

Universidad de Las Américas
Facultad de Ingenierías y Ciencias Agropecuarias
Ingeniería de Software
Informe de laboratorio

1. DATOS DEL ALUMNO:

David Terán

2. TEMA DE LA PRÁCTICA:

Evaluación Práctica Progreso 1 – Caso BioNet

3. OBJETIVO DE LA PRÁCTICA

Diseñar e implementar una solución funcional que permita integrar los sistemas de laboratorio con el sistema central de forma controlada, utilizando los patrones:

- Transferencia de Archivos para recolectar los archivos .csv.
- Base de Datos Compartida para consolidar la información en un sistema común.

4. IDENTIFICACION DEL PROBLEMA

Actualmente, la empresa BioNet enfrenta varios problemas debido a la forma en que se integran los resultados de exámenes clínicos desde distintos laboratorios:

- Datos duplicados o sobrescritos: Debido a que múltiples procesos pueden escribir simultáneamente en la base de datos compartida sin control, se generan inconsistencias en los registros.
- Errores de sincronización: Los archivos CSV que contienen los resultados se suben manualmente, y a menudo están incompletos o corruptos, lo que afecta la integridad de los datos.
- Problemas de concurrencia: Cuando varios laboratorios suben archivos o procesos escriben a la vez en la base de datos, no hay un mecanismo eficiente para manejar conflictos o evitar pérdida de información.
- Falta de automatización: El proceso manual para subir los archivos al servidor FTP genera retrasos y posibles errores humanos, afectando la disponibilidad y calidad de los datos.
- Dificultad en la trazabilidad: No hay un registro claro de qué cambios o inserciones se han hecho en la base de datos, dificultando auditorías o detección de errores.

5. JUSTIFICACIÓN DE USO DE LOS PATRONES TRANSFERENCIA DE ARCHIVOS Y BD COMPARTIDA.

- Transferencia de Archivos:
 - Facilita la integración entre sistemas heterogéneos sin requerir conexión directa permanente.
 - Permite la entrega diaria o periódica de resultados mediante archivos CSV, lo que se adapta a la realidad actual de los laboratorios.
 - Ayuda a desacoplar los sistemas: los laboratorios generan archivos de forma independiente y el sistema central procesa los datos cuando están disponibles.

- El control sobre la integridad y validación de los archivos antes de procesarlos reduce la incidencia de errores y datos corruptos.
- Base de Datos Compartida:
 - Centraliza la información de resultados en un único repositorio, lo que facilita la consulta y generación de reportes.
 - Permite establecer restricciones, índices únicos y triggers para mantener la integridad y evitar duplicados.
 - Proporciona un mecanismo de trazabilidad mediante triggers que registran las operaciones de inserción o actualización.
 - Facilita la sincronización controlada y la gestión concurrente de múltiples fuentes de datos.

6. DISEÑO DE ALTO NIVEL DE LA SOLUCIÓN

El diseño propuesto considera la modularidad, la validación de datos, y la trazabilidad de los cambios en los registros. Se estructura en tres grandes bloques: ingreso de archivos, procesamiento y validación, y consolidación en base de datos.

▪ Arquitectura

Laboratorios → CSV diarios → Carpeta /input-labs → Script de procesamiento

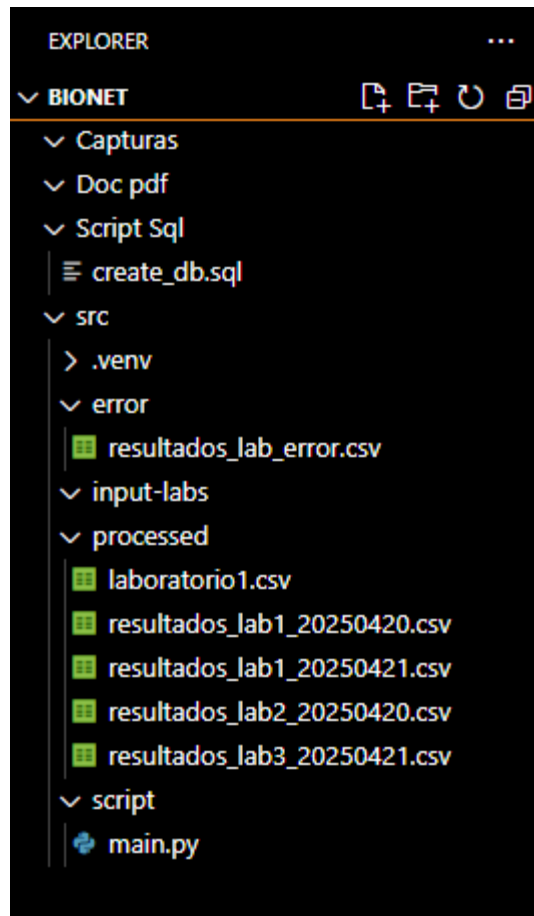
→

→ Validación → BD (resultados_exámenes) → Trigger → log_cambios_resultados

▪ Componentes claves

- **Input:** Carpeta input-labs/ que actúa como punto de entrada de los archivos .csv.
- **Procesamiento:** Un script en Python que:
 - Verifica la validez del archivo (estructura, campos obligatorios, sin errores).
 - Inserta en la base solo si el archivo es válido.
 - Mueve los archivos válidos a processed/ y los inválidos a error/.
- **Base de datos:** SQL Server, donde se consolidan los resultados.
 - Evita duplicados con un índice único.
 - Implementa un trigger que registra las operaciones.
- **Logs:** La tabla log_cambios_resultados permite rastrear qué datos se insertaron o actualizaron y cuándo.

7. ESTRUCTURA DE CARPETAS



8. FLUJO DE INTEGRACIÓN

- Generación de archivos CSV: Cada laboratorio crea archivos CSV diarios con los resultados de exámenes.
- Depósito en carpeta /input-labs: Los archivos CSV son copiados o transferidos a una carpeta de entrada centralizada.
- Validación y procesamiento: Un script Python lee los archivos, valida que estén completos y correctamente formateados.
 - Si el archivo es válido, se insertan los datos en la base de datos.
 - Si hay errores, el archivo se mueve a una carpeta /error para revisión.
- Movimientos de archivos: Los archivos procesados válidos se mueven a /processed para evitar reprocesamiento.
- Registro en base de datos: Los datos se insertan en la tabla resultados_examenes.
- Trazabilidad: Un trigger registra cada inserción o actualización en la tabla log_cambios_resultados.
- Consulta centralizada: El sistema central accede a la base de datos para consultas y reportes unificados.

9. ESQUEMA DE BASE DE DATOS SUGERIDO

- Log_cambios_resultados

	Column Name	Data Type	Allow Nulls
▶	id	int	<input type="checkbox"/>
	operacion	varchar(10)	<input type="checkbox"/>
	paciente_id	int	<input type="checkbox"/>
	tipo_examen	varchar(100)	<input type="checkbox"/>
	fecha	datetime	<input type="checkbox"/>
			<input type="checkbox"/>

- Resultados_exámenes

	Column Name	Data Type	Allow Nulls
▶	id	int	<input type="checkbox"/>
	laboratorio_id	int	<input type="checkbox"/>
	paciente_id	int	<input type="checkbox"/>
	tipo_examen	varchar(100)	<input type="checkbox"/>
	resultado	varchar(100)	<input type="checkbox"/>
	fecha_examen	date	<input type="checkbox"/>
			<input type="checkbox"/>