



# PROYECTO

# AHORRO SEMANAL

**Grupo: 1**

**Integrantes:**

- **Roberth Gancino**
- **Mateo Macas**
- **Alexis Chacón**
- **Anthony Goyes**



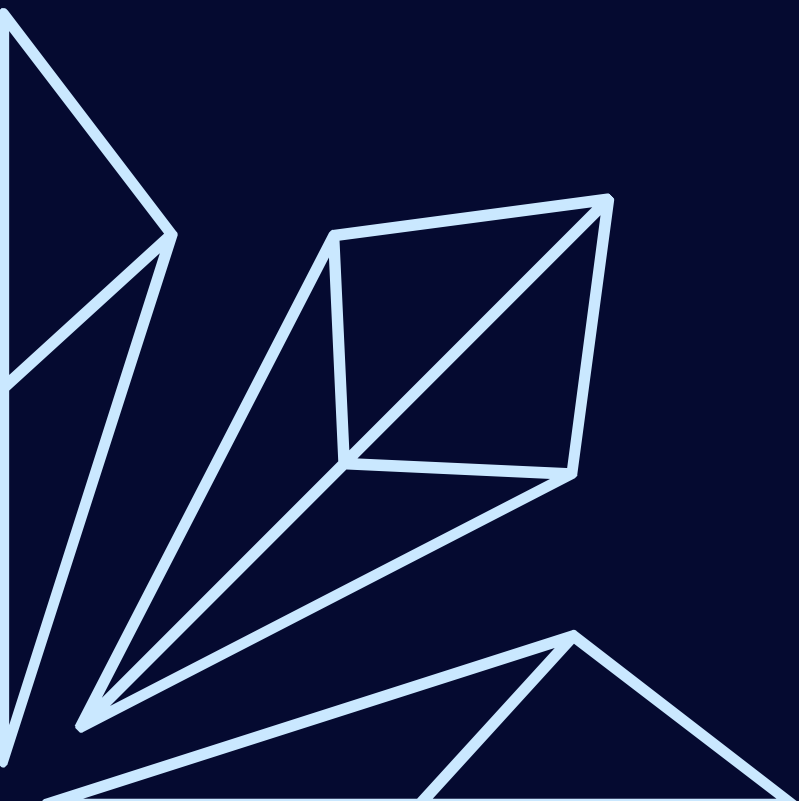
# OBJETIVO

El objetivo principal del proyecto es desarrollar una calculadora que permita a los usuarios calcular la tasa de interés efectiva para diferentes planes de ahorro (semanal, mensual, trimestral y bimestral) utilizando el método de bisección. La calculadora debe ser fácil de usar y comprender, permitiendo a los usuarios ingresar los datos relevantes de su plan de ahorro y obtener la tasa de interés efectiva correspondiente.



# INTRODUCCION

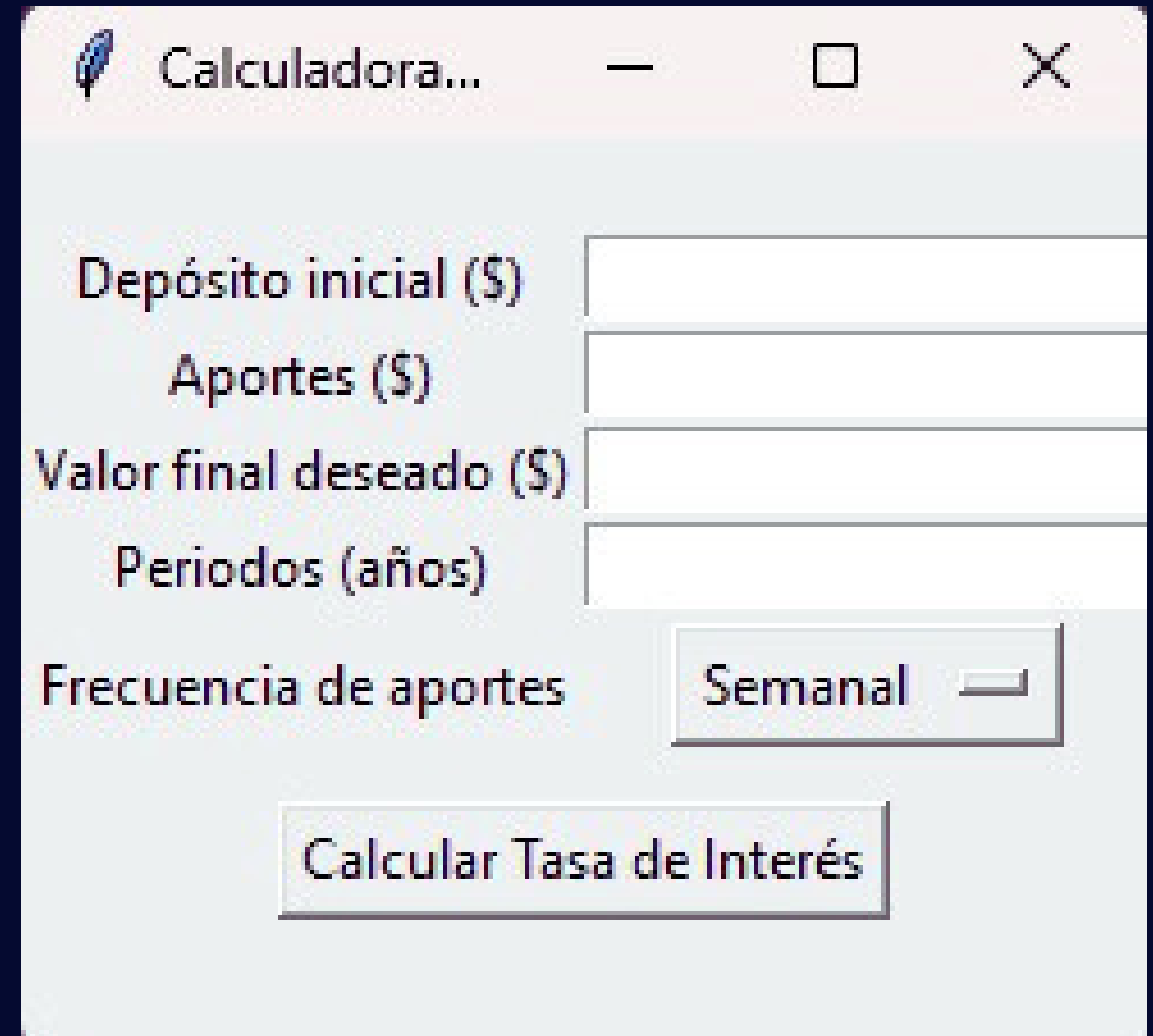
El proyecto "Ahorro Semanal" se centra en la creación de una calculadora para bancos, que solicita cálculos de tasas de interés en planes de ahorro con diferentes frecuencias de aportes. El enfoque se basa en una implementación de Python con una interfaz gráfica.



# METODOLGIA

## Desarrollo de la Calculadora:

- Entrada de Usuario: El usuario ingresa valores como depósito inicial, aportes, valor final deseado, periodos y frecuencia de aportes.
- Condiciones del Plan de Ahorro: Aporte inicial mínimo de \$50 y aportes semanales mínimos de \$5.



A screenshot of a Windows-style application window titled "Calculadora...". The window contains several input fields and a button. The fields are labeled: "Depósito inicial (\$)", "Aportes (\$)", "Valor final deseado (\$)", "Periodos (años)", and "Frecuencia de aportes". The "Frecuencia de aportes" field is a dropdown menu currently showing "Semanal". At the bottom of the window is a button labeled "Calcular Tasa de Interés".

Depósito inicial (\$)	
Aportes (\$)	
Valor final deseado (\$)	
Periodos (años)	
Frecuencia de aportes	Semanal
Calcular Tasa de Interés	

# METODOLGIA

Método Numérico: Bisección:  
Fórmula del Interés Compuesto:

$$V_f = V_o(1 + i)^n$$

Donde:

- $V_f$  = Valor Final deseado
- $V_o$  = Deposito inicial
- $i$  = Tasa de interes
- $n$  = Numero de periodos

Despeje de la Tasa de Interés: Utilización de la fórmula para encontrar  $i$  a través del método de bisección en el intervalo  $[0, 1]$ .

Razón para el Uso del Método de Bisección: Convergencia garantizada, a pesar de ser más lento que otros métodos como Newton-Raphson o La Secante

# METODOLGIA

Implementación con Python y SciPy:

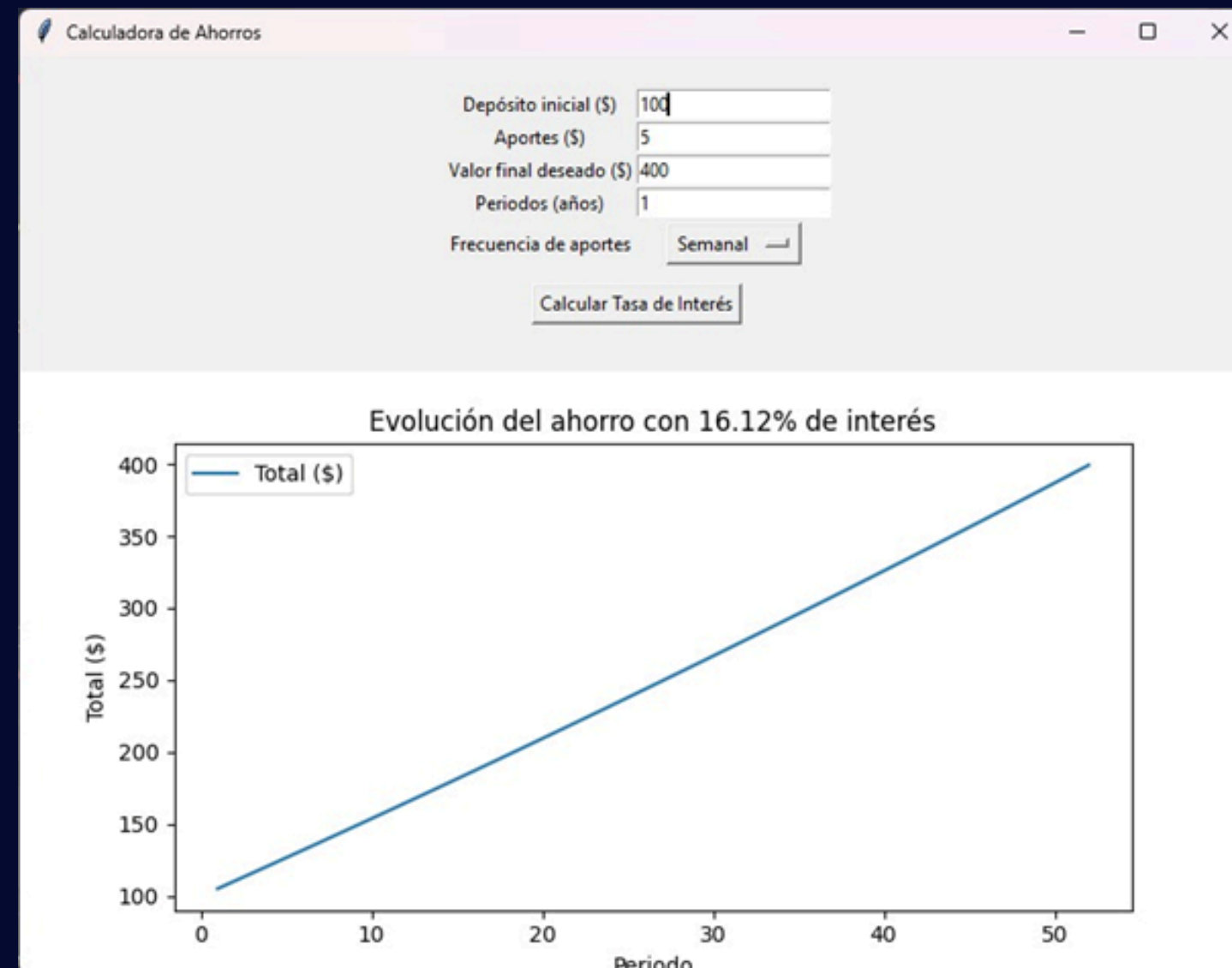
Implementación con Python y SciPy:

- Librería Utilizada: SciPy para el método de bisección.
- Diagrama de Flujo: Proceso lógico detallado para el cálculo del ahorro semanal.

# RESULTADOS

Ahorro Semanal:

Ejemplo: Depósito inicial de \$100, aportes de \$5 semanales, valor de ahorro total de \$400 en un año, resultando en una tasa de interés de 16.12%.

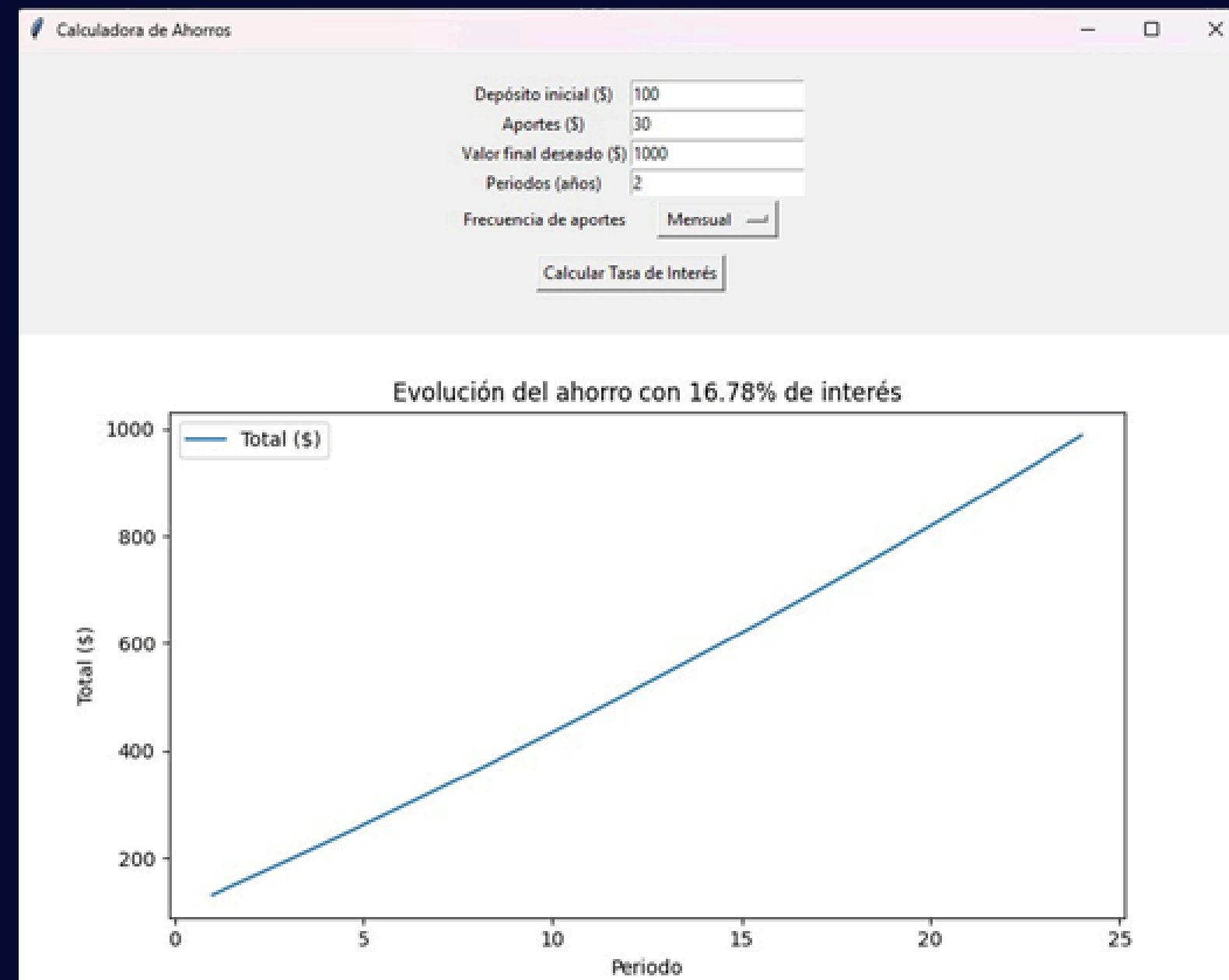


16,12% de interés

# RESULTADOS

Frecuencia de Aportes:

Mensual: Diferente tasa de interés ajustada por la frecuencia de los aportes.



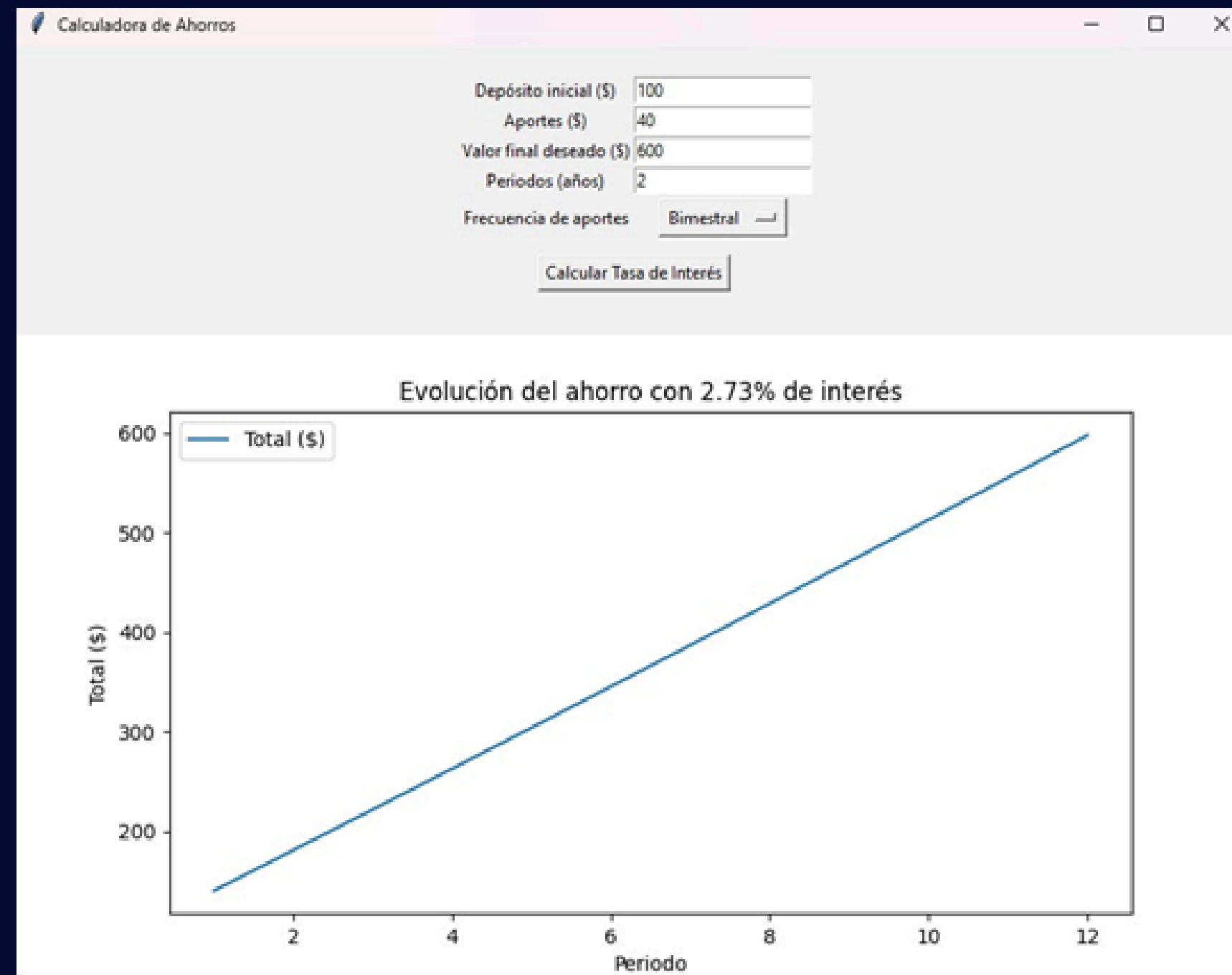
16,78% de interés



# RESULTADOS

Frecuencia de Aportes:

Bimestral: Otra variación de la tasa de interés.

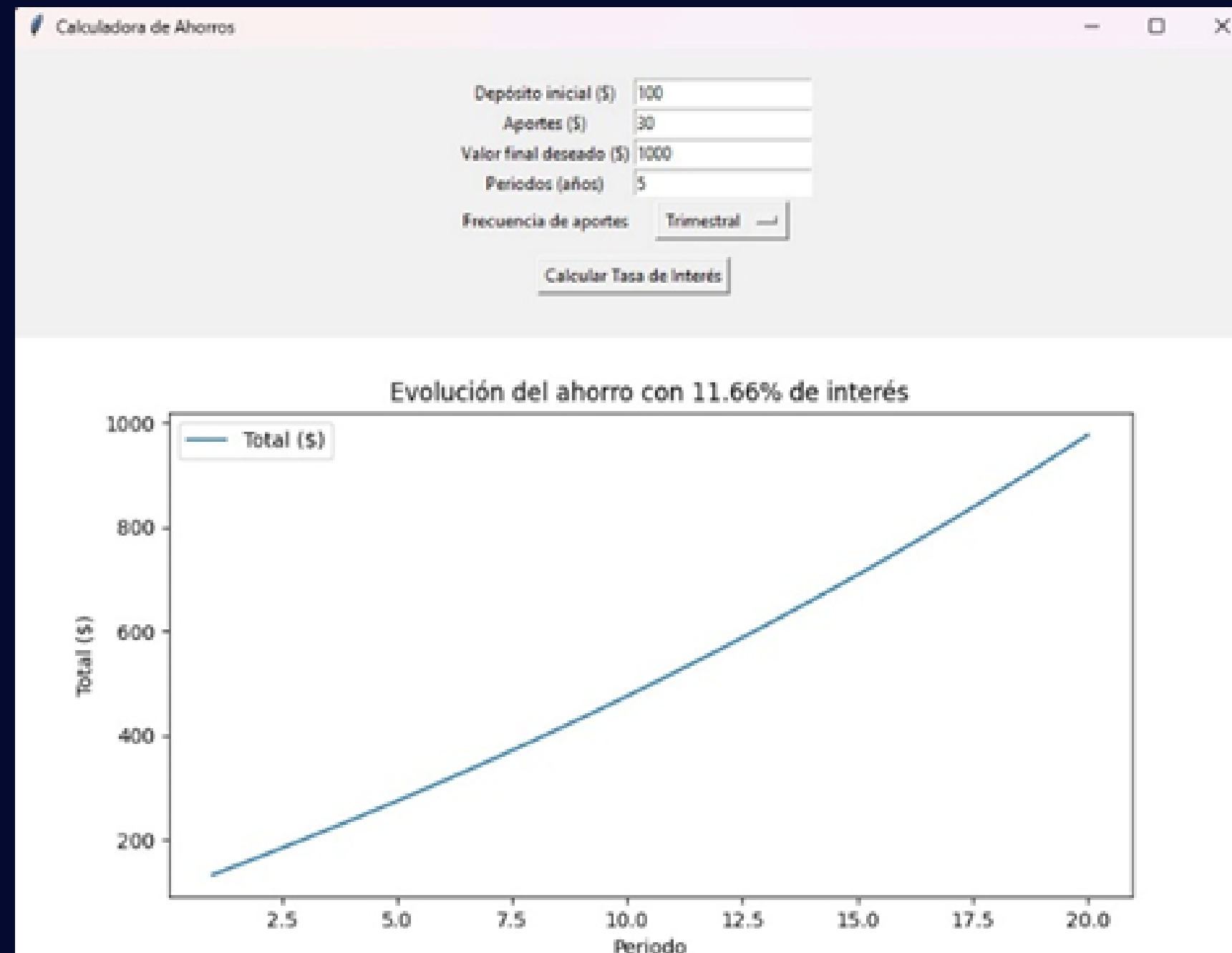


2,73% de interés

# RESULTADOS

Frecuencia de Aportes:

Trimestral: Nuevamente, una tasa de interés distinta.



11,66% de interés

# CONCLUSIONES RECOMENDACIONES

- Conclusiones: La calculadora desarrollada utilizando el método de bisección es efectiva para calcular las tasas de interés en diferentes planes de ahorro. El método garantiza convergencia y precisión en los cálculos.
- Recomendaciones: Explorar otros métodos numéricos para comparar tiempos de procesamiento y eficiencia.



GRACIAS