### **Tecnologías y Librerías**

#### **Lenguajes de programación**

* **Python 3.9**Usado para el procesamiento de imágenes, ejecución de modelos de Deep Learning y servicios REST.
* **Java**Responsable de la gestión de usuarios, manejo de archivos, consulta y almacenamiento de datos, interfaz gráfica y generación de reportes.

### **Módulo Python**

#### **Frameworks y bibliotecas**

* **TensorFlow 2.10 + Keras** Para la carga y ejecución del modelo CNN (MobileNetV2) con soporte GPU (CUDA/cuDNN).
* **ONNX Runtime 1.19** Usado para ejecutar modelos en formato ONNX, específicamente el modelo de segmentación de fondo rembg.
* **OpenCV 4.10** Procesamiento de imágenes (lectura, transformación, preprocesamiento).
* **FastAPI** Comunicación entre los módulos Python y Java a través de endpoints REST.
* **NumPy** Manejo de arreglos y operaciones numéricas para procesamiento de imágenes.
* **time** Medición de tiempos de ejecución para análisis de rendimiento.

### **Módulo Java**

#### **Responsabilidades**

* **Gestión de usuarios y roles.**
* **Manejo de archivos y envío de imágenes al backend Python.**
* **Consulta y almacenamiento de datos desde y hacia la base de datos.**
* **Interfaz gráfica desarrollada con:**
  + **JavaFX** Para el diseño y ejecución de la GUI (interfaz de usuario) de la aplicación de escritorio.
* **Generación de reportes PDF mediante:**
  + **iText**

#### **Comunicación**

* Conecta al servidor FastAPI mediante peticiones HTTP.
* Coordina el flujo completo de análisis: envío de imagen → espera de respuesta → guarda resultados → genera reportes.

### **Servicios Externos**

* **Cloudinary** Almacenamiento en la nube de imágenes originales y procesadas. Permite obtener URLs públicas para integración en reportes o base de datos.
* **Supabase (PostgreSQL)** Motor de base de datos donde se almacenan los resultados de los análisis, URLs, metadatos e historial de procesamiento.

### **Modelos y algoritmos utilizados**

* **MobileNetV2 (TensorFlow)**Modelo preentrenado CNN para clasificación eficiente de imágenes. Utilizado como núcleo del sistema de categorización.
* **rembg (ONNX Runtime)** Herramienta de eliminación automática de fondo. Usa internamente un modelo basado en U-2-Net para segmentación de objetos.

### **Requisitos mínimos del sistema**

* **GPU**: NVIDIA con Compute Capability ≥ 5.0 (Maxwell o superior), compatible con CUDA 11.2 y cuDNN 8.1.
* **CPU**: Mínimo Intel Core i3-4130 (arquitectura x86\_64 con soporte AVX).
* **RAM**: Mínimo 8 GB (recomendado 16 GB).
* **SO**: Windows 10/11 64 bits.