

Informe de Avance del Proyecto: Sistema de Monitoreo Electroquímico para Detección de Metales Pesados

1. Objetivo Principal del Proyecto

El objetivo principal de este sistema es desarrollar una solución digital que permita al usuario realizar mediciones de metales pesados en muestras de agua mediante sensores electroquímicos conectados a un potenciostato PalmSens. El sistema deberá procesar estos datos usando técnicas como voltametría cíclica, aplicar análisis quimiométrico (PCA) y estimar concentraciones en ppm, mostrando al usuario los resultados de forma intuitiva a través de una interfaz gráfica.

2. Recomendaciones Clave de la Profesora Angela

- Eliminar el primer ciclo de cada curva y usar los 4 siguientes para generar un promedio de corrientes por potencial.
- Transformar esos datos promedio (corrientes) de columna a fila para alimentar una matriz.
- La matriz alimenta un análisis de componentes principales (PCA) para clasificar las muestras.
- Asociar los resultados del PCA con valores aproximados en ppm según un factor de conversión.
- La interfaz gráfica debe permitir al usuario:
 - Cargar archivos `.pssession`.
 - Ver el Session ID generado.
 - Consultar por fecha o ID.
 - Visualizar los datos gráficamente.
 - Recibir alertas de contaminación.

3. Estructura de la Base de Datos (PostgreSQL)

La base de datos se llama `deteccion_metales` y se compone de las siguientes tablas principales:

- `sessions`:
 - `id`: serial (PK)
 - `filename`: nombre del archivo `.pssession`

- `loaded_at`: timestamp (NOT NULL)
- `measurements`:
 - `id`: serial (PK)
 - `session_id`: FK a `sessions`
 - `title`: título de la medición
 - `timestamp`: fecha de medición
 - `device_serial`: número de serie del dispositivo
 - `curve_count`: número de curvas
 - `pca_data`: arreglo de floats
 - `ppm_estimations`: arreglo de floats
- `curves`, `points` (no utilizados directamente hasta este punto, pero forman parte del esquema completo)

4. Estructura del Proyecto

COINVESTIGACION1/

```

├── .venv/
├── .vscode/
├── data/
│   ├── b_d.txt
│   ├── BD/
│   ├── Captura de pantalla 2025-03-01 150555.png
│   ├── Coinvestigacion_UM.docx
│   ├── CV SPE-M11_Induma.pssession
│   ├── CV SPE-M39_Induma.pssession
│   ├── INFORME 2.pdf
│   └── informe.pdf

```

- | |— Matriz_de_datos_Muestra_180225.xlsx
- | |— matriz_oca.csv
- | |— Muestra de prueba.xlsx
- | |— Planificador de proyectos de Gantt_Coinvesti...
- | |— ultima_medicion.pssession
- |— sdk/
 - | |— PSPythonSDK/
 - | |— pspython/
 - | |— cv.psmethod
 - | |— CVPeakExample.py
 - | |— Demo CV DPV EIS IS-C electrode.pssession
 - | |— eis.1161.b.pssession
 - | |— LoadEISWithCircuitFit.py
 - | |— LoadExample.py
 - | |— MeasurementExample.py
 - | |— PSDiffPulse.psmethod
 - | |— readme.txt
 - | |— requirements.txt
 - | |— swv.psmethod
- |— src/
 - |— __pycache__/
 - |— __init__.py
 - |— db_connection.py
 - |— insert_data.py
 - |— interfaz_grafica.py
 - |— main.py

```
|— pstrace_connection.py
|— pstrace_session.py
|— debug.log
|— limits_ppm.json
|— requirements.txt
└— schema.sql
```

5. Cambios Implementados Hasta Ahora

- Se eliminan los primeros ciclos de cada medición y se promedian los siguientes cuatro.
- Se genera la matriz PCA a partir de estos promedios.
- Se multiplica el vector PCA por un factor de conversión en `limits_ppm.json`.
- Se genera `matriz_pca.csv` incluyendo encabezados de PCA y PPM.
- Se corrigió la carga de archivos `.pssession` y se integró con `pgadmin4`.
- Se evita el error de campo `loaded_at` nulo agregando `datetime.now()` al momento de inserción.
- Se mostró correctamente un mensaje en la interfaz sobre carga de archivo, incluyendo el ID de sesión.
- Se puede consultar por ID o fecha, y se cargan las sesiones en un combobox para selección rápida.
- El sistema emite alerta visual si se detectan corrientes elevadas.

6. Problemas Solucionados

- Conflicto con permisos al generar el CSV.
- Fallo de inicialización de Python.NET.
- Error en la inserción SQL debido a campo `loaded_at` nulo.
- Problema de lectura en JSON por salida mixta (stdout + logs).

7. Cambios por Implementar

- Mejorar el mensaje visual de carga para que el ID de sesión sea más visible y fácil de copiar.
- Implementar una pestaña o cuadro de información detallado por sesión (ej. potenciales, corrientes, ppm).
- Incluir exportación manual de resultados a CSV o PDF desde la interfaz.
- Agregar validación visual si el archivo no contiene suficientes ciclos.
- Optimizar integración con el PCA real y no simulado por ahora.
- Incorporar autenticación de usuario si se convierte en aplicación multiusuario.

8. Conclusión

El sistema ha avanzado de forma significativa. Se han implementado la mayor parte de los requisitos de la profesora Angela y ya existe integración completa entre procesamiento de datos, generación de PCA/ppm, y visualización gráfica. La base de datos está en funcionamiento y las pruebas de inserción y consulta han sido exitosas. Falta ahora completar detalles visuales y de interacción para garantizar una experiencia de usuario intuitiva y confiable.