

Diagrama Arquitectura - Audio a Texto Pro (Whisper Edition)

Visión general de la estructura modular y las interacciones principales.

`main.py`
Punto de Entrada
Inicia la aplicación. Crea la ventana raíz Tkinter y la instancia de `AudioTranscriptorPro`. Gestiona el bucle principal y la limpieza al salir.

↓ Lanza

`gui.py`
AudioTranscriptorPro (Clase Principal GUI)

- Construye y gestiona todos los widgets de la interfaz (Tkinter).
- Mantiene el estado de la aplicación (modelo cargado, archivo seleccionado, modo depuración, etc.).
- Recibe acciones del usuario (ej: **Click Botón**).
- ****Orquesta**** las operaciones: llama a otros módulos para cargar modelo, convertir audio, iniciar transcripción, controlar playback.
- Actualiza la UI (estado, texto, resaltado) en respuesta a **callbacks** desde otros módulos/hilos.

↓ Delega tareas / Usa ↓



(`unpause`), detener (`stop`), obtener posición (`get_pos`) y descargar audio.

Usado por `gui.py` exclusivamente durante el modo Depuración.

`whisper_transcriber.py`

WhisperTranscriber (Clase)

- Lógica para cargar modelo Whisper (en `hilo`, con caché global) y transcribir audio (en `hilo`).
- Usa la librería `openai-whisper`.
- Gestiona el estado del modelo cargado.
- Reporta progreso de carga y resultado de transcripción (dict completo con texto y segmentos) a `gui.py` vía **callbacks**.

Instanciado y controlado por `gui.py`.

`utils.py`

Utilidades

- Funciones auxiliares: dibujar círculos de estado, copiar al portapapeles, exportar a TXT, comprobar entorno (`nvidia-smi` , `pytorch-cuda`).

Llamado por `gui.py` para acciones y comprobaciones.



`config.py`

Configuración

- Almacena constantes globales: versión, tipos de archivo, lista de modelos Whisper, colores UI, timeouts, etc.

Importado por la mayoría de los otros módulos para acceder a valores fijos.

↑ Comunican resultados (**Callbacks**) / Usan utilidades ↑

`gui.py`

AudioTranscriptorPro (Receptor)

- Métodos **callback** (ej: `_update_texto_whisper` , `_on_whisper_transcription_complete`) reciben datos de los hilos/módulos y actualizan de forma segura los widgets de Tkinter (áreas de texto, etiquetas de estado, botones).

Este diagrama muestra la separación de responsabilidades y el flujo general de interacción entre los componentes principales de la aplicación refactorizada.