**Tarea de Optimización**

**Escuela de Ingeniería de Sistemas y Computación**

**Universidad del Valle**

* Johanna Alzate 0741194
* Franco Cundar
* Cyntia Herrera

**1. MODELO**

Se busca minimizar el retraso en la entrega de la sopa que se entrega más tarde respecto con la fecha de entrega definida.

**1.1 Variables de decisión:**

1 si la sopa i se prepara justo antes que la sopa j

0 si no

**1.2 Función objetivo:**

**1.3 Restricciones:**

* Todas las sopas se entregan con tiempo de retraso:

,

* Restricción que garantiza que se haga una sola sopa al tiempo:

Para garantizar la satisfactibilidad del modelo, se asigna a M un valor lo suficientemente grande. Inicialmente se escogio M = 1000 constante, pero este resultó ser un valor pequeño para los casos en los cuales los tiempos de mezcla, entrega y alistamiento superaban ese valor. Por lo que se decidió establecer M como la sumatoria de los tiempos de entrega, tiempos de mezcla y tiempos de alistamiento de todas las sopas, ya que ese resultaría siendo el valor más grande que pudiese existir en el modelo.

* Restricciones obvias:

*| | |*

**2. DETALLES DE IMPLEMENTACIÓN**

Para el desarrollo de la tarea se utilizó la librería LpSolve integrada con Java.

El programa consta de cuatro archivos java:

* GUI.java

En esta clase esta implementada la interfaz gráfica de la aplicación que permite cargar un archivo con el formato especificado en el enunciado de la tarea y muestra los datos importantes del modelo y los resultados.

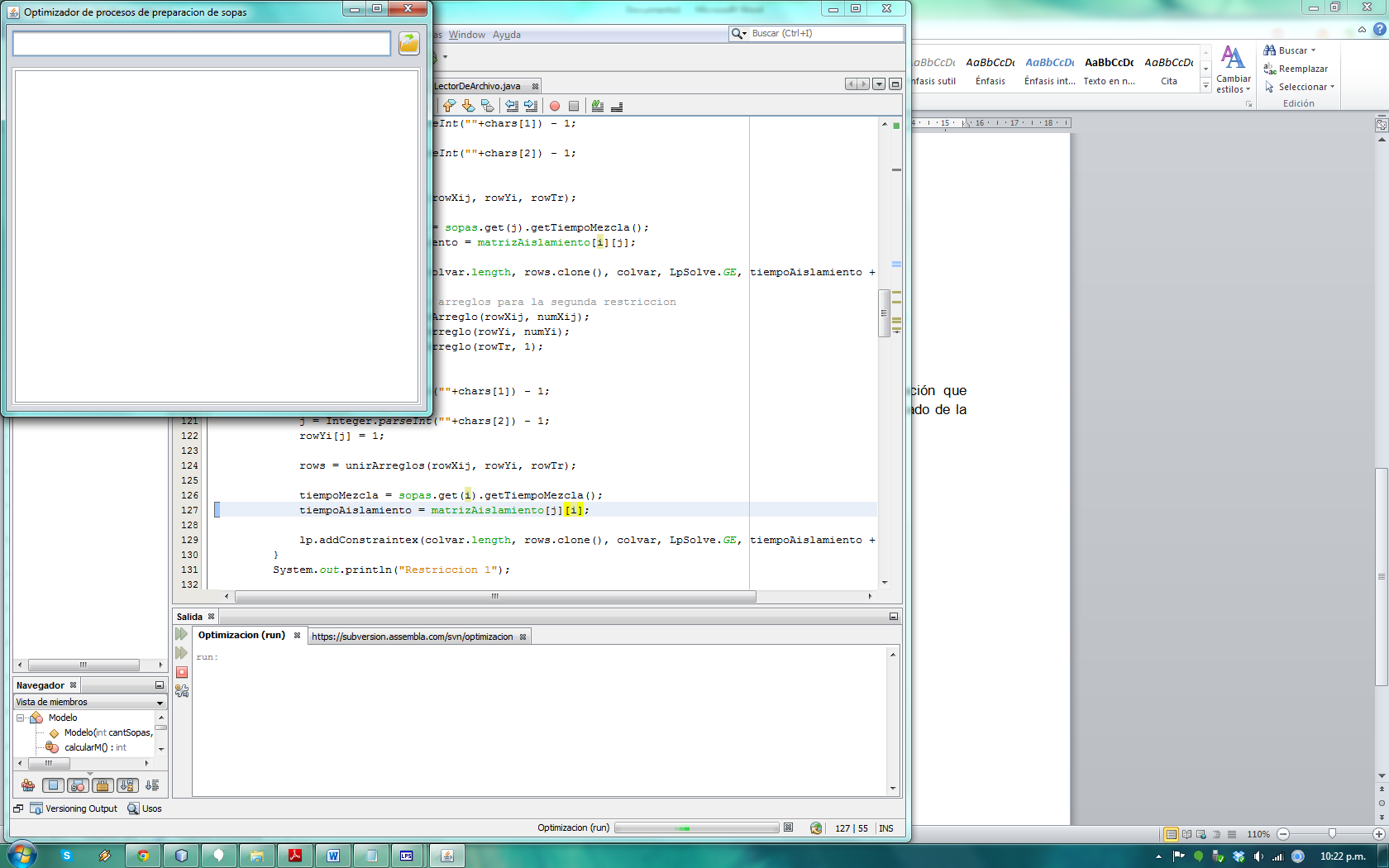


Figura 1. Interfaz gráfica

* LectorDeArchivo.java: Esta clase contiene los métodos necesarios para cargar los datos del archivo de entrada, guardando los datos en un arreglo que almacena objetos del tipo Sopa y en una matriz que representa la tabla de tiempos de alistamiento.
* Sopa.java: Esta clase contiene la representación de una sopa, almacenando su tipo, tiempo de mezcla y tiempo de entrega.
* Modelo.java: Esta clase es la encargada de recibir los datos de entrada en sus respectivas estructuras de almacenamiento y procesarlos para crear el modelo.

Contiene cuatro atributos importantes:

* + M: doublé que almacena la constante que asegura satisfactibilidad.
  + cantSopas: Integer que almacena el número de sopas en el modelo.
  + sopas: ArrayList que almacena todos los tipos de sopa.
  + matrizAlistamiento: que almacena los tiempos de alistamiento.

Además de estos atributos, contiene el método execute(), que es el encargado de crear el modelo, calcular el resultado y enviar a la GUI la información necesaria para imprimir.