# Documento Explicativo para Sustentación: Módulo de Tramas (FSM - GUI)

Este documento explica el funcionamiento del **módulo de tramas** implementado con interfaz gráfica **Tkinter**, parte del proyecto final de Matemáticas Discretas II. Este módulo simula la transmisión binaria con validación FSM.

## 🎯 Objetivo del Módulo

1. Permitir el ingreso de tramas binarias y lista de validación.
2. Validar las tramas según condiciones de múltiplos (3 y 5).
3. Mostrar porcentaje de error y si la transmisión fue aceptada.
4. Sustituir completamente la consola por GUI para mejorar usabilidad.

## 🧱 Estructura del Módulo

### Interfaz gráfica (Tkinter)

* **Entradas gráficas:**
  + Lista de validación: una línea por valor (4 bits cada uno).
  + Tramas: una línea por trama (32 bits por línea).
* **Botón principal:**
  + “Evaluar Transmisión” → lanza la validación completa.
* **Zona de resultados:**
  + Cuadro ScrolledText donde se muestra:
    - Estado de cada trama (válida o inválida).
    - Total de errores.
    - Si la transmisión es correcta (menos de 20% de error).
* **Botón “Salir”** que cierra la ventana y vuelve al menú principal.

## ⚙️ Lógica interna

### ingresar\_lista\_validacion\_gui()

* Lee múltiples líneas.
* Cada línea debe contener exactamente 4 bits.
* Convierte cada grupo binario a entero (int(bits, 2)).

### ingresar\_tramas\_gui()

* Lee múltiples líneas.
* Cada línea debe contener exactamente 32 bits.

### validar\_tramas(tramas, lista)

* Para cada trama:
  + Toma los bits 10 a 14 (trama[9:14]).
  + Verifica:
    - Si ese valor es múltiplo de 3.
    - Si al sumarlo con el valor de la lista, es múltiplo de 5.
  + Retorna lista con resultado de cada trama.

### evaluar\_transmision\_gui()

* Cuenta cuántas tramas fueron inválidas.
* Calcula el porcentaje de error.
* Muestra si la transmisión fue aceptada o fallida.

## 🧠 Argumentos para la Sustentación

* “Diseñé la FSM para que valide transmisiones usando reglas matemáticas con bits.”
* “Usé bits del 10 al 14 para evaluar la trama, como lo exige el enunciado.”
* “Controlé la entrada para que cada valor sea binario puro (0 y 1).”
* “La interfaz reemplaza por completo la consola, facilitando el uso y presentación.”
* “El módulo calcula errores y verifica si el porcentaje es menor al 20%.”
* “Todo se muestra en pantalla sin necesidad de consola, ideal para la sustentación.”

## 📝 Evaluación según la Rúbrica

| Criterio | Cumplimiento |
| --- | --- |
| Ingreso manual por GUI | ✔️ 100% reemplazo de consola |
| Validación FSM | ✔️ Se aplican condiciones lógicas |
| Evaluación de transmisión | ✔️ Porcentaje y mensaje |
| Modularidad | ✔️ Funciones claras y separadas |
| Sustentabilidad del código | ✔️ Fácil de explicar paso a paso |
| Usabilidad en presentación final | ✔️ Interfaz clara, sin consola |

## ✅ Conclusión

Este módulo cumple con todos los requisitos: validación lógica por FSM, entrada manual bit a bit, visualización clara y todo integrado en una interfaz gráfica que puede ser fácilmente sustentada ante jurado o docente.