

Reto Docker

Natalia Flórez Guzmán

Gestión de la Configuración

José Ignacio López Vélez

Universidad de Medellín

2025

Segunda parte

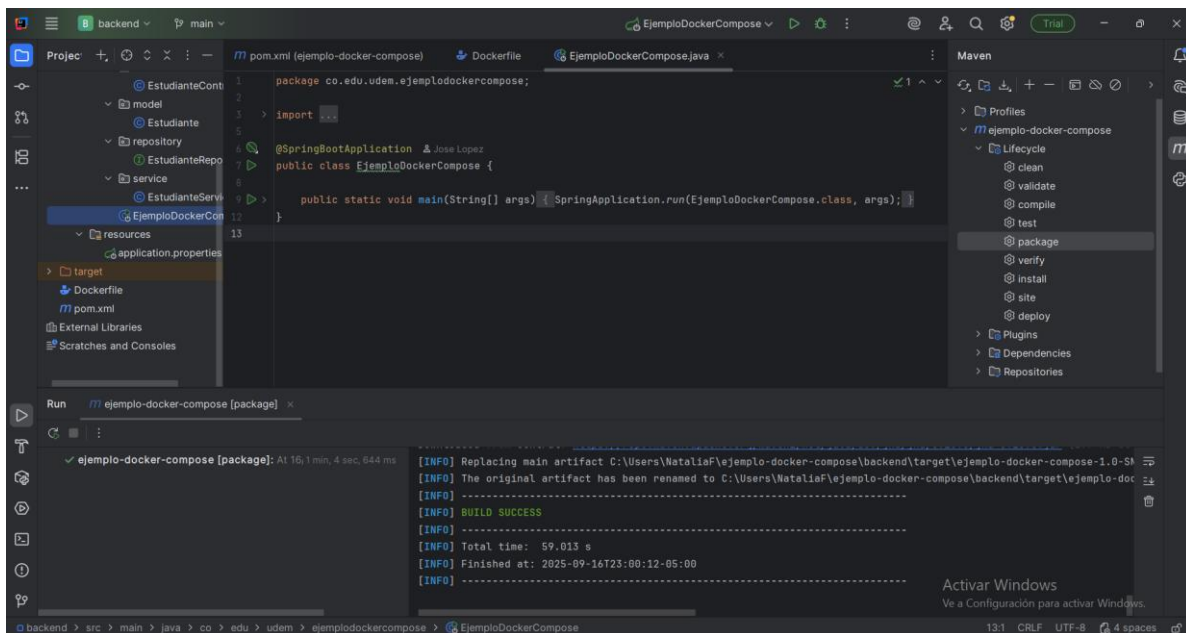
Paso 1. Clonar el repo

```
Microsoft Windows [Versión 10.0.26100.6584]
(c) Microsoft Corporation. Todos los derechos reservados.

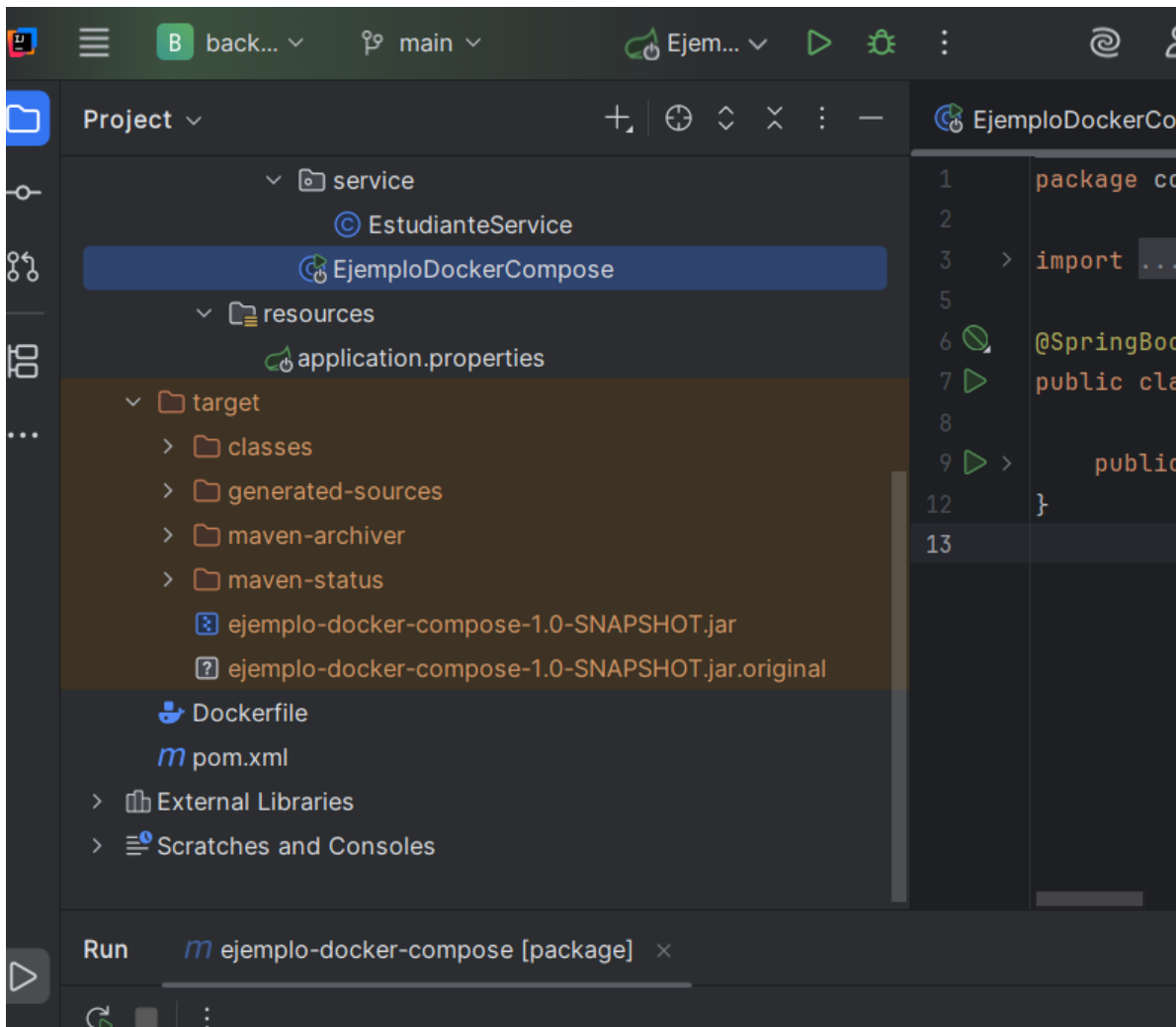
C:\Users\NataliaF>git clone https://github.com/jilopezv-udem/ejemplo-docker-compose.git
Cloning into 'ejemplo-docker-compose'...
remote: Enumerating objects: 95, done.
remote: Counting objects: 100% (95/95), done.
remote: Compressing objects: 100% (62/62), done.
remote: Total 95 (delta 17), reused 90 (delta 15), pack-reused 0 (from 0)
Receiving objects: 100% (95/95), 127.99 KiB | 627.00 KiB/s, done.
Resolving deltas: 100% (17/17), done.

C:\Users\NataliaF>|
```

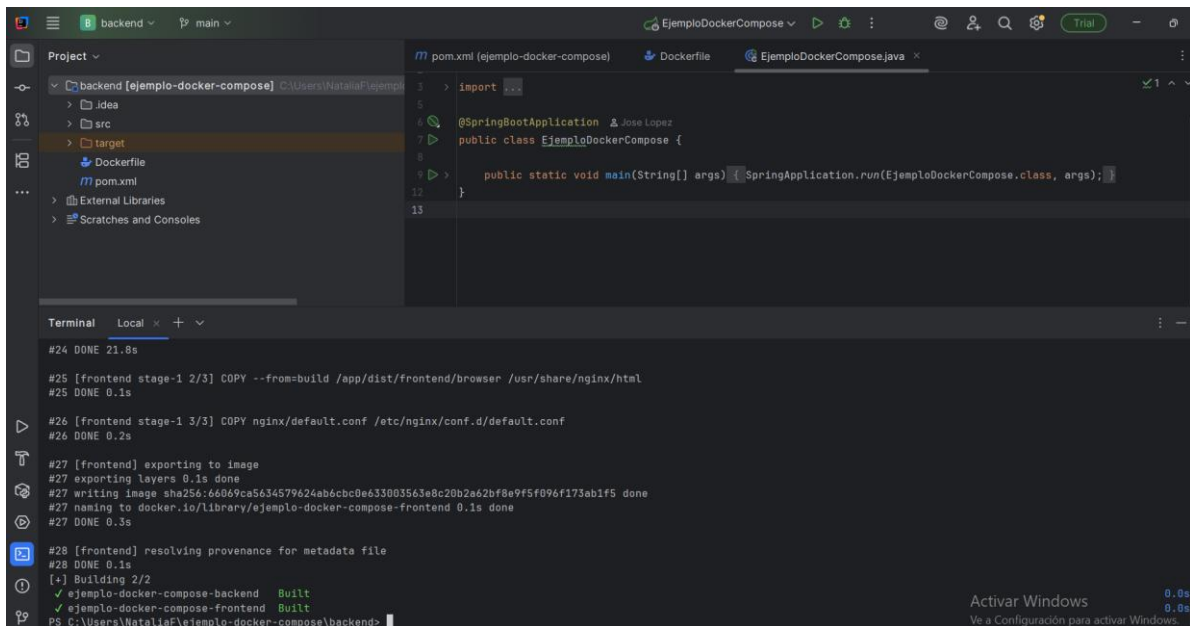
Paso 2. Se compila con Maven-Lifecycle-package



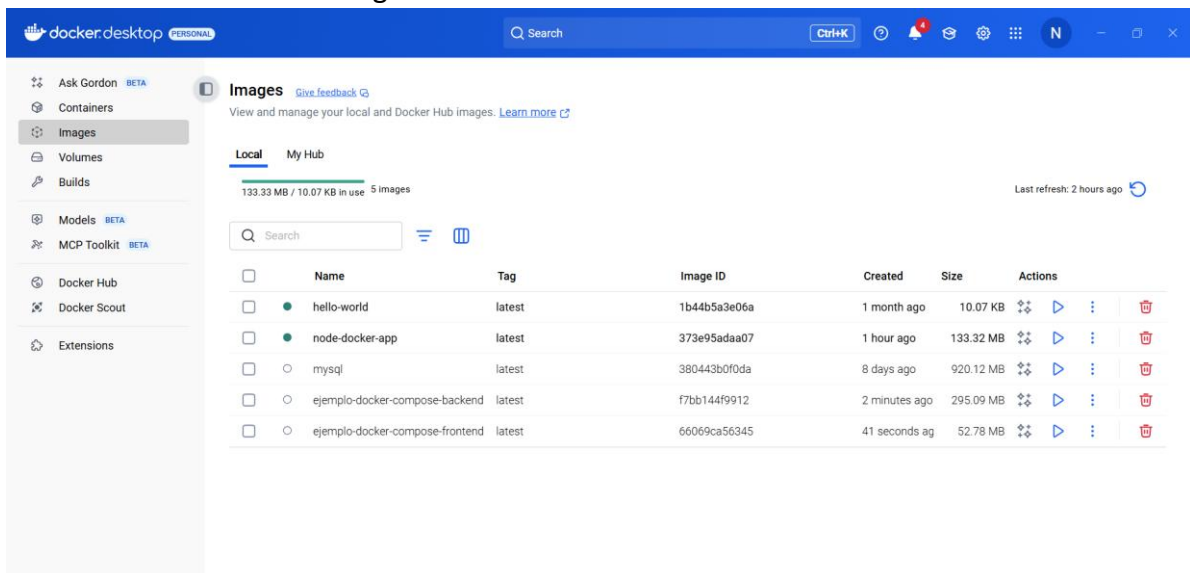
Al hacerlo se creó el .jar



Paso 3. Crear la imagen con docker-compose build



Miramos en Docker si la imagen se creó



Paso 4. Para levantar los servicios ejecutamos en la raíz docker-compose up -d
Y los listamos con Docker ps

The screenshot shows an IDE with two main panels. The top panel displays a Docker Compose file named `docker-compose.yml` with the following content:

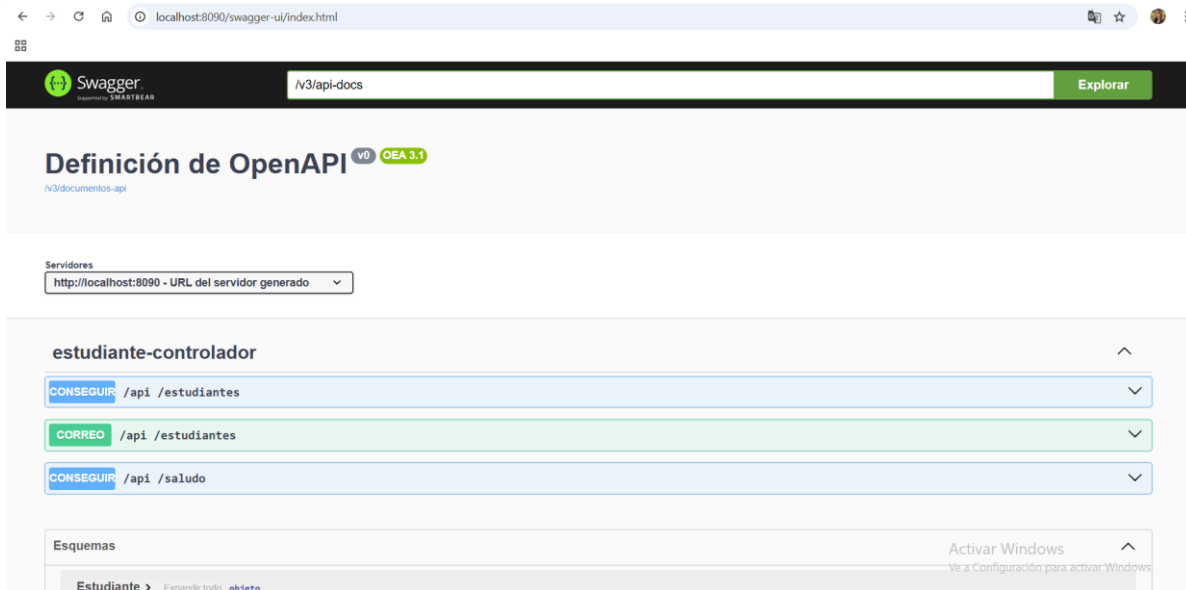
```
services:
  db_mysql:
    image: mysql:latest
    MYSQL_PASSWORD: usuario123
    ports:
      - "3308:3306"
  backend_app:
    image: ejemplo-docker-compose-backend
    ports:
      - "8090:8080"
  frontend_app:
    image: ejemplo-docker-compose-frontend
    ports:
      - "80:80"
```

The bottom panel shows a PowerShell terminal window with the following output:

```
C:\Users\NataliaF\ejemplo-docker-compose> docker-compose up -d
Running 3/3
Container frontend_app Running 0.0s
Container db_mysql Running 0.0s
Container backend_app Started 0.6s
C:\Users\NataliaF\ejemplo-docker-compose> docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
45332b17	mysql:latest	"docker-entrypoint.s..."	3 minutes ago	Up
ff95230b	ejemplo-docker-compose-backend	"/_cacert_entrypoin..."	15 minutes ago	Up
a02a2aa1	ejemplo-docker-compose-frontend	"/docker-entrypoint..."	15 minutes ago	Up
b14694ef	373e95adaa07	"docker-entrypoint.s..."	2 hours ago	Up

Comprobamos que haya arrancado bien la app y esté conectada a MySql



Paso 5. Crear y agregar el método DELETE en EstudianteController

```
@DeleteMapping("/estudiantes/{id}")
public void deleteEstudiante(@PathVariable Long id) {
    estudianteService.delete(id);
}
```

Y se agrega el método en EstudianteService

```
public void delete(Long id) {
    if (!estudianteRepository.existsById(id)) {
        throw new RuntimeException("Estudiante no encontrado con id " + id);
    }
    estudianteRepository.deleteById(id);
}
```

paso 6. Con el Maven le damos el clean para limpiar y que no se compile lo viejo y luego package de nuevo para compilarlo de nuevo

docker-compose build backend construimos la imagen del back de nuevo con el Dockerfile
docker-compose up -d levantamos los servicios para correr la app

← → ↺ 🏠 ⓘ localhost:8090/swagger-ui/index.html

Swagger
Supported by SMARTBEAR

/v3/api-docs

Explore

OpenAPI definition v0 OAS 3.1

/v3/api-docs

Servers

http://localhost:8090 - Generated server url

estudiante-controller

- GET** /api/estudiantes
- POST** /api/estudiantes
- GET** /api/saludo
- DELETE** /api/estudiantes/{id}

Activar Windows
Ve a Configuración para activar Windows.

Schemas

Para probarlo primero verificamos que haya estudiantes con el GET/api/estudiantes

← → ↻ 🏠 ⓘ localhost:8090/swagger-ui/index.html#/estudiante-co... 🔍 ☆ ⬇️ 👤

☐☐☐☐

Servers
http://localhost:8090 - Generated server url ▾

estudiante-controller

GET /api/estudiantes

Parameters Cancel

No parameters

Execute Clear

Responses

Curl

```
curl -X 'GET' \
'http://localhost:8090/api/estudiantes' \
-H 'accept: */*'
```

Request URL

```
http://localhost:8090/api/estudiantes
```

Server response

Code	Details
200	<p>Response body</p> <pre>[]</pre> <p>Response headers</p> <pre>connection: keep-alive content-type: application/json date: Wed, 17 Sep 2025 05:49:08 GMT keep-alive: timeout=60 transfer-encoding: chunked vary: Origin,Access-Control-Request-Method,Access-Control-Request-Headers</pre>

Responses

Code	Description	Links
------	-------------	-------

Como no hemos agregado estudiantes está la lista vacía, vamos a agregar estudiantes para poder comprobar el método DELETE

Agregamos estudiantes para hacer la prueba

← → ↻ 🏠 ⓘ localhost:8090/swagger-ui/index.html#/estudiante-co... 📄 🔍 ☆ ⬇️ 👤 ⋮

🗖

POST /api/estudiantes

Parameters Cancel Reset

No parameters

Request body required application/json

Edit Value | Schema

```
{
  "nombres": "Juliana",
  "apellidos": "Casas Ramirez",
  "email": "julianacasasramirez@gmail.com"
},
{
  "nombres": "Angie",
  "apellidos": "Diaz Abaunza",
  "email": "angied@gmail.com"
},
{
  "nombres": "Natalia",
  "apellidos": "Flórez Guzmán",
  "email": "nataliaflorez@gmail.com"
}

```

Execute Clear

Responses

Curl

```
curl -X 'POST' \
'http://localhost:8090/api/estudiantes' \
-H 'accept: */*' \
-H 'Content-Type: application/json' \
-d '
{
  "nombres": "Juliana",
  "apellidos": "Casas Ramirez",
  "email": "julianacasasramirez@gmail.com"
},
{
  "nombres": "Angie",
  "apellidos": "Diaz Abaunza",
  "email": "angied@gmail.com"
},
{
  "nombres": "Natalia",
  "apellidos": "Flórez Guzmán",
  "email": "nataliaflorez@gmail.com"
}
'

```

Verificamos de nuevo la lista de estudiantes

← → ↻ 🏠 ⓘ localhost:8090/swagger-ui/index.html#/estudiante-controll... 🔍 ☆ 👤 ⋮

☐☐

estudiante-controller

GET /api/estudiantes

Parameters Cancel

No parameters

Execute Clear

Responses

Curl

```
curl -X 'GET' \
'http://localhost:8090/api/estudiantes' \
-H 'accept: */*'
```

Request URL

```
http://localhost:8090/api/estudiantes
```

Server response

Code Details

200

Response body

```
[
  {
    "id": 3,
    "nombres": "Juliana",
    "apellidos": "Casas Ramirez",
    "email": "julianacasasramirez@gmail.com"
  },
  {
    "id": 6,
    "nombres": "Angie",
    "apellidos": "Díaz Abaunza",
    "email": "angied@gmail.com"
  },
  {
    "id": 7,
    "nombres": "Natalia",
    "apellidos": "Flórez Guzmán",
    "email": "nataliaflorez@gmail.com"
  }
]
```

Response headers

```
connection: keep-alive
content-type: application/json
date: Wed, 17 Sep 2025 06:00:13 GMT
keep-alive: timeout=60
transfer-encoding: chunked
vary: Origin, Access-Control-Request-Method, Access-Control-Request-Headers
```

Download

Ahora para hacer el ensayo vamos a borrar la estudiante 7

DELETE

/api/estudiantes/{id}

^

Parameters

Cancel

Name	Description
id <small>* required</small>	
integer(\$int64)	7
(path)	

Execute

Clear

Responses

Curl

```
curl -X 'DELETE' \
'http://localhost:8090/api/estudiantes/7' \
-H 'accept: */*'
```

Request URL

```
http://localhost:8090/api/estudiantes/7
```

Server response

Code	Details
200	<div><div>Response headers</div><div><pre>connection: keep-alive content-length: 0 date: Wed, 17 Sep 2025 06:00:44 GMT keep-alive: timeout=60 vary: Origin,Access-Control-Request-Method,Access-Control-Request-Headers</pre></div></div>

Responses

Code	Description	Links
200	OK	No links

Volvemos a verificar en la lista de estudiantes que si se haya eliminado

← → ↻ 🏠 ⓘ localhost:8090/swagger-ui/index.html#/estudiante-controll... 📄 🔍 ☆ 👤 ⋮

estudiante-controller

GET /api/estudiantes

Parameters Cancel

No parameters

Execute Clear

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8090/api/estudiantes' \
  -H 'accept: */*'
```

Request URL

```
http://localhost:8090/api/estudiantes
```

Server response

Code	Details
200	<p>Response body</p> <pre>{ "id": 3, "nombres": "Juliana", "apellidos": "Casas Ramírez", "email": "julianacasasramirez@gmail.com" }, { "id": 6, "nombres": "Angie", "apellidos": "Díaz Abaunza", "email": "angied@gmail.com" } }</pre> <p>Response headers</p> <pre>connection: keep-alive content-type: application/json date: Wed, 17 Sep 2025 06:01:17 GMT keep-alive: timeout=60 transfer-encoding: chunked vary: Origin,Access-Control-Request-Method,Access-Control-Request-Headers</pre>

Responses

Code	Description	Links
------	-------------	-------

El método quedó sirviendo correctamente