

OCTUBRE 2030



# PROYECTO

*Chomsky Classifier*

*MA*

MAITE ALLENDE



# ¿QUÉ PROBLEMA RESUELVE?

- Verificar automáticamente a qué nivel de la Jerarquía de Chomsky pertenece una gramática.
- Ayuda pedagógica para estudiantes de lenguajes formales.
- Motivación: reduce errores manuales y enseña el razonamiento.

# OBJETIVOS DEL SISTEMA

- Clasificar gramáticas (Tipo 0–3).
- Modo explicativo paso a paso.
- Visualizaciones y reportes PDF.
- Modo tutor / comparación entre gramáticas.

# ARQUITECTURA GENERAL

- 
- Módulos principales:
  - `analizador_gramatica.py` — parseo de reglas.
  - `clasificador.py` — lógica de clasificación.
  - `explicaciones.py` — trazas explicativas.
  - `interfaz_grafica.py` — GUI (customtkinter).
  - `reporte.py`, `visualizador.py`, `utilidades.py`.
  - Entrada: texto de reglas. Salida: tipo + explicación + diagrama + PDF.
-

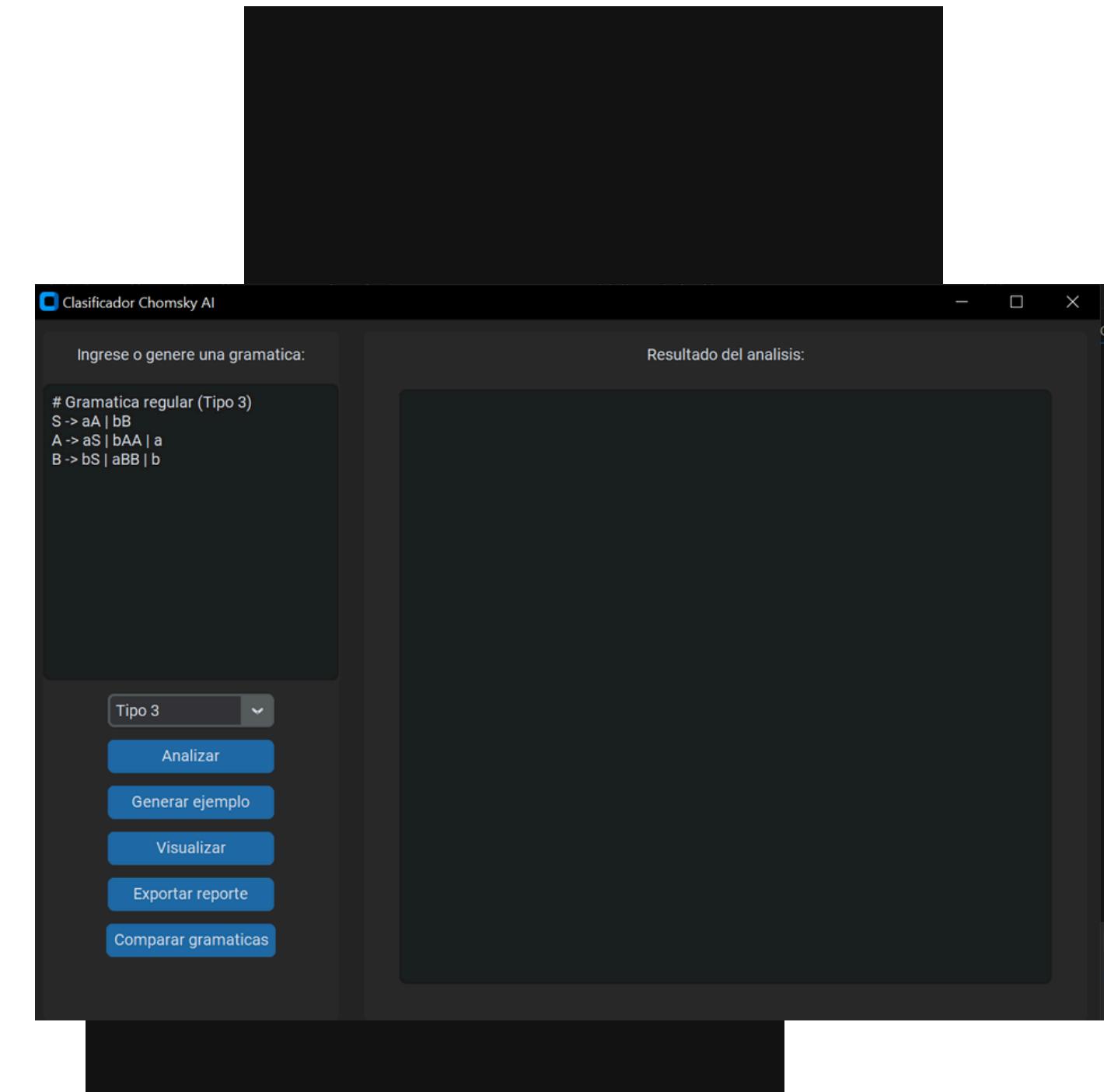
# CLASIFICACIÓN

- Comprobaciones realizadas (en orden):
  1. ¿Es regular? (lineal izquierda/derecha)
  2. ¿Es libre de contexto? (un NT en LHS)
  3. ¿Es sensible al contexto? ( $|LHS| \leq |RHS|$ )  
Si no, Tipo 0
- Resultado: tipo más restrictivo que cumple.

# MODO EXPLICATIVO

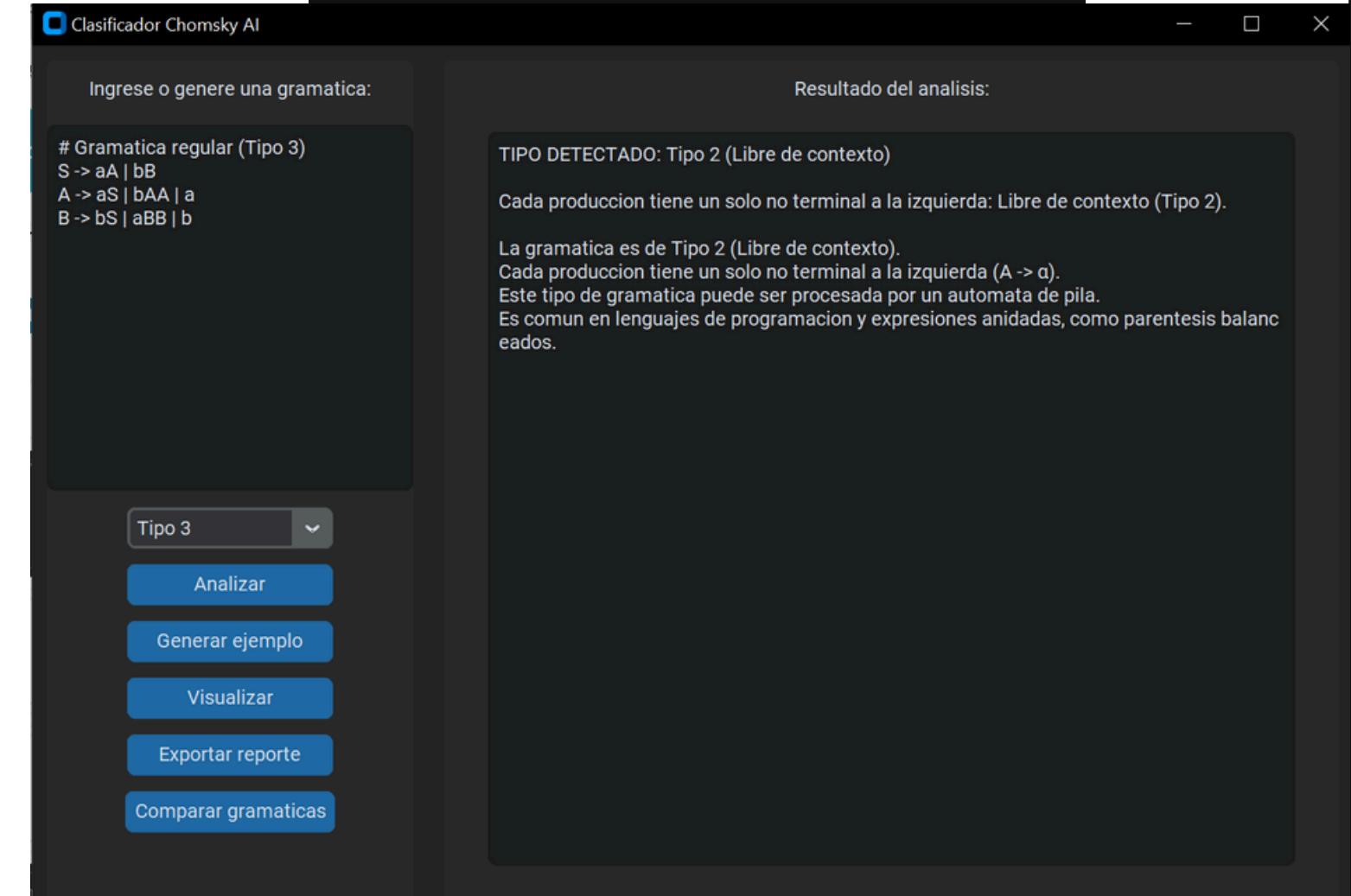
## IA

- Qué muestra la traza:
- Análisis por producción (qué condición cumple/rompe).
- Paso a paso por las reglas que determinan el tipo.
- Interfaz: botón "Siguiente / Anterior" para recorrer la traza.
- Beneficio: convierte la salida en herramienta de aprendizaje.



# COMPARACIÓN AUTOMÁTICA ENTRE GRAMÁTICAS

- Método:
- Genera cadenas desde S hasta longitud n (heurística).
- Compara conjuntos ( $G_1$  vs  $G_2$ ): cardinales, intersección y ejemplos.
- Uso: aproximación práctica (no decide equivalencia general).



# VISUALIZACIONES Y EXPORTACIÓN

- Visualización:
  - Texto ordenado (nativo).
  - Diagrama PNG (Graphviz si está instalado).
  - Alternativa: diagrama dibujado en Canvas (no requiere Graphviz).
  - Exportación:
- PDF con clasificación, explicación y diagrama embeddo.
- También un fallback TXT.