

Examen practico arquitectura de software – Prueba de funcionamiento

1. Post de un almacen

The screenshot displays a REST client interface with two tabs. The top tab shows the request configuration for a POST to `http://localhost:8015/api/v1/warehouses` with a JSON body. The bottom tab shows the response, which is a 201 Created status with a JSON body containing the created warehouse details and links.

Request Configuration:

- Method: POST
- URL: `http://localhost:8015/api/v1/warehouses`
- Body Type: raw (JSON)

Request Body (JSON):

```
1 {
2   "name": "Almacén Central",
3   "location": "Ciudad X",
4   "products": [
5     {
6       "name": "Producto A",
7       "quantity": 1000
8     },
9     {
10      "name": "Producto B",
11      "quantity": 500
12    }
13  ]
14 }
```

Response Configuration:

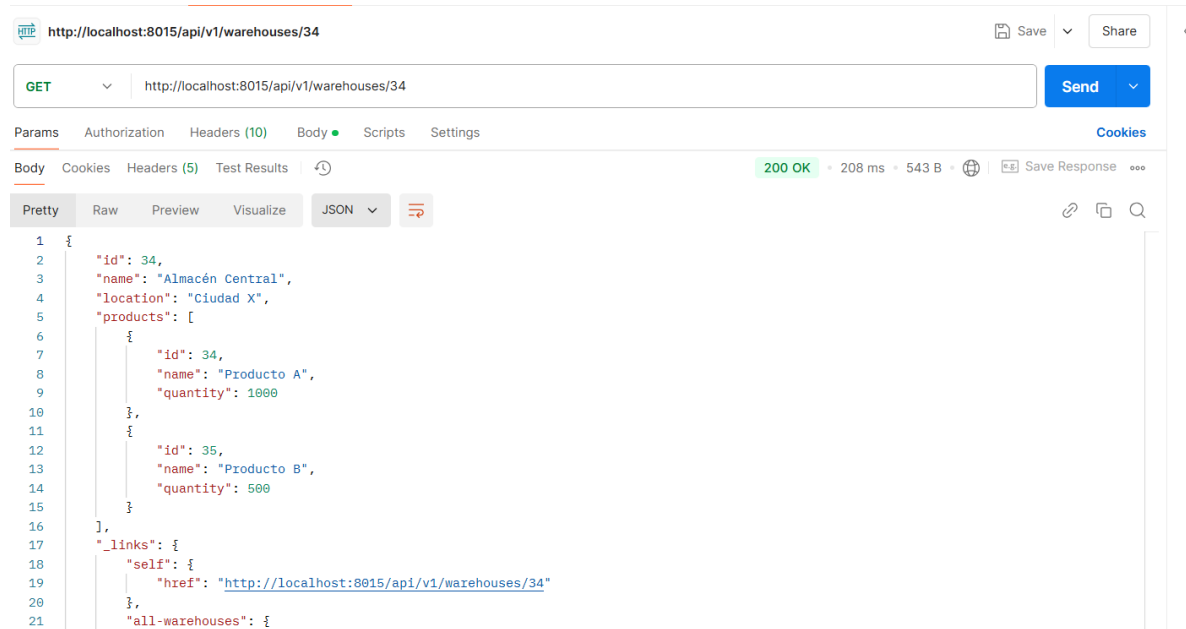
- Method: POST
- URL: `http://localhost:8015/api/v1/warehouses`
- Headers (10): Content-Type: application/json

Response Body (JSON):

```
1 {
2   "id": 34,
3   "name": "Almacén Central",
4   "location": "Ciudad X",
5   "products": [
6     {
7       "id": 34,
8       "name": "Producto A",
9       "quantity": 1000
10    },
11    {
12      "id": 35,
13      "name": "Producto B",
14      "quantity": 500
15    }
16  ],
17   "_links": {
18     "self": {
19       "href": "http://localhost:8015/api/v1/warehouses/34"
20     },
21     "all-warehouses": {
22       "href": "http://localhost:8015/api/v1/warehouses"
23     }
24   }
25 }
```

Response Status: 201 Created (322 ms, 528 B)

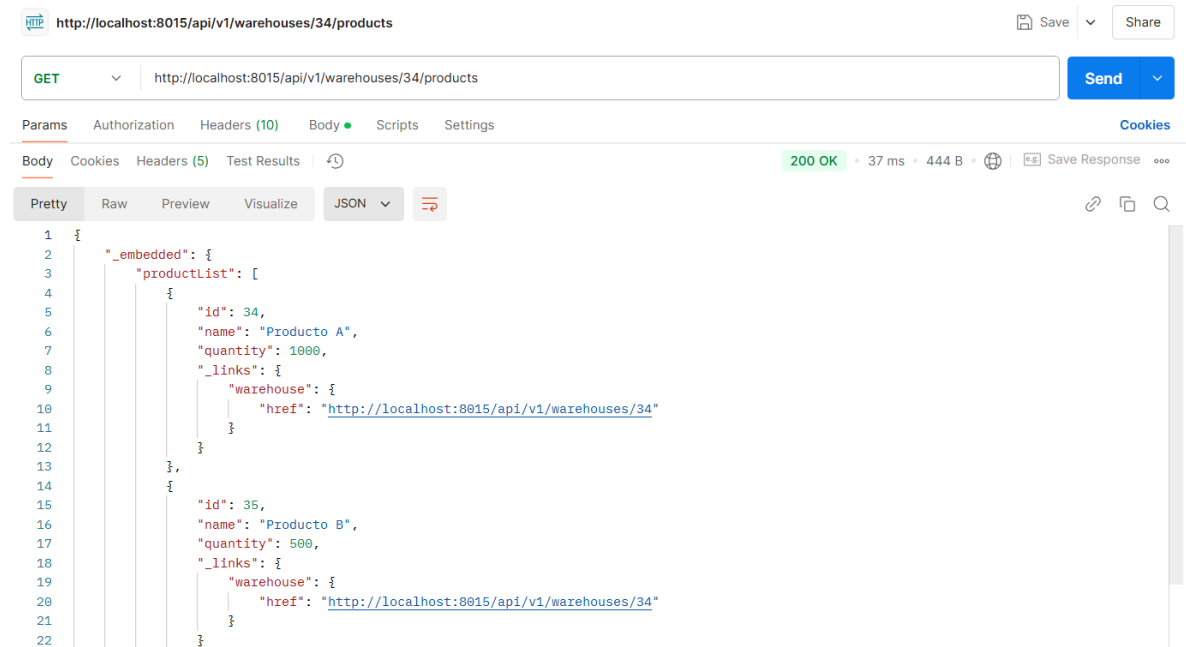
2. Get el almacen por ID



REST client interface showing a GET request to `http://localhost:8015/api/v1/warehouses/34`. The response is a 200 OK status with a JSON body. The JSON body contains the following data:

```
1 {
2   "id": 34,
3   "name": "Almacén Central",
4   "location": "Ciudad X",
5   "products": [
6     {
7       "id": 34,
8       "name": "Producto A",
9       "quantity": 1000
10    },
11    {
12      "id": 35,
13      "name": "Producto B",
14      "quantity": 500
15    }
16  ],
17  "_links": {
18    "self": {
19      "href": "http://localhost:8015/api/v1/warehouses/34"
20    },
21    "all-warehouses": {
```


3. Obtener los productos del almacen.






REST client interface showing a GET request to `http://localhost:8015/api/v1/warehouses/34/products`. The response is a 200 OK status with a JSON body. The JSON body contains the following data:



```
1 {
2   "_embedded": {
3     "productList": [
4       {
5         "id": 34,
6         "name": "Producto A",
7         "quantity": 1000,
8         "_links": {
9           "warehouse": {
10             "href": "http://localhost:8015/api/v1/warehouses/34"
11           }
12         }
13       },
14       {
15         "id": 35,
16         "name": "Producto B",
17         "quantity": 500,
18         "_links": {
19           "warehouse": {
20             "href": "http://localhost:8015/api/v1/warehouses/34"
21           }
22         }
23       }
24     ]
25   }
26 }
```



4. Prueba de dockerización del servicio y la base de datos





**almacen_2**
C:\Users\USER\Downloads\almacen_2

[View Configurations](#)



**postgres**
postgres:15
5432:5432 

**almacen_2-app-1**
almacen_2-app:<none>
8015:8015 



```
2024-11-29 19:22:47 app-1 | 2024-11-30T00:22:47.674Z DEBUG 1 --- [nio-8015-exec-4] o.s.w.s.r.ResourceHttpRequestHandler : Resource not found
2024-11-29 19:22:47 app-1 | 2024-11-30T00:22:47.678Z DEBUG 1 --- [nio-8015-exec-4] .w.s.m.s.DefaultHandlerExceptionResolver : Resolved [org.springframework.k.web.servlet.resource.NoResourceFoundException: No static resource .]
2024-11-29 19:22:47 app-1 | 2024-11-30T00:22:47.680Z DEBUG 1 --- [nio-8015-exec-4] o.s.web.servlet.DispatcherServlet : Completed 404 NOT_FOUND
2024-11-29 19:22:47 app-1 | 2024-11-30T00:22:47.686Z DEBUG 1 --- [nio-8015-exec-4] o.s.web.servlet.DispatcherServlet : "ERROR" dispatch for GET "/error", parameters={}
2024-11-29 19:22:47 app-1 | 2024-11-30T00:22:47.690Z DEBUG 1 --- [nio-8015-exec-4] s.w.s.m.a.RequestMappingHandlerMapping : Mapped to org.springframework.k.boot.autoconfigure.web.servlet.error.BasicErrorController#error(HttpServletRequest)
2024-11-29 19:22:47 app-1 | 2024-11-30T00:22:47.696Z DEBUG 1 --- [nio-8015-exec-4] o.s.w.s.m.a.HttpEntityMethodProcessor : Using 'application/json', given [*/*] and supported [application/json, application/*+json, application/yaml]
2024-11-29 19:22:47 app-1 | 2024-11-30T00:22:47.699Z DEBUG 1 --- [nio-8015-exec-4] o.s.w.s.m.a.HttpEntityMethodProcessor : Writing [{timestamp=Sat Nov 30 00:22:47 UTC 2024, status=404, error=Not Found, path=/}]
2024-11-29 19:22:47 app-1 | 2024-11-30T00:22:47.703Z DEBUG 1 --- [nio-8015-exec-4] o.s.web.servlet.DispatcherServlet : Exiting from "ERROR" dispatch, status 404
```

5. OpenApi documentation

OpenAPI definition v0 QAS3
/v3/api-docs

Servers

http://localhost:8015 - Generated server url

warehouse-controller

GET /api/v1/warehouses

POST /api/v1/warehouses

GET /api/v1/warehouses/{id}

GET /api/v1/warehouses/{id}/products