

Examen Practico Progreso 1

Nombre: Andrei Flores

Evidencias:

Funcionamiento Consola

```
PROBLEMS  DEBUG CONSOLE  TERMINAL  PORTS

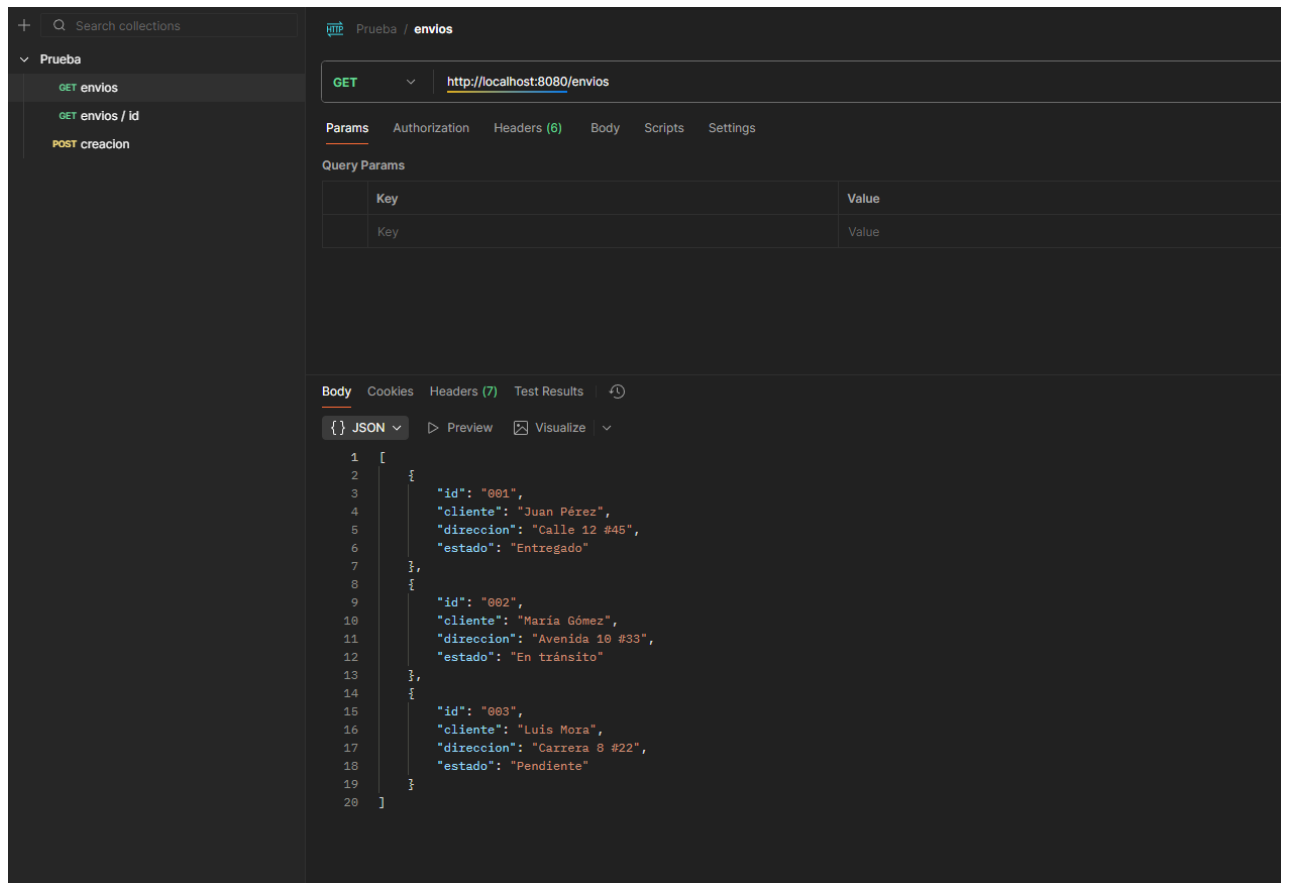
2025-10-29T20:59:19.287-05:00  INFO 4308 --- [inRunController] org.apache.camel.main.MainSupport : Apache Camel (Main) 4.6.0 is starting
2025-10-29T20:59:19.386-05:00  INFO 4308 --- [main] c.n.SingleTCNettyServerBootstrapFactory : ServerBootstrap binding to 0.0.0.0:8080
2025-10-29T20:59:19.432-05:00  INFO 4308 --- [main] o.a.camel.component.netty.NettyConsumer : Netty consumer bound to: 0.0.0.0:8080
2025-10-29T20:59:19.435-05:00  INFO 4308 --- [main] o.a.camel.component.netty.NettyConsumer : Netty consumer bound to: 0.0.0.0:8080
2025-10-29T20:59:19.436-05:00  INFO 4308 --- [main] o.a.camel.component.netty.NettyConsumer : Netty consumer bound to: 0.0.0.0:8080
2025-10-29T20:59:19.439-05:00  INFO 4308 --- [main] o.a.c.impl.engine.AbstractCamelContext : Routes startup (total:5 rest-dsl:4)
2025-10-29T20:59:19.439-05:00  INFO 4308 --- [main] o.a.c.impl.engine.AbstractCamelContext : Started route1 (rest-api:///api-doc)
2025-10-29T20:59:19.439-05:00  INFO 4308 --- [main] o.a.c.impl.engine.AbstractCamelContext : Started get-todos-envios (rest://get:/envios)
2025-10-29T20:59:19.439-05:00  INFO 4308 --- [main] o.a.c.impl.engine.AbstractCamelContext : Started get-envio-id (rest://get:/envios/{%7Bid%7D})
2025-10-29T20:59:19.439-05:00  INFO 4308 --- [main] o.a.c.impl.engine.AbstractCamelContext : Started post-envio (rest://post:/envios)
2025-10-29T20:59:19.439-05:00  INFO 4308 --- [main] o.a.c.impl.engine.AbstractCamelContext : Started filetransfer-carga-csv (file://)
2025-10-29T20:59:19.448-05:00  INFO 4308 --- [main] com.ejemplo.MainApp : Apache Camel 4.6.0 (camel-1) started in 235ms (build:0ms init:0ms start:235ms)
2025-10-29T20:59:19.451-05:00  INFO 4308 --- [main] com.ejemplo.MainApp : Started MainApp in 2.017 seconds (process running for 2.332)
2025-10-29T20:59:20.454-05:00  INFO 4308 --- [ad #2 - file://] com.ejemplo.FileTransferRoute : [INFO] API iniciada en puerto 8080.
2025-10-29T20:59:20.454-05:00  INFO 4308 --- [ad #2 - file://] com.ejemplo.FileTransferRoute : [INFO] Archivo cargado con 3 registros.
2025-10-29T20:59:20.454-05:00  INFO 4308 --- [ad #2 - file://] com.ejemplo.FileTransferRoute : [INFO] Datos transformados a formato JSON.
2025-10-29T20:59:20.460-05:00  INFO 4308 --- [ad #2 - file://] filetransfer-carga-csv : [INFO] Archivo JSON generado en processed/json/envios.json
```

Api – GET envíos

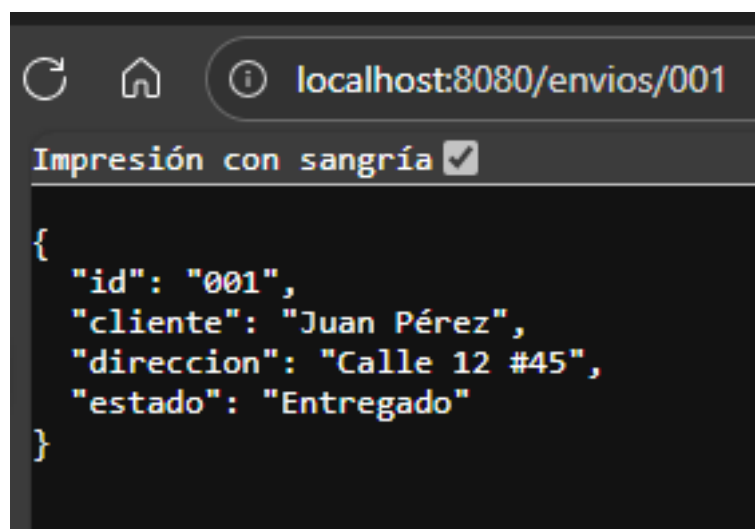
```
localhost:8080/envios

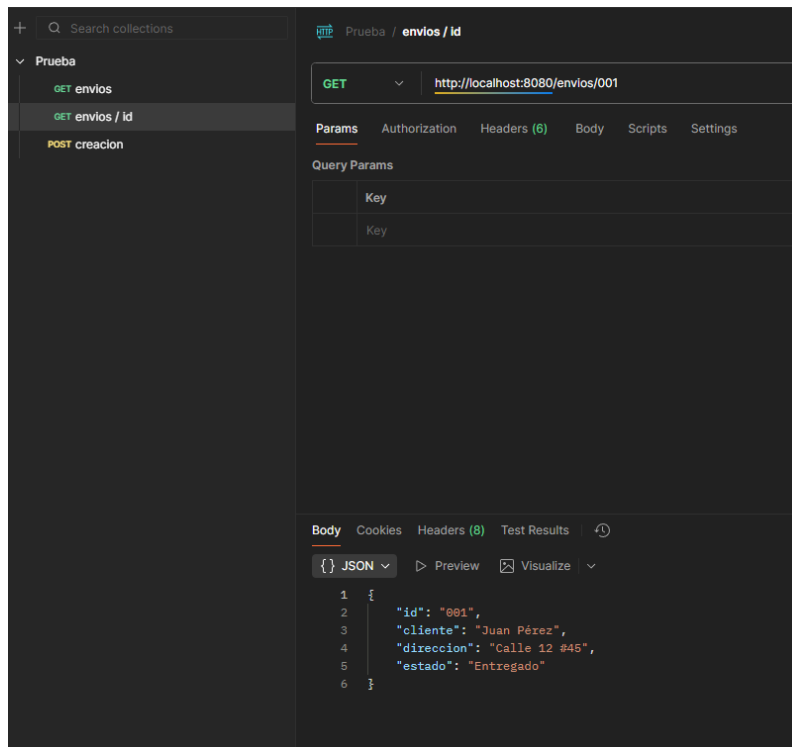
Impresión con sangría ☒

[
  {
    "id": "001",
    "cliente": "Juan Pérez",
    "direccion": "Calle 12 #45",
    "estado": "Entregado"
  },
  {
    "id": "002",
    "cliente": "María Gómez",
    "direccion": "Avenida 10 #33",
    "estado": "En tránsito"
  },
  {
    "id": "003",
    "cliente": "Luis Mora",
    "direccion": "Carrera 8 #22",
    "estado": "Pendiente"
  }
]
```

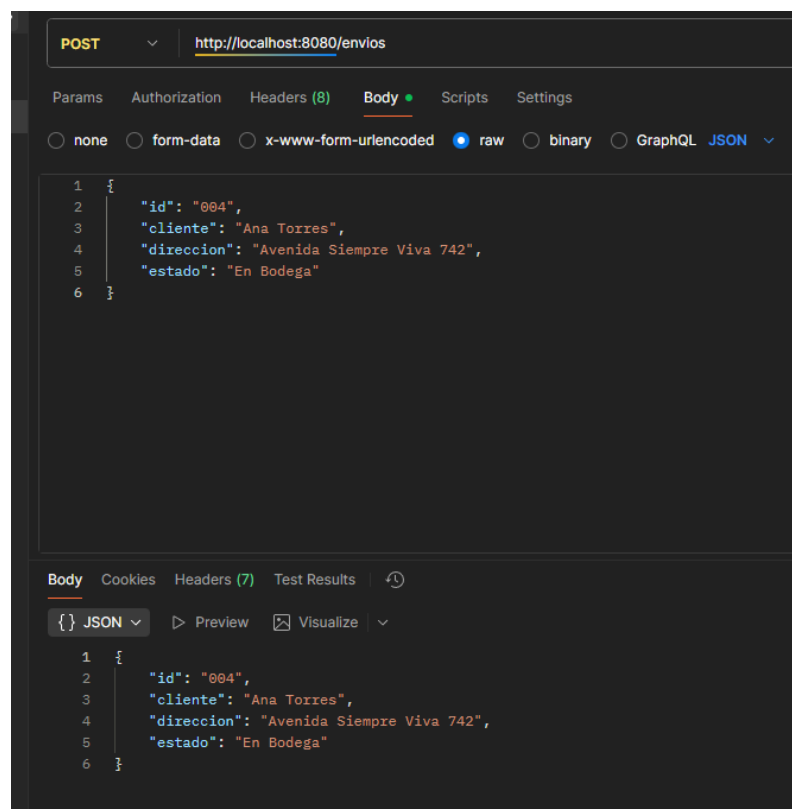


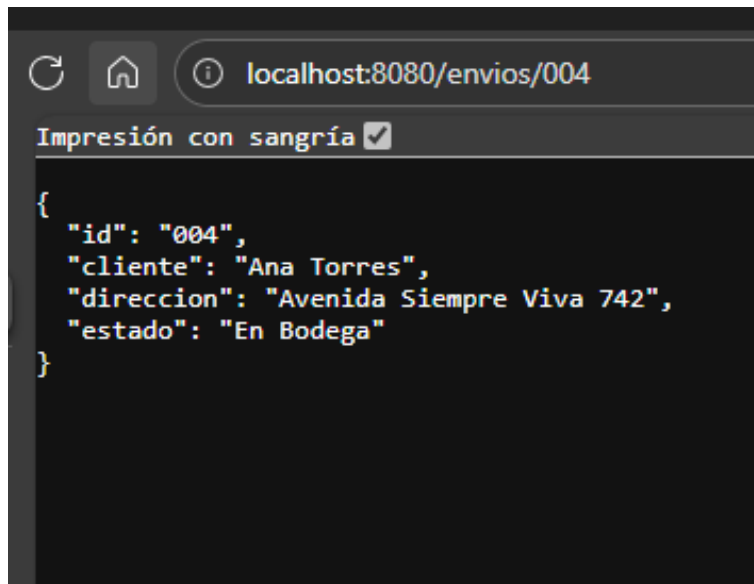
Api – GET envío





Api – POST creacion





```
{
  "id": "004",
  "cliente": "Ana Torres",
  "direccion": "Avenida Siempre Viva 742",
  "estado": "En Bodega"
}
```

Preguntas:

1. ¿Qué patrón de integración aplicaste y cómo se refleja en tu solución?

El patrón que apliqué fue File Transfer, combinado con Polling Consumer y Message Translator. En mi solución, esto se refleja así: la ruta de Camel lee periódicamente un archivo CSV, convierte sus datos en objetos Envío, y luego los serializa a formato JSON. Por otro lado, la API funciona como un servicio REST, exponiendo esos datos.

2. ¿Qué ventajas identificas al pasar de File Transfer a APIs REST?

Pasar a APIs REST aporta acceso en tiempo real y bajo demanda, desacoplamiento, mejor control de seguridad, versionado y observabilidad, además de facilitar escalado horizontal y consumo selectivo de datos.

3. ¿Qué riesgos o limitaciones encontraste en tu enfoque?

Sobreescritura sobre envios.json, datos en memoria sin durabilidad ni transacciones, problemas de concurrencia y consistencia, acoplamiento al formato CSV y falta de control de versiones.

4. ¿Cómo escalarías esta integración si EcoLogistics tuviera 50 sistemas distintos?

Introducir un bus de eventos, usar APIs gestionadas tras un API Gateway, centralizar esquemas con un registry, aplicar consumidores idempotentes, almacenamiento distribuido, autoscaling y observabilidad para garantizar rendimiento y resiliencia.