

Universidad de Las Américas
Facultad de Ingenierías y Ciencias Agropecuarias
Ingeniería De Software
Progreso 1

Nombres: Sammy Porras

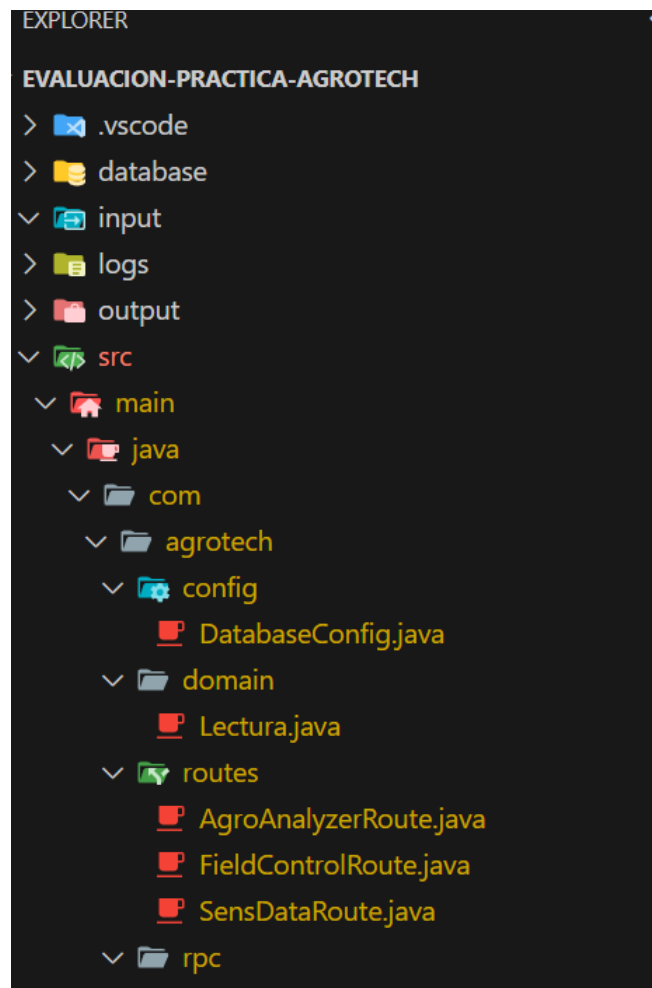
Fecha: 30/10/2025

TRANFERENCIA DE ARCHIVOS – APACHE CAMEL

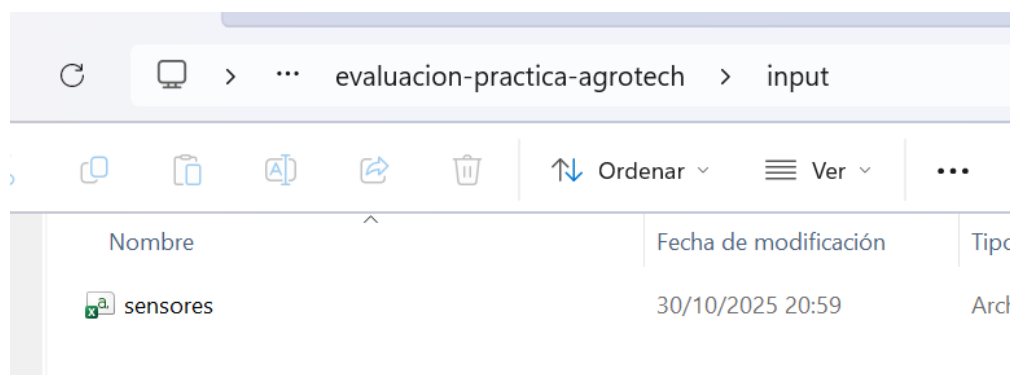
Objetivo de la actividad:

1. Evidencias

Estructura



Archivo en input



Compilación

```
ngestar)
[com.agrotech.App.main()] INFO org.apache.camel.impl.engine.AbstractCamelContext - Stopped sensdata-file-transfer (file:
sin%5Cworkspace%5CIntegracion%20de%20sistemas%5CEvaluacion-practica-agrotech)
[com.agrotech.App.main()] INFO org.apache.camel.impl.engine.AbstractCamelContext - Apache Camel 4.7.0 (camel-1) shutdown in
)
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 32.769 s
[INFO] Finished at: 2025-10-30T20:17:37-05:00
[INFO] -----
```

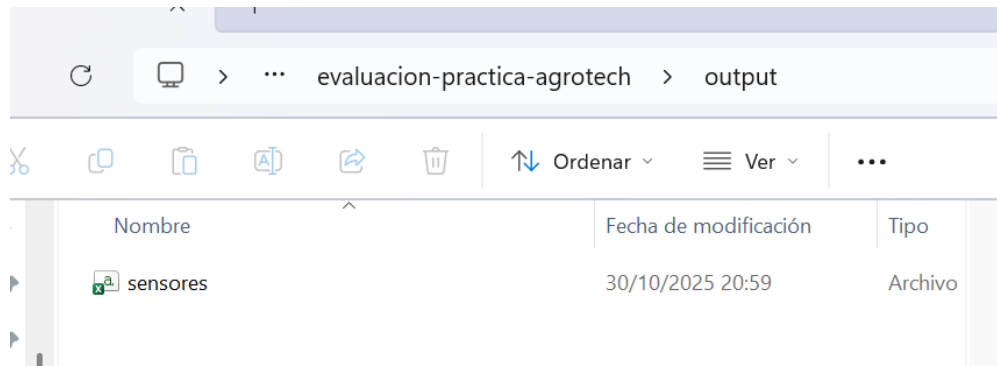
Ejecución

Durante la ejecución, Apache Camel detecta el archivo sensores.csv, procesa cada registro, lo inserta en la base de datos embebida H2 y posteriormente el componente FieldControl muestra las lecturas almacenadas.

```
PS C:\Users\sysin\workspace\Integracion de sistemas\evaluacion-practica-agrotech> mvn exec:java
>>
26.4}, {ID_SENSOR=S002, FECHA=22/5/2025, HUMEDAD=50.0, TEMPERATURA=25.1}, {ID_SENSOR=S003, FECHA=22/5/2025, HUMEDAD=48.0, TEMPE
RATURA=26.3}, {ID_SENSOR=S004, FECHA=22/5/2025, HUMEDAD=42.0, TEMPERATURA=25.5}, {ID_SENSOR=S005, FECHA=22/5/2025, HUMEDAD=55.0
, TEMPERATURA=25.9}]
[Camel (camel-1) thread #2 - timer://fieldcontrol] INFO fieldcontrol-shared-db - [FIELDCONTROL] Última lectura: [{ID_SENSOR=S00
2, FECHA=22/5/2025, HUMEDAD=45.0, TEMPERATURA=26.4}, {ID_SENSOR=S002, FECHA=22/5/2025, HUMEDAD=50.0, TEMPERATURA=25.1}, {ID_SEN
SOR=S003, FECHA=22/5/2025, HUMEDAD=48.0, TEMPERATURA=26.3}, {ID_SENSOR=S004, FECHA=22/5/2025, HUMEDAD=42.0, TEMPERATURA=25.5},
{ID_SENSOR=S005, FECHA=22/5/2025, HUMEDAD=55.0, TEMPERATURA=25.9}, {ID_SENSOR=S001, FECHA=22/5/2025, HUMEDAD=45.0, TEMPERATURA=
26.4}, {ID_SENSOR=S002, FECHA=22/5/2025, HUMEDAD=50.0, TEMPERATURA=25.1}, {ID_SENSOR=S003, FECHA=22/5/2025, HUMEDAD=48.0, TEMPE
RATURA=26.3}, {ID_SENSOR=S004, FECHA=22/5/2025, HUMEDAD=42.0, TEMPERATURA=25.5}, {ID_SENSOR=S005, FECHA=22/5/2025, HUMEDAD=55.0
, TEMPERATURA=25.9}]
[com.agrotech.App.main()] INFO org.apache.camel.impl.engine.AbstractCamelContext - Stopped agroanalyzer-shared-
db (direct://agroanalyzer.ingestar)
[com.agrotech.App.main()] INFO org.apache.camel.impl.engine.AbstractCamelContext - Stopped sensdata-file-transf
er (file://input)
[com.agrotech.App.main()] INFO org.apache.camel.impl.engine.AbstractCamelContext - Apache Camel 4.7.0 (camel-1) shu
tdown in 10ms (uptime:15s)
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 17.263 s
[INFO] Finished at: 2025-10-30T21:07:01-05:00
[INFO] -----
```

Archivo en output

El archivo de entrada se mueve a la carpeta output/ tras su procesamiento, indicando que fue consumido exitosamente.



Se incluye la carpeta de logs como parte de la evidencia

2. Link a repositorio GIT

Se utilizó el siguiente repositorio de git:

<https://github.com/sy-sinai/evaluacion-practica-agrotech.git>

3. Preguntas

1. ¿Qué patrón aplicaste en cada fase del flujo y por qué?

- **File Transfer:** se usó para mover automáticamente el archivo `sensores.csv` desde “SensData” hacia “AgroAnalyzer”, simulando la transferencia de datos entre sistemas.
- **Shared Database:** permitió que “AgroAnalyzer” guardara los datos procesados en una base H2 compartida para que “FieldControl” los consulte directamente.
- **RPC simulado:** representó una comunicación síncrona entre “FieldControl” (cliente) y “AgroAnalyzer” (servidor), emulando una llamada remota de solicitud y respuesta.

2. ¿Qué riesgos observas al usar una base de datos compartida?

- Acopla fuertemente los sistemas.
- Cambios en la estructura pueden romper otros módulos.
- Posibles conflictos de acceso y pérdida de consistencia.
- Escalabilidad y mantenimiento limitados al depender todos del mismo recurso.

3. ¿Cómo ayuda el RPC simulado a representar un flujo síncrono?

El RPC simulado muestra cómo un sistema cliente envía una solicitud al servidor y espera su respuesta antes de continuar el proceso. Esto refleja un flujo síncrono, donde ambos sistemas deben estar disponibles al mismo tiempo para completar la comunicación. En el proyecto, "FieldControl" solicita los datos del sensor a "AgroAnalyzer" y recibe la lectura más reciente, imitando una interacción directa típica de servicios conectados en tiempo real.

4. ¿Qué limitaciones tienen los patrones clásicos frente a arquitecturas modernas?

Los patrones clásicos presentan limitaciones porque fueron pensados para entornos más simples. No escalan fácilmente, generan acoplamiento entre sistemas y suelen trabajar de forma manual o por lotes. En cambio, las arquitecturas modernas como microservicios, permiten comunicación en tiempo real, independencia entre componentes, mayor seguridad y facilidad para mantener o desplegar cambios sin afectar todo el sistema.