
Informe Final del Proyecto

Diseño de un Sistema Inteligente Basado en Biosensores y Sensores Electroquímicos para la Detección de Metales Pesados en Aguas Residuales

1. Resumen Ejecutivo

El proyecto tiene como objetivo desarrollar un sistema inteligente, portable y de bajo costo para la detección en tiempo real de metales pesados en aguas residuales. Se integran biosensores y sensores electroquímicos, se gestionan los datos mediante una base de datos (PostgreSQL) y se ofrece una interfaz de usuario intuitiva. Este informe consolida los avances de los laboratorios (Lab1 a Lab4) y detalla la metodología, la organización del proyecto y los próximos pasos.

2. Introducción y Objetivos

2.1. Introducción

Ante la necesidad de monitorear de forma eficiente la calidad del agua, se propone un sistema que permita la detección temprana de metales pesados en aguas residuales. Este sistema se basa en la integración de un conjunto de sensores, el procesamiento de datos mediante técnicas de machine learning y la conexión de dispositivos mediante el SDK de PalmSens.

2.2. Objetivos del Proyecto

- **Objetivo General:**
Desarrollar un sistema inteligente que integre la adquisición, procesamiento y visualización en tiempo real de datos para identificar la presencia de metales pesados en aguas residuales.
- **Objetivos Específicos:**
 - Diseñar una interfaz de usuario intuitiva para el monitoreo y control del sistema.

- Implementar algoritmos de procesamiento y clasificación basados en machine learning.
- Conectar el potenciostato al sistema mediante el SDK de PalmSens y gestionar la adquisición de datos.
- Configurar una base de datos en PostgreSQL para el almacenamiento organizado de la información.
- Integrar el sistema en un entorno IoT que permita la transmisión remota de datos.

3. Estructura del Proyecto y Organización de Carpetas

Basándose en el documento PDF "estructura_carpetas_Converted.pdf", se define la siguiente organización del proyecto, donde se corrige la estructura interna de la carpeta **src**:

yaml

CopiarEditar

Coinvestigacion_UM/

```

├── .vscode/
│   └── settings.json
├── data/
│   ├── Coinvestigacion_UM.docx           # Documento principal del
proyecto
│   ├── estructura.txt
│   ├── Captura de pantalla 2025-03-01 150555.png
│   ├── Muestra de prueba.xlsx
│   └── Planificador de proyectos de Gantt_Coinvestigacion.xlsx
├── interfaz/
│   └── inter.html                       # Prototipo de la interfaz
de usuario
├── proyecto/                           # Carpeta dedicada al
desarrollo del sistema
│   ├── sdk/
│   │   └── PSPythonSDK/
│   │       └── pspython/
│   │           ├── pspymethods.py
│   │           └── PalmSens.Core.xml
│   ├── src/
│   │   └── main.py                     # Script principal para la
conexión y pruebas con PSTrace

```

```

|   |   | — db_connection.py           # Gestión de la base de
datos PostgreSQL
|   |   | — insert_data.py             # Script para inserción de
datos en la base de datos
|   |   | — pstrace_connection.py       # Conexión al potencioestado
a través del SDK de PalmSens
|   |   | — query_visualize.py         # Consultas y visualización
de los datos
|   | — venv/                          # Entorno virtual para el
proyecto
|   | — docs/
|       | — informes/                  # Informes de laboratorio
(Lab1, Lab2, Lab3, Lab4)
|       | — lab1.docx
|       | — lab2.docx
|       | — lab3.docx
|       | — lab4.docx
| — MethodSCRIPT_Examples/             # Ejemplos y proyectos
relacionados
    | — (Subdirectorios y archivos correspondientes)

```

Esta estructura separa claramente la documentación y los informes (ubicados en **data** y **docs/informes**) de los componentes técnicos y de desarrollo (almacenados en **proyecto/src**, **proyecto/sdk** y **proyecto/venv**). La carpeta **src** ahora contiene los scripts actualizados:

- **main.py**
- **db_connection.py**
- **insert_data.py**
- **pstrace_connection.py**
- **query_visualize.py**

4. Metodología y Desarrollo

4.1. Configuración del Entorno de Desarrollo

- **IDE y Entorno Virtual:**
Se utiliza Visual Studio Code y se gestiona un entorno virtual (venv) para las dependencias.

- **Librerías y Dependencias:**

Se instalaron librerías como psycopg2, pandas, matplotlib, dash, flask, scikit-learn y pythonnet para la conexión a la base de datos, el procesamiento de datos y el desarrollo de la interfaz.

4.2. Integración del SDK de PalmSens y Conexión con PSTrace

- **Configuración del SDK:**

El SDK se ubica en `proyecto/sdk/PSPythonSDK/pspython` y se añade al `sys.path` en `main.py` para importar el módulo `pspymethods`.

Solución de Problemas:

Se resolvió el error de carga del ensamblado (PalmSens.Core.dll) mediante el desbloqueo del archivo, utilizando el comando de PowerShell:

mathematica

CopiarEditar

```
Unblock-File -Path
```

```
"C:\coinvestigacion\sdk\PSPythonSDK\pspython\Palmsens.Core.dll"
```

-
- **Verificación de Conexión:**

Se implementó una función de prueba en `main.py` que verifica la correcta importación y conexión a PSTrace 5.9.

4.3. Desarrollo y Configuración de la Base de Datos

- **Diseño del Esquema:**

Se han creado tablas para gestionar sensores, mediciones y usuarios, incluyendo índices para la optimización de consultas.

- **Conexión desde Python:**

El script `db_connection.py` utiliza psycopg2 para conectar a PostgreSQL, permitiendo la creación y actualización de las tablas.

4.4. Desarrollo de la Interfaz de Usuario

- **Objetivos de la UI:**

Crear una interfaz intuitiva que permita visualizar en tiempo real los datos, controlar la adquisición y generar gráficos dinámicos.

- **Herramientas Utilizadas:**

Se evaluaron frameworks como Tkinter, Dash y PySimpleGUI para la implementación de la interfaz.

4.5. Procesamiento y Análisis de Datos

- **Algoritmos de Clasificación:**
Se han implementado algoritmos de machine learning (ej. Random Forest) y técnicas quimiométricas en los scripts de procesamiento para clasificar muestras y detectar la presencia de metales pesados.
- **Filtrado y Normalización:**
Se desarrollaron procedimientos para filtrar ruido y normalizar los datos obtenidos en múltiples ciclos de medición.

4.6. Integración IoT y Almacenamiento en la Nube

- **Transmisión de Datos:**
Se evaluará el uso de protocolos como MQTT o HTTP para la transmisión remota de datos.
 - **Seguridad y Escalabilidad:**
Se estudiarán mecanismos de autenticación y autorizaciones para asegurar el acceso y la integridad de los datos.
-

5. Resultados y Conclusiones

- **Integración y Pruebas:**
Se ha logrado conectar el potenciómetro mediante el SDK de PalmSens, superando el problema de carga del ensamblado y estableciendo la conexión exitosa con PSTrace 5.9.
 - **Base de Datos:**
La estructura diseñada permite almacenar de forma organizada los datos de sensores, mediciones y usuarios, facilitando consultas eficientes.
 - **Interfaz de Usuario:**
Los prototipos desarrollados permiten visualizar y controlar la adquisición de datos, sentando las bases para una interfaz completa.
 - **Procesamiento de Datos:**
Los algoritmos implementados permiten identificar patrones y clasificar la presencia de metales pesados en las muestras, ofreciendo resultados prometedores.
-

6. Tareas y Próximos Pasos

Para culminar el proyecto y avanzar hacia una solución completa se recomienda:

- **Optimización de la Interfaz de Usuario:**
Desarrollar y perfeccionar una UI robusta (posiblemente basada en frameworks web como Dash o Flask) que permita el control integral del sistema.

- **Verificación en Tiempo Real:**
Realizar pruebas integrales para garantizar la adquisición y actualización automática de datos en la interfaz.
 - **Perfeccionamiento del Procesamiento de Datos:**
Ajustar y optimizar los algoritmos de clasificación y análisis para mejorar la precisión en la detección de metales pesados.
 - **Implementación IoT:**
Integrar protocolos para la transmisión remota de datos, permitiendo el monitoreo en tiempo real desde la nube.
 - **Documentación y Pruebas Finales:**
Completar la documentación técnica y realizar pruebas unitarias e integrales que aseguren la robustez y confiabilidad del sistema.
-

7. Conclusión Final

El desarrollo del sistema ha avanzado significativamente, logrando integrar la conexión con el potenciómetro mediante el SDK de PalmSens, gestionar la adquisición y el almacenamiento de datos, y desarrollar prototipos de la interfaz de usuario. La estructura de carpetas, especialmente la actualizada en la carpeta **src**, refleja con precisión la organización del proyecto, facilitando la colaboración y el mantenimiento a largo plazo.

Este informe final consolida los avances y sienta las bases para las siguientes fases: optimización de la interfaz, perfeccionamiento de los algoritmos de procesamiento y la integración completa en un entorno IoT para el monitoreo en tiempo real.