

# Portada

**Nombre del sistema:** *MultiMed Analyzer*

**Versión:** 1.0

**Autores:** *Cristian Camilo Davia Bohórquez*

*Samia Selena Rhenals Arrieta*

*Daniela Stefany Luna Popayan*

*Gyohan David Perdomo Audor*

**Curso:** Informática II - Proyecto Final

**Fecha:** Diciembre 2025

# Índice

1. Introducción
  2. Objetivo del sistema
  3. Requisitos del sistema
  4. Instalación
  5. Inicio del sistema
  6. Navegación general de la interfaz
  7. Módulos del sistema
    - 7.1 Autenticación y perfil de usuario
    - 7.2 Captura y procesamiento de imágenes
    - 7.3 Procesamiento de señales biomédicas
    - 7.4 Análisis de datos tabulares
    - 7.5 Registro de actividad y base de datos
  8. Preguntas frecuentes (FAQ)
  9. Información de contacto
- 

## 1. Introducción

*MultiMed analyzer* es una plataforma multimodal desarrollada en Python bajo la arquitectura MVC (Modelo–Vista–Controlador). El sistema permite cargar, analizar y visualizar tres tipos de datos biomédicos:

- Imágenes médicas (DICOM, NIFTI, JPG, PNG)
- Señales biomédicas (ECG/EEG en formato .mat)
- Datos tabulares (CSV)

Además, cuenta con autenticación de usuarios mediante XML, captura de imágenes vía webcam (OpenCV), visualización avanzada en PyQt y registro de actividad en base de datos SQL.

Este manual guía al usuario en el uso correcto de cada módulo del sistema.

---

## **2. Objetivo del sistema**

El objetivo principal es proporcionar una plataforma que permita:

- Procesar imágenes médicas
- Analizar señales biomédicas con FFT
- Visualizar y graficar datos tabulares
- Gestionar usuarios y perfiles
- Registrar la actividad del sistema
- Integrar todo de forma intuitiva mediante PyQt

---

## **3. Requisitos del sistema**

### **3.1 Requisitos de hardware**

- Procesador Intel/AMD mínimo 2 núcleos
- 4 GB de RAM (8 GB recomendado)
- Cámara web integrada o externa
- Al menos 500 MB de espacio en disco

### **3.2 Requisitos de software**

- Windows 10/11 o Linux
- Python 3.10 o superior
- Librerías necesarias: PyQt5, OpenCV, numpy, pydicom, nibabel, pandas, matplotlib, scipy, mysql.connector.

## 5. Inicio del sistema

Al abrir *MultiMed Analyzer*, aparece la **ventana de autenticación**.

El usuario debe ingresar:

- Nombre de usuario
- Contraseña

La verificación se realiza comparando los datos ingresados con un archivo XML interno del sistema.

Para ingresar como usuario selecciones el nombre de usuario <invitado> con contraseña <0000>.

En caso de credenciales incorrectas, aparecerá un mensaje rojo indicando “Usuario o contraseña incorrectos”.



# **6. Navegación general de la interfaz**

La interfaz principal contiene:

1. Menú superior con accesos a los módulos:
    - Inicio
    - Imágenes > Visor 3D
    - Señales
    - CVS
  2. Inicio donde se realizará Captura de Imagen y Validación
  3. Procesamiento de Imágenes médicas (DICOM, NIFTI, JPG o PNG)
  4. Procesamiento de señales Biomédicas (ECG o EEG)
  5. Datos Tabulares
- 

## **7. Módulos del sistema**

---

### **7.1 Autenticación y perfil de usuario**

Después del inicio de sesión, el usuario puede dirigirse a las secciones de:

- Inicio donde se realizará Captura de Imagen y Validación
  - Procesamiento de Imágenes médicas (DICOM, NIFTI, JPG o PNG)
  - Procesamiento de señales Biomédicas (ECG o EEG)
  - Datos Tabulares
- 

### **7.2 Procesamiento de imágenes médicas**

El sistema permite cargar archivos:

- DICOM (.dcm)
- NIFTI (.nii, .nii.gz)
- PNG/JPG

## Funciones principales

- Visualización de cortes sagital, coronal y axial
- Sliders para navegar entre los cortes
- Normalización, filtrado y umbralización (para JPG/PNG)
- Conversión a valores Hounsfield

## Guardado

El sistema guarda automáticamente:

- Metadatos del estudio en CSV
  - Imagen procesada temporalmente
- 

## 7.3 Procesamiento de señales biomédicas (.mat)

El usuario puede:

- Cargar una señal ECG/EEG
- Visualizar cada canal
- Calcular FFT por canal
- Mostrar frecuencias dominantes en una tabla
- Exportar resultados a CSV
- Generar gráfica del espectro seleccionado

Además, existe un botón para calcular la **desviación estándar** y mostrarla como **histograma**.

---

## 7.4 Análisis de datos tabulares

Tras cargar un archivo CSV, el sistema muestra:

- Una tabla con los datos cargados
- Lista desplegable para seleccionar columnas
- Gráfica individual de cada columna seleccionada

Formatos soportados:

- ✓ CSV
  - ✓ TSV
  - ✓ Archivos separados por comas/espacios
- 

## 7.5 Registro de actividad y base de datos

Cada acción realizada se registra en:

- Una base de datos de MySQL

El registro incluye:

- Usuario
- Fecha y hora
- Tipo de análisis
- Ruta de los archivos generados

El usuario puede revisar su historial dentro del módulo correspondiente.

## 8. Preguntas frecuentes

### ¿Puedo cargar varios archivos al tiempo?

No, la versión actual solo permite uno por módulo.

### ¿Puedo exportar PDF de gráficos?

Sí, en los módulos donde esté habilitado.

## 9. Información de contacto

**Equipo de desarrollo:**

*Cristian Camilo Dávila Bohórquez*

*Samia Selena Rhenals Arrieta*

*Daniela Stefany Luna Popayan*

*Gyohan David Perdomo Audor]*

**Correo:** [cristian.devia@udea.edu.co](mailto:cristian.devia@udea.edu.co)

[samia.rhenalsa@udea.edu.co](mailto:samia.rhenalsa@udea.edu.co)

[stefany.luna@udea.edu.co](mailto:stefany.luna@udea.edu.co)

[gyohand.perdomo@udea.edu.co](mailto:gyohand.perdomo@udea.edu.co)

**GitHub:** [https://github.com/crisdevia101/ProyectoFinal\\_info2.git](https://github.com/crisdevia101/ProyectoFinal_info2.git)