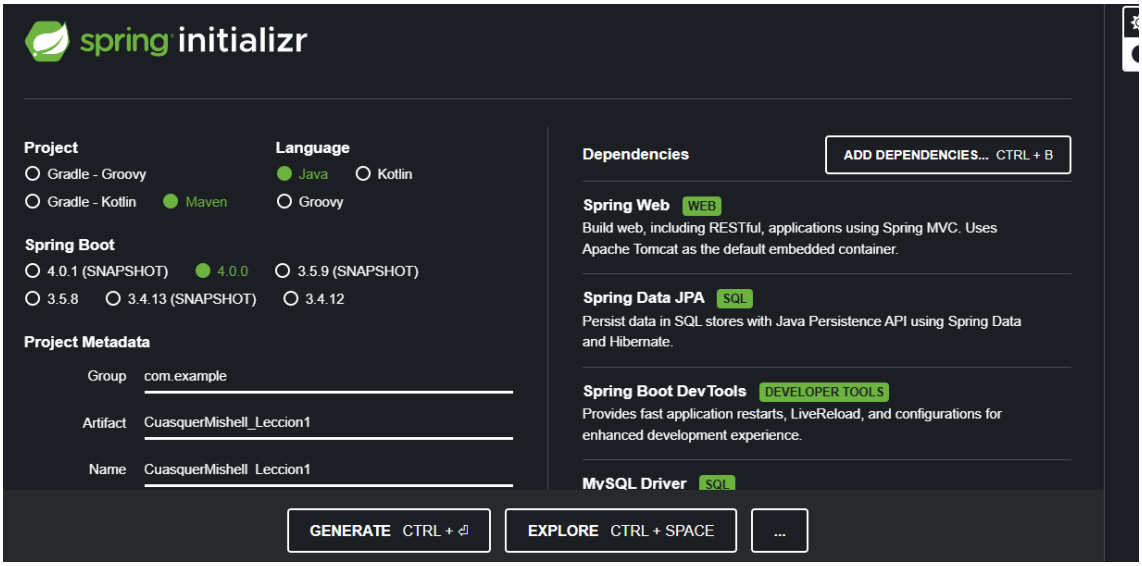
Universidad de las Fuerzas Armadas Espe

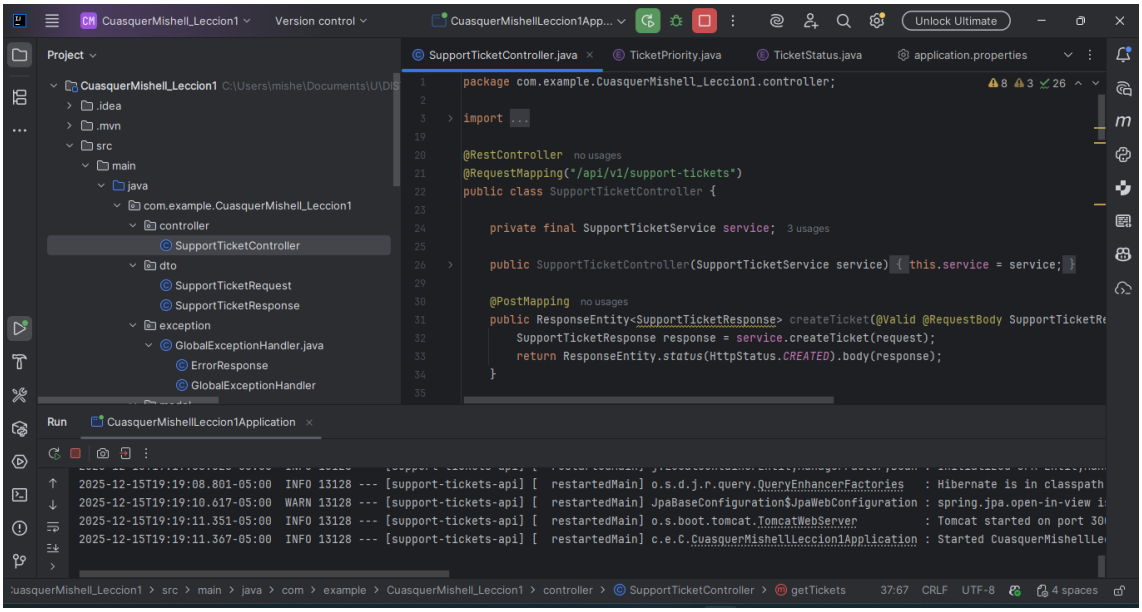
Nombre: Mishell Cuasquer

Taller JPA- Diseño e Implementación de una API RESTful con Filtros

1. Creamos nuestra plantilla



2. Ejecutamos el proyecto

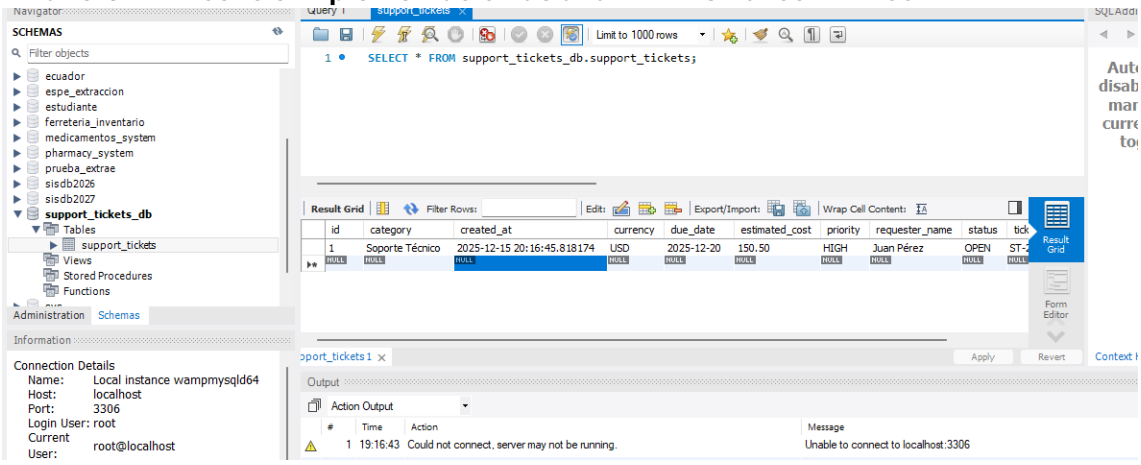


3. Verificamos la conexión a la bd

Universidad de las Fuerzas Armadas Espe

Nombre: Mishell Cuasquer

Taller JPA- Diseño e Implementación de una API RESTful con Filtros



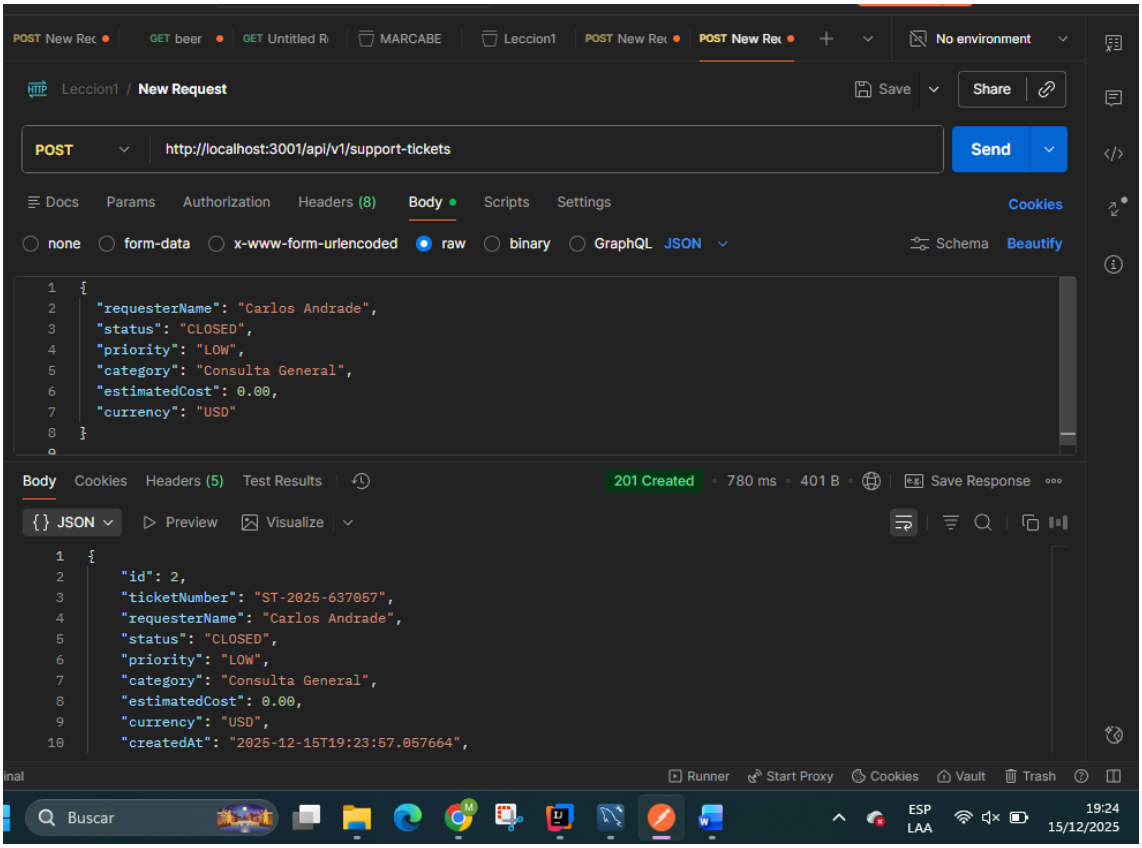
Verificamos y realizamos las pruebas en el postman

1) Crear ticket

POST /api/v1/support-tickets

Registra un nuevo ticket.

createdAt debe ser generado por el sistema.



2) Listar tickets con filtros

GET /api/v1/support-tickets

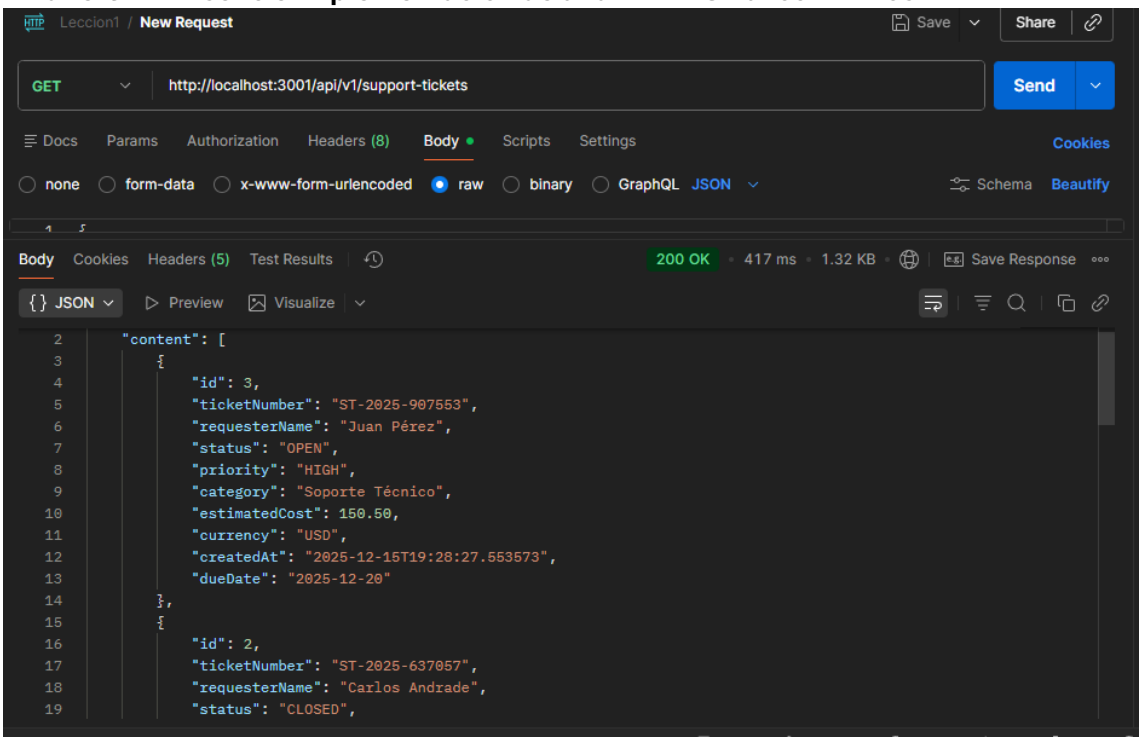
Debe soportar filtros opcionales con query params.

Debe ser paginado y ordenable (page, size, sort).

Universidad de las Fuerzas Armadas Espe

Nombre: Mishell Cuasquer

Taller JPA- Diseño e Implementación de una API RESTful con Filtros



Filtros obligatorios a implementar

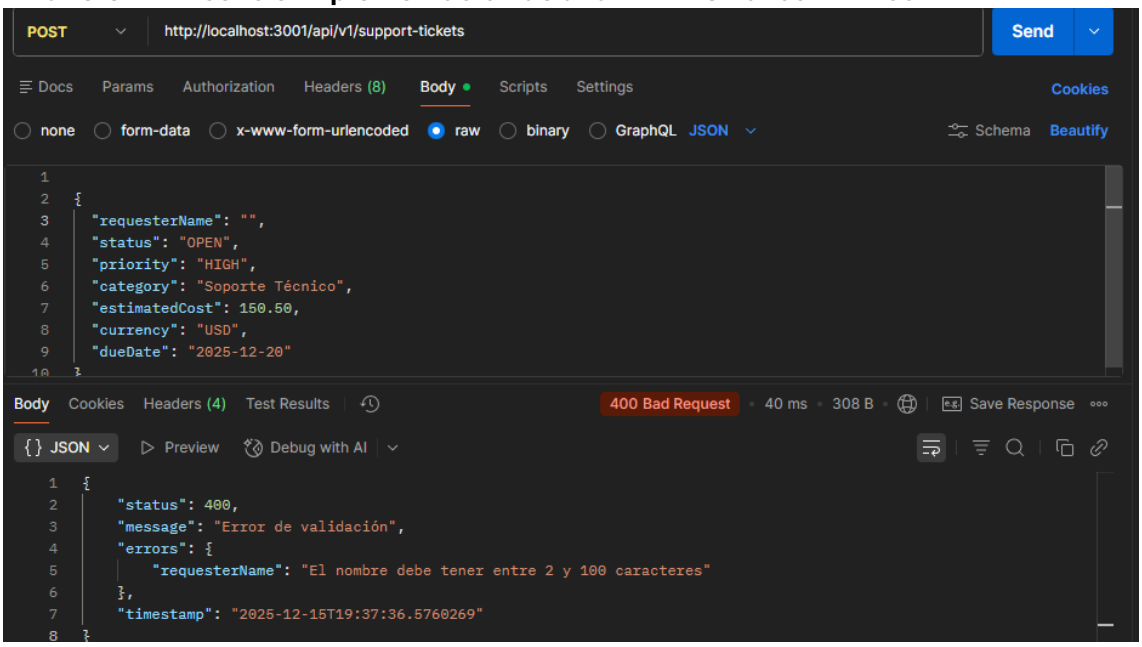
1) Búsqueda de texto (q)
Campos afectados: ticketNumber, requesterName
Búsqueda case-insensitive
Si q está vacío o no se envía → no filtra
Ejemplo:
GET /api/v1/support-tickets?q=chicaiza

Si envoi un dato sin la info del nombre no me permite registrar.

Universidad de las Fuerzas Armadas Espe

Nombre: Mishell Cuasquer

Taller JPA- Diseño e Implementación de una API RESTful con Filtros



2) Estado (status)

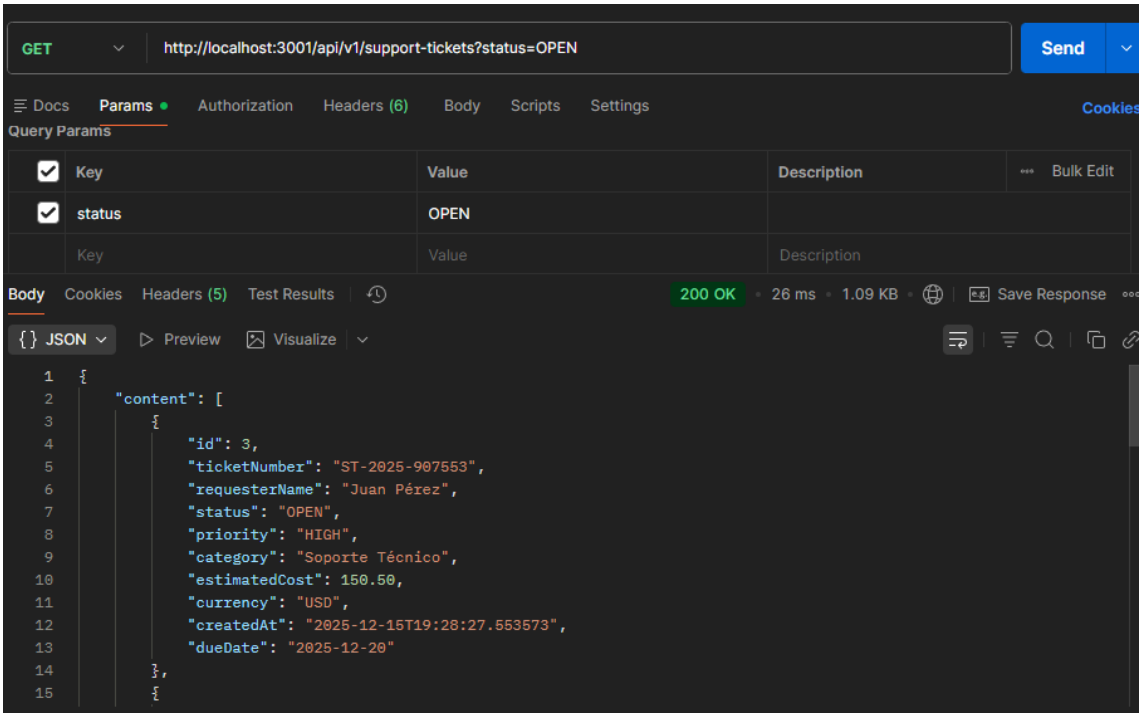
Valores permitidos: OPEN, IN_PROGRESS, RESOLVED, CLOSED, CANCELLED

Si no se envía → no filtra

Valor inválido → 400 Bad Request

Ejemplo:

GET /api/v1/support-tickets?status=OPEN



3) Moneda (currency)

Valores permitidos: USD, EUR

Coincidencia exacta

Valor inválido → 400 Bad Request

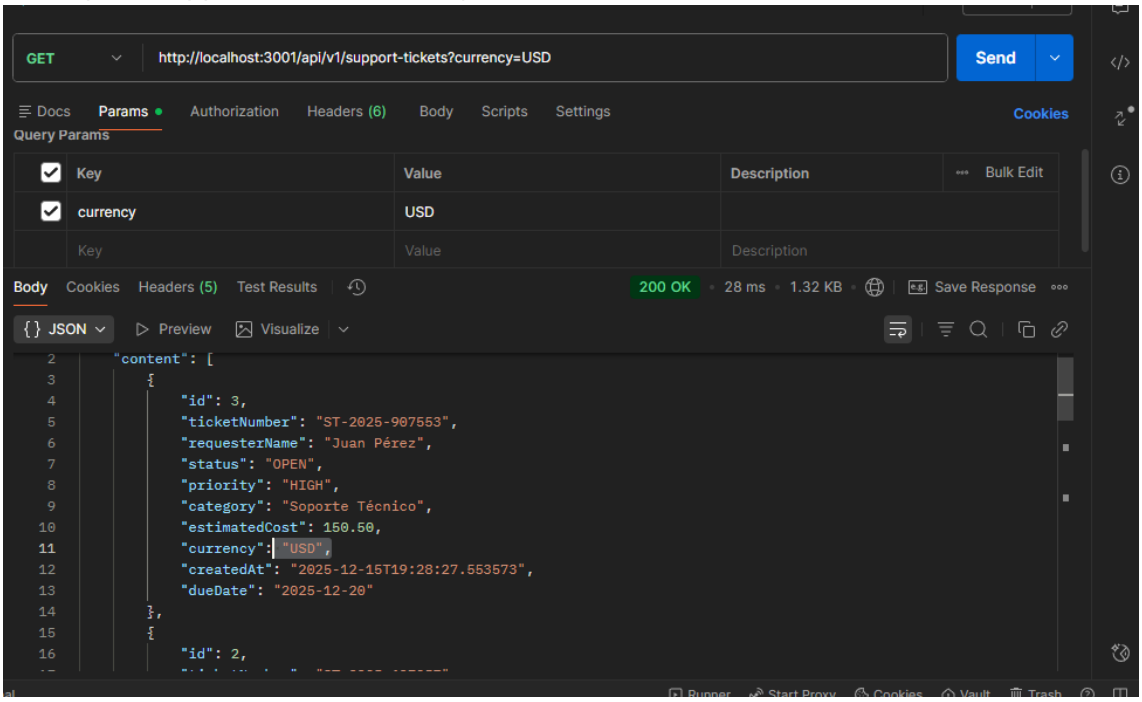
Universidad de las Fuerzas Armadas Espe

Nombre: Mishell Cuasquer

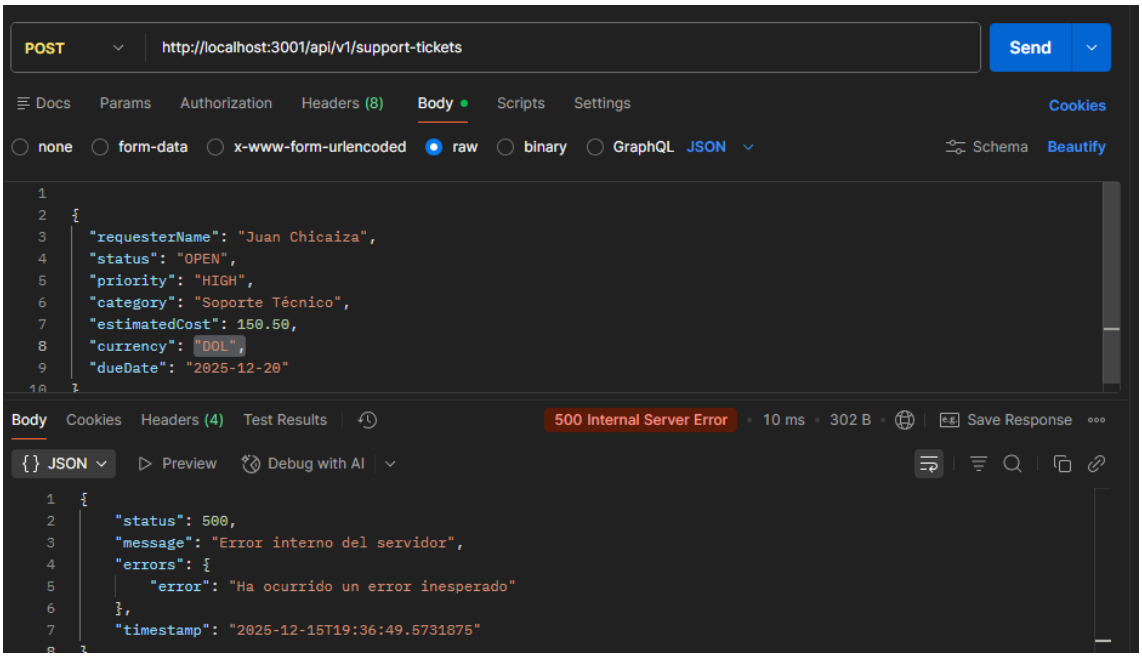
Taller JPA- Diseño e Implementación de una API RESTful con Filtros

Ejemplo:

GET /api/v1/support-tickets?currency=USD



SI coloco otro valor en monedas me sale lo siguiente



4) Monto mínimo (minCost)

Campo afectado: estimatedCost

Regla: estimatedCost >= minCost

Validación: minCost >= 0

Si no se envía → no filtra

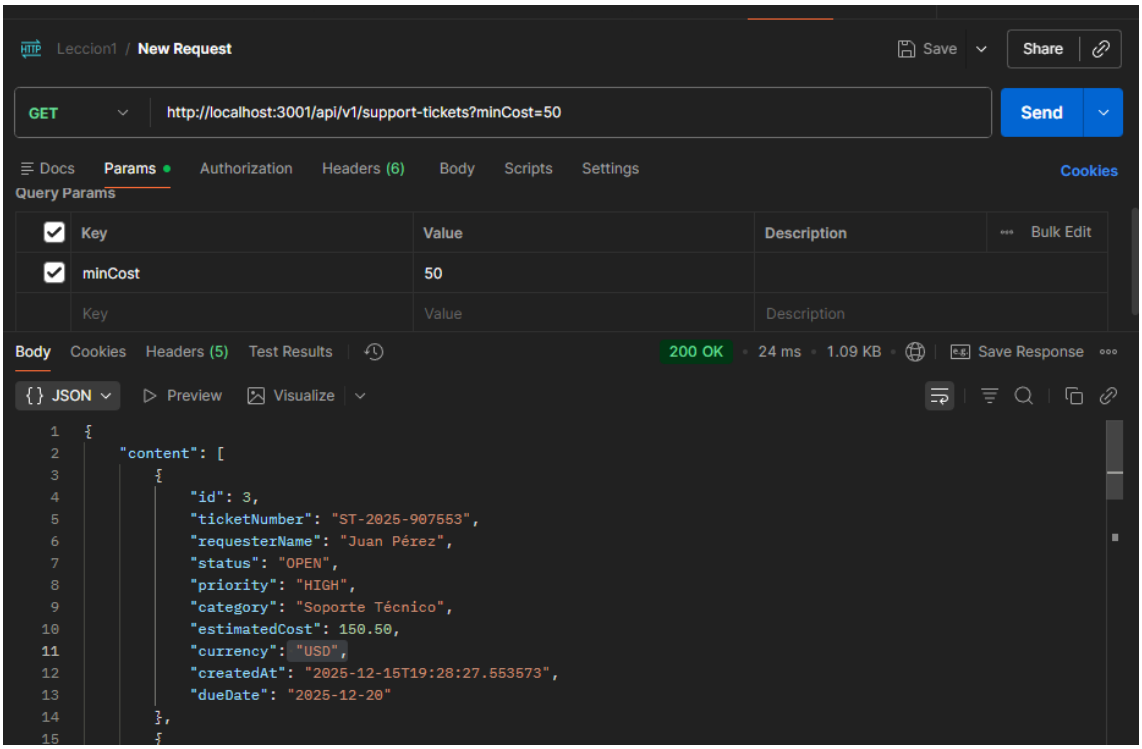
Ejemplo:

Universidad de las Fuerzas Armadas Espe

Nombre: Mishell Cuasquer

Taller JPA- Diseño e Implementación de una API RESTful con Filtros

GET /api/v1/support-tickets?minCost=50



5) Monto máximo (maxCost)

Campo afectado: estimatedCost

Regla: estimatedCost <= maxCost

Validación: maxCost >= 0

Si no se envía → no filtra

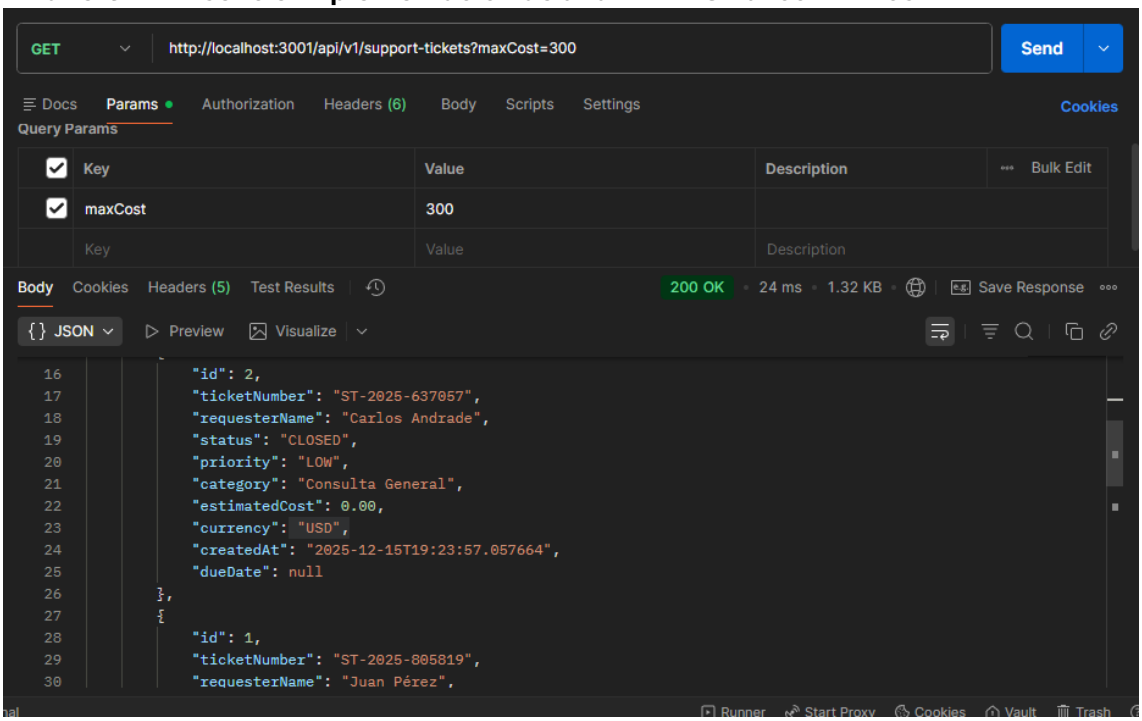
Ejemplo:

GET /api/v1/support-tickets?maxCost=300

Universidad de las Fuerzas Armadas Espe

Nombre: Mishell Cuasquer

Taller JPA- Diseño e Implementación de una API RESTful con Filtros



6) Regla combinada de fechas (from y to)

Campo afectado: createdAt

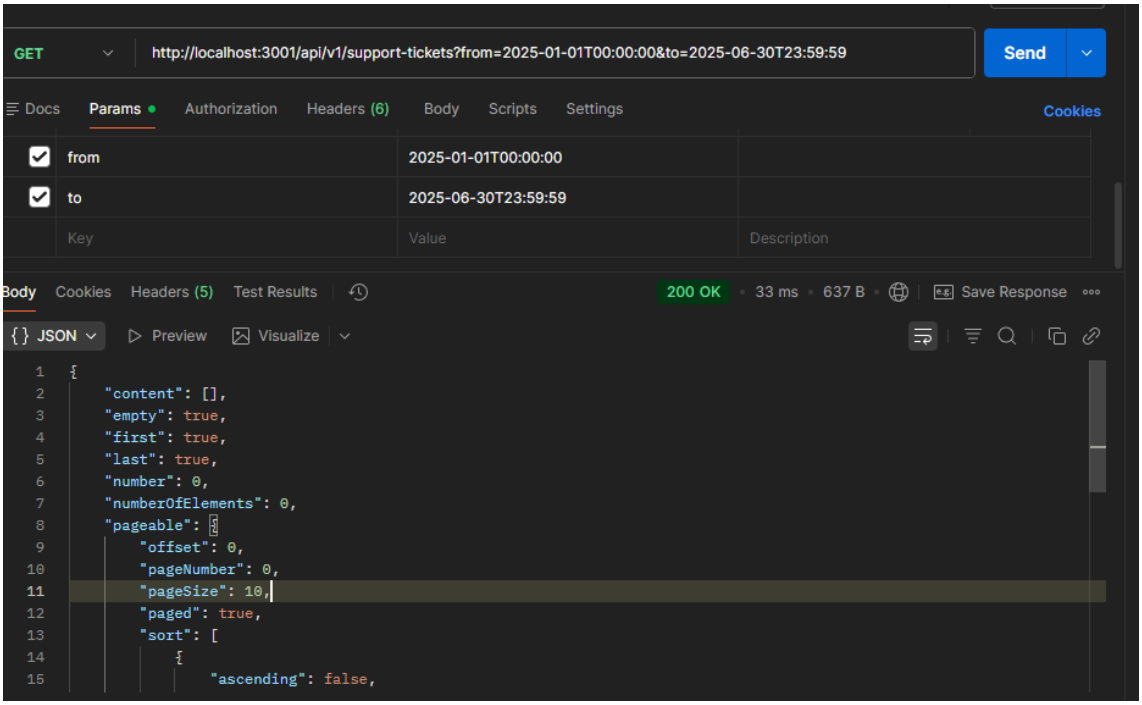
Formato ISO-8601: yyyy-MM-dd'T'HH:mm:ss

Regla: si ambos se envían → from <= to

Si from > to → 400 Bad Request

Ejemplo:

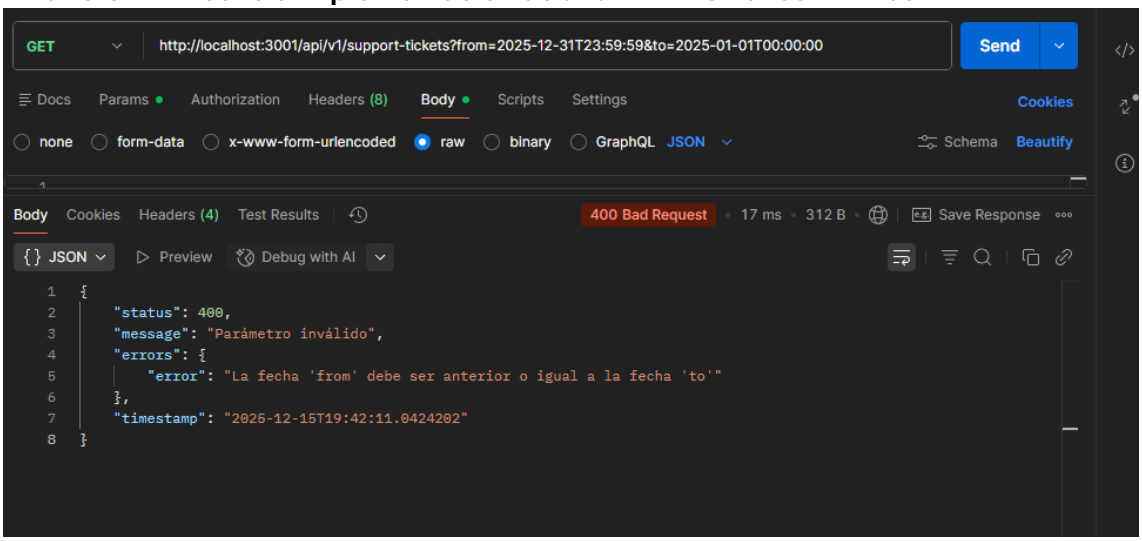
GET /api/v1/support-tickets?from=2025-01-01T00:00:00&to=2025-06-30T23:59:59



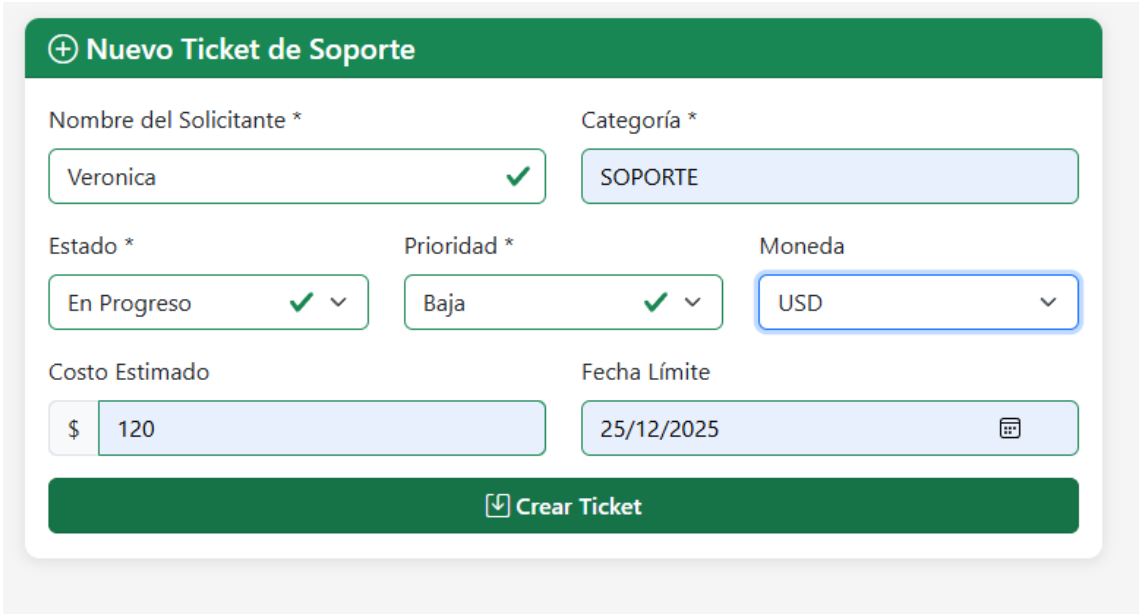
Universidad de las Fuerzas Armadas Espe

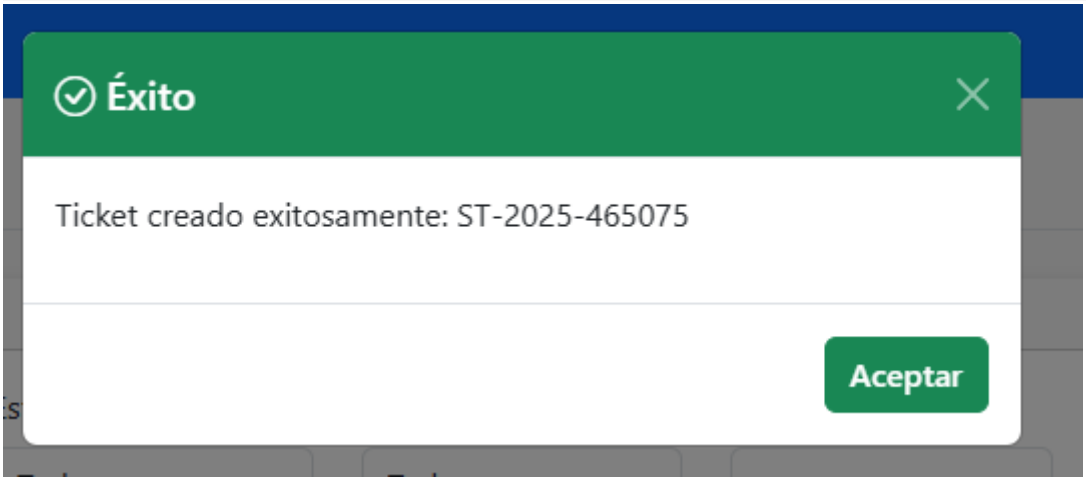
Nombre: Mishell Cuasquer

Taller JPA- Diseño e Implementación de una API RESTful con Filtros



Frond

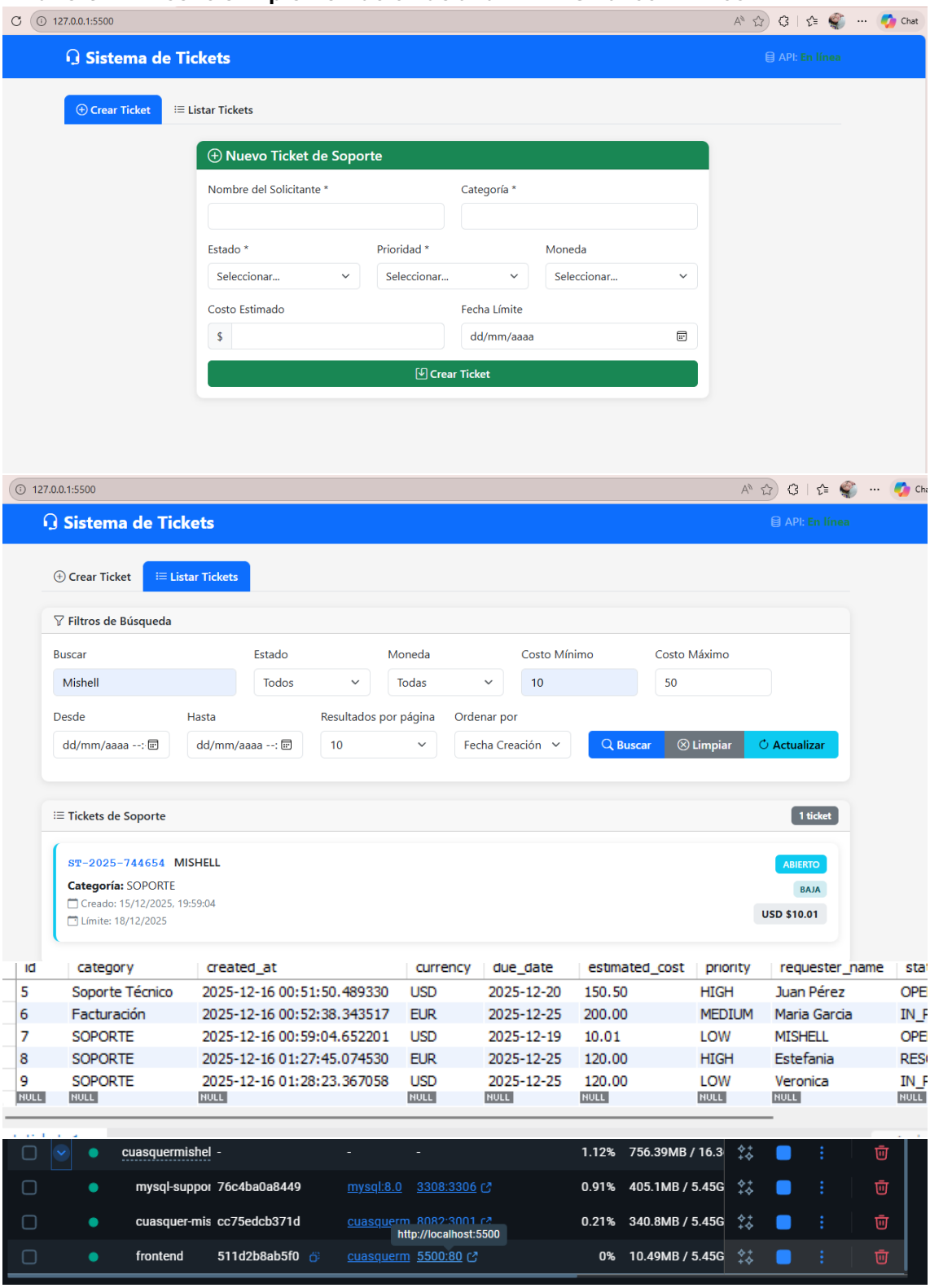




Universidad de las Fuerzas Armadas Espe

Nombre: Mishell Cuasquer

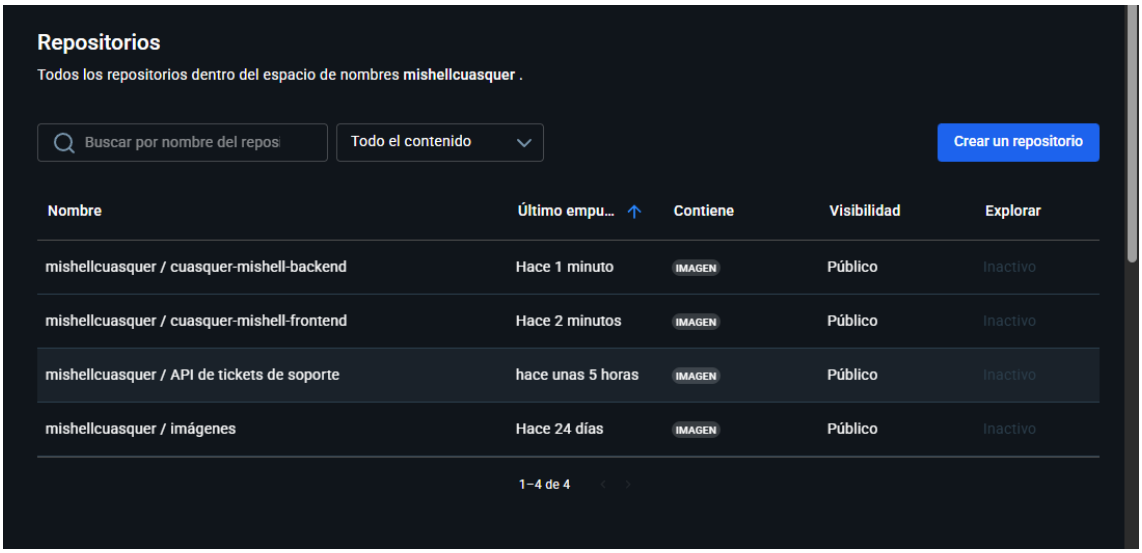
Taller JPA- Diseño e Implementación de una API RESTful con Filtros



Universidad de las Fuerzas Armadas Espe

Nombre: Mishell Cuasquer

Taller JPA- Diseño e Implementación de una API RESTful con Filtros



Link de Github.

https://github.com/MishellCuasquer/ExamenP2Cuasquer.git

Link´s de Dockerbub.

General: https://hub.docker.com/repositories/mishellcuasquer
Backend: https://hub.docker.com/repository/docker/mishellcuasquer/cuasquer-mishell-backend/general
Frond: https://hub.docker.com/repository/docker/mishellcuasquer/cuasquer-mishell-frontend/general