

Prueba Redarbor que cumpla los siguientes puntos:

1. Construir una WebAPI en Python
2. Seguir estructura MVC
3. Utilizar ElasticSearch para guardar los datos
4. Evitar utilizar frameworks (Django...etc)

Estas son las llamadas que puedes hacer a la API:

API	Description	Request body	Response body
GET /api/redarbor	Get all employees items	None	Array of employee items
GET /api/redarbor/{id}	Get an item by ID	None	Employee item
POST /api/redarbor	Add a new item	Employee item	Employee item
PUT /api/redarbor/{id}	Update an existing item	Employee item	None
DELETE /api/redarbor/{id}	Delete an item	None	None

1. Inserta un nuevo Employee con este json:

```
{
  "CompanyId": 1,
  "CreatedOn": "2000-01-01 00:00:00",
  "DeletedOn": "2000-01-01 00:00:00",
  "Email": "test1@test.test.tmp",
  "Fax": "000.000.000",
  "Name": "test1",
  "Lastlogin": "2000-01-01 00:00:00",
```

```
"Password": "test",
"PortalId": 1,
"RoleId": 1,
"StatusId": 1,
"Telephone": "000.000.000",
"UpdatedOn": "2000-01-01 00:00:00",
"Username": "test1"
}
```

Los campos obligatorios (no pueden ser nulls) son: CompanyId, Email, Password, PortalId, RoleId, StatusId, Username.

Ejemplo en curl:

```
$ curl -i -XPOST -H "Content-Type: application/json" -d @example.json
http://localhost:<PORT>/api/redarbor/
```

NOTA: Puedes utilizar Postman o curl para realizar las llamadas.

2. Ejecuta la llamada GET para ver que está insertado correctamente:

```
$ curl -s http://localhost:<PORT>/api/redarbor/ | jq .
```

NOTA: jq es una utilidad para tratar json strings. Puedes ejecutarlo con Postman.

3. Ejecuta la llamada GET para ese id.

```
$ curl -s http://localhost:<PORT>/api/redarbor/1 | jq .
```

NOTA: jq es una utilidad para tratar json strings. Puedes ejecutarlo con Postman.

4. Modifica el registro insertado para que su usuario sea “test1updated”

```
curl -s -i -XPUT -H "Content-Type: application/json" -d @example.json
http://localhost:<PORT>/api/redarbor/1
```

NOTA: Puedes ejecutarlo con Postman.

5. Verificar que se ha modificado.

```
$ curl -s http://localhost:<PORT>/api/redarbor/ | jq .
```

NOTA: jq es una utilidad para tratar json strings. Puedes ejecutarlo con Postman.

6. Borrar el nuevo registro.

```
$ curl -s -XDELETE http://localhost:<PORT>/api/redarbor/1
```

NOTA: Puedes ejecutarlo con Postman.

7. Comprobar que no existe.

```
$ curl -s http://localhost:<PORT>/api/redarbor/ | jq .
```

NOTA: jq es una utilidad para tratar json strings. Puedes ejecutarlo con Postman.