nuevo estudiante en la base de datos.

```
Selecionar Administrador: Símbolo del sistema
Microsoft Windows [Versión 10.0.22631.3535]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Windows\System32>curl -X POST http://localhost:8080/estudiantes ^
-H "Content-type: application/json" ^
-d '{"nombres": "Ana", "apellidos": "Ramirez", "dni": "12345678", "codigoPucp": "20203266", "fechaNacimiento": "2000-05-01", "sexo": "F", "correoInstitucional": "aramirez@pucp.edu.pe", "correoPersonal": "ana@gmail.com", "telefono": "987654321", "direccion": "Av. Primavera 100", "departamento": "Lima", "provincia": "Lima", "carrera": "Ingeniería Informática"}'

Estudiante registrado correctamente.
C:\Windows\System32>
```

ENDPOINT 2: Listar todos los estudiantes

Método: GET

Ruta: /estudiantes

Descripción: Devuelve todos los estudiantes registrados (incluye campos como estado, id, carrera, etc.)

```
localhost:8080/estudiantes

[[{"id":1,"nombres":"Ana","apellidos":"Ramirez","dni":"12345678","codigoPucp":"20203266","correoInstitucional":"aramirez@pucp.edu.pe","correoPersonal":"ana@gmail.com","telefono":"987654321","direccion":"Av. Primavera 100","departamento":"Lima","provincia":"Lima","carrera":"Ingeniería Informática","sexo":"F","estado":"Activo"}]]
```

ENDPOINT 3: Buscar estudiante por ID

Método: GET

Ruta: /estudiantes/id/{id}

Descripción: Devuelve los datos de un estudiante por su ID

```
localhost:8080/estudiantes/id/1

{"id":1,"nombres":"Ana","apellidos":"Ramirez","dni":"12345678","codigoPucp":"20203266","correoInstitucional":"aramirez@pucp.edu.pe","correoPersonal":"ana@gmail.com","telefono":"987654321","direccion":"Av. Primavera 100","departamento":"Lima","provincia":"Lima","carrera":"Ingeniería Informática","sexo":"F","estado":"Activo"}
```

ENDPOINT 4: Eliminar lógicamente estudiante

Método: DELETE

Ruta: /estudiantes/{id}

Descripción: Cambia el estado del estudiante a "Inactivo"

```
C:\Windows\System32>
C:\Windows\System32>curl -X DELETE http://localhost:8080/estudiantes/1
Estudiante dado de baja (borrado lógico).
C:\Windows\System32>
```



PRUEBAS DE MANEJO DE ERRORES

DNI INVÁLIDO (menos de 8 dígitos)

Método: POST /estudiantes

Error esperado: "dni": "El DNI debe tener exactamente 8 dígitos"

```
C:\Windows\System32>curl -X POST http://localhost:8080/estudiantes -H 'Content-Type: application/json' -d '{"id":1,"nombres":"Carlos","apellidos":"Ramos","dni":"1234","codigoPucp":"202099999","fechaNacimiento":"2001-01-01","sexo":"M","correoInstitucional":"cramos@pucp.edu.pe","correoPersonal":"cramos@gmail.com","telefono":"987654321","direccion":"Av. Siempre Viva","departamento":"Lima","provincia":"Lima","carrera":"Electrónica"}'
{"dni":"El DNI debe tener exactamente 8 dígitos"}
C:\Windows\System32>
```

CORREO INSTITUCIONAL INVÁLIDO (sin dominio @pucp.edu.pe)

Método: POST /estudiantes

Error esperado: "correoInstitucional": "Debe ser dominio @pucp.edu.pe"

```
C:\Windows\System32>
C:\Windows\System32>curl -X POST http://localhost:8080/estudiantes -H 'Content-Type: application/json' -d '{"id":1,"nombres":"Carlos","apellidos":"Ramos","dni":"987654321","codigoPucp":"202099999","fechaNacimiento":"2001-01-01","sexo":"M","correoInstitucional":"cramos@gmail.com","correoPersonal":"cramos@gmail.com","telefono":"987654321","direccion":"Av. Siempre Viva","departamento":"Lima","provincia":"Lima","carrera":"Electrónica"}'
{"correoInstitucional":"Debe ser dominio @pucp.edu.pe"}
C:\Windows\System32>
```

ID NO EXISTE (GET)

Método: GET /estudiantes/id/99

Respuesta esperada: Estudiante no encontrado

```
C:\Windows\System32>
C:\Windows\System32>curl http://localhost:8080/estudiantes/id/99
Estudiante no encontrado
C:\Windows\System32>
```

ID NO EXISTE (DELETE)

Método: DELETE /estudiantes/99

Respuesta esperada: Estudiante no encontrado

```
C:\Windows\System32>curl -X DELETE http://localhost:8080/estudiantes/99
Estudiante no encontrado
C:\Windows\System32>_
```

La API REST desarrollada en Spring Boot funciona correctamente para registrar, listar, buscar, actualizar (borrado lógico) y validar errores al ingresar datos inválidos. Las pruebas con curl en Windows han sido exitosas y se documentan con sus respectivas capturas.