**Explicación de la Solución - Bryan Pérez – Dylan Torres**

**I. BOSQUEJO DEL PROBLEMA:**

Primero que todo, vamos a describir el problema planteado por medio de un diagrama de actividades.

Queremos que nuestro sistema lleve a cabo el siguiente procedimiento:

**1)** Recibir Un conjunto de entradas (S, E, r, sigma, T y M)

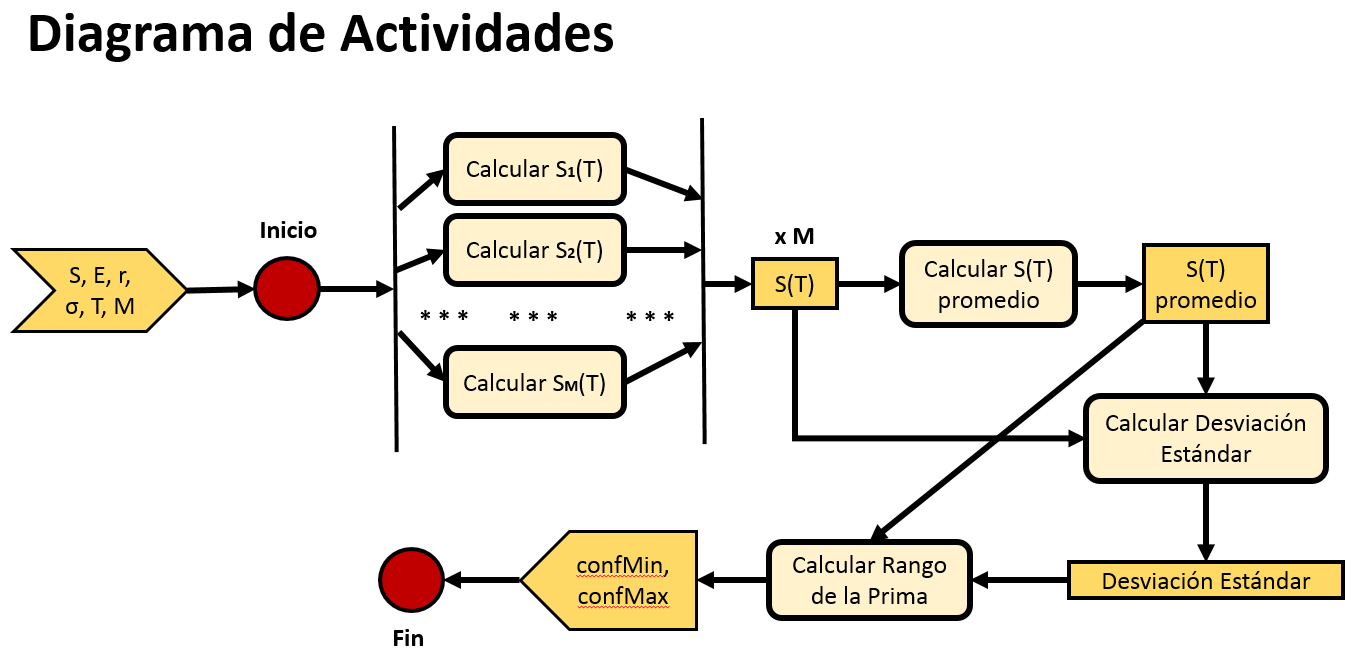
**2)** Con base en ese grupo de entradas, calcular varios valores **S(T),** de tal forma que formen una distribución normal.

**3)** Cuando los M cálculos anteriores hayan terminado, calculamos el S(T) **promedio**.

**4)** Con el conjunto de los M S(T) simulados y el S(T) promedio, generamos la **desviación** **estándar**.

**5)** Con la desviación estándar y el S(T) promedio, obtenemos el **Rango** de la **Prima**, representado en dos valores (el límite **Mínimo** (confMin) y el límite **Máximo** (confMax))

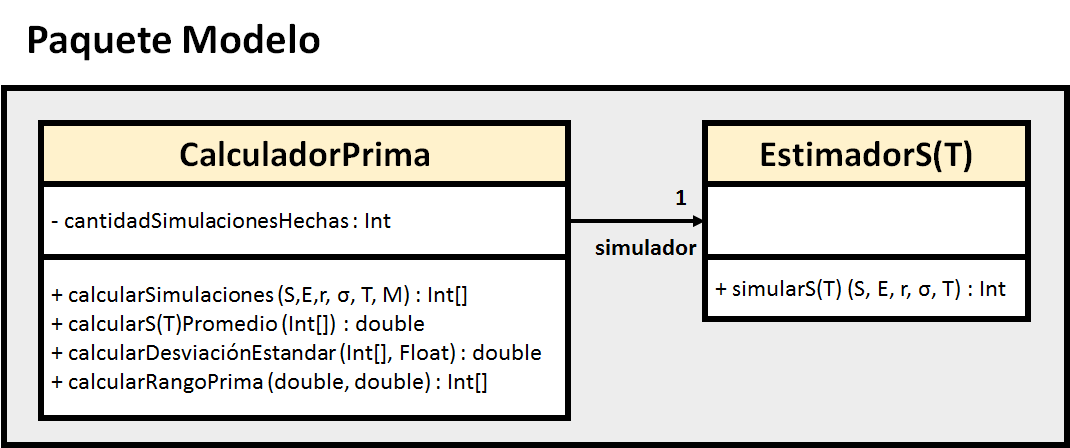
**6)** Se lanzan esos valores, o por **pantalla** o por un **archivo** .txt (Según sea lo deseado por el usuario)



**II. SOLUCION PROPUESTA:**

Ahora la incógnita pregunta es: ¿Cómo hacer esto?

Pensando en la parte lógica (sin tener en cuenta la interfaz gráfica) de la solución, se planteó lo siguiente:



Tenemos dos clases, cuyas **responsabilidades** están repartidas de la siguiente manera:

**A) EstimadorS(T):** Cumple el rol de ayudante de la clase principal. Está pensada para poder distribuir la solución en un tiempo futuro.

**1.** Dado un conjunto de parámetros, retorna un valor de S(T)

**B) CalculadorPrima:** Cumple el rol de clase principal dentro de la solución

**1.** Dirige a la clase EstimadorS(T) para poder generar diferentes valores de S(T) que cumplan con una distribución normal

**2.** Se encarga de calcular el S(T) promedio por medio de las M simulaciones generadas al azar (Random)

**3.** Calcula la Desviación Estándar de las M simulaciones

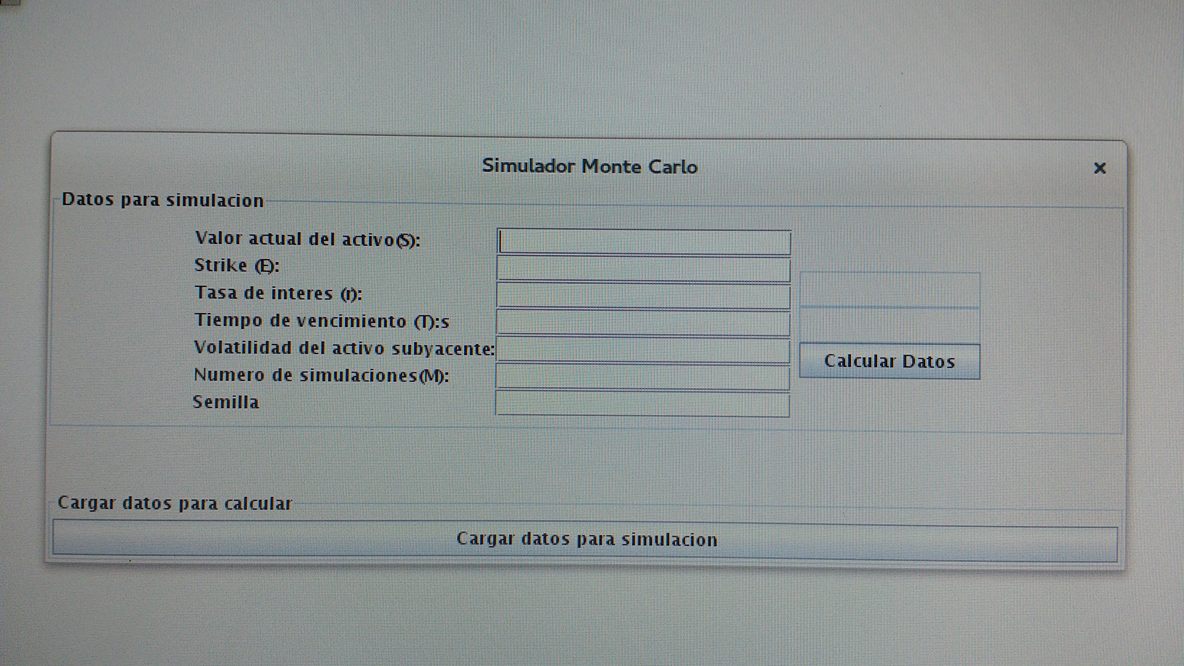
4. Obtiene el Límite Inferior y el Limite Superior del Rango de la Prima

Ahora, se pide que el sistema recolecte los datos de entrada por medio de dos maneras:

- Por componente GUI (Interfaz Gráfica)

- Por Medio de un archivo de Texto Plano.

Para satisfacer estos dos requerimientos, generamos la siguiente interfaz gráfica:

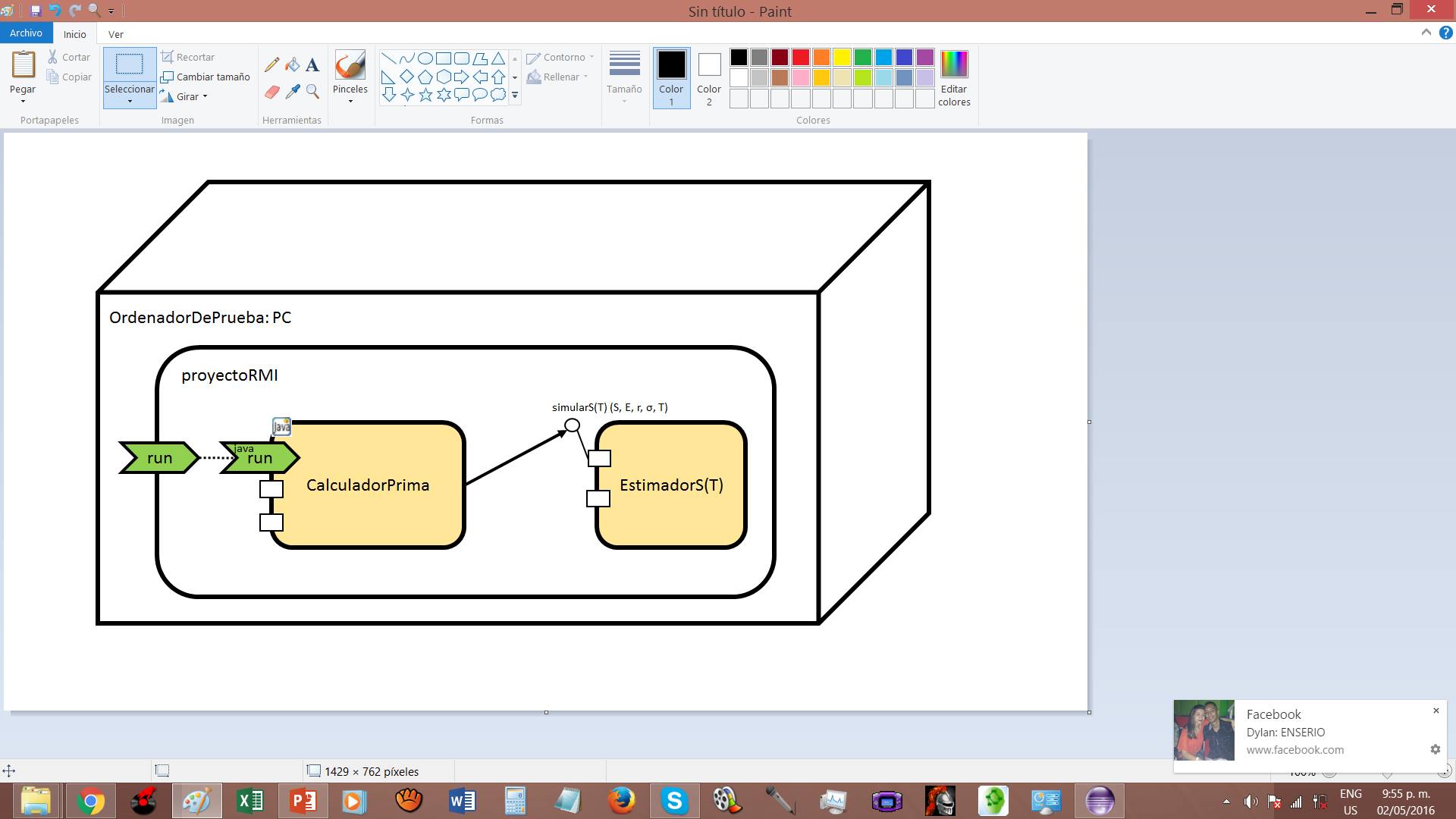


La interfaz consta de dos botones, para recoger los datos de las dos maneras explicadas anteriormente.

**Composite y Diagrama de Deployment:**

Vamos a formar un Composite completo, constituido por las clases del Modelo (Calculador y SimuladorS(T)) y las clases de la Interfaz (El JFrame y sus Paneles).

El diagrama de Deployment resultante es el siguiente:



El archivo XML que representa al .Composite anterior y que nos va a permitir conectar la solución implementado con FrASCati es el siguiente:

