JEFFRY ALBERTO ARAYA CHAVES 2022437443

MELANIE WONG 2023042240

NORMAN SAENZ 2019003307

[company name]

Documentación Externa

Proyecto 2

Índice

[Descripción de problemas 2](#_Toc156636716)

[Diseño del Programa 3](#_Toc156636717)

[Librerías Usadas 10](#_Toc156636718)

[Análisis de resultados 10](#_Toc156636719)

[Objetivos alcanzados 10](#_Toc156636720)

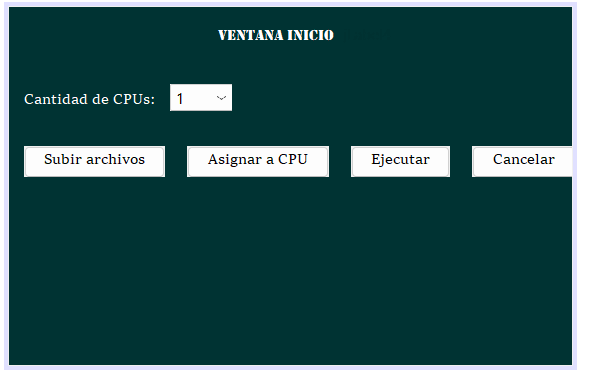
[Objetivos fallidos 11](#_Toc156636721)

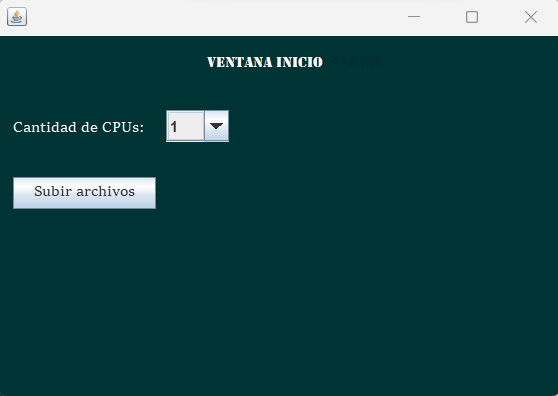
# Descripción de problemas

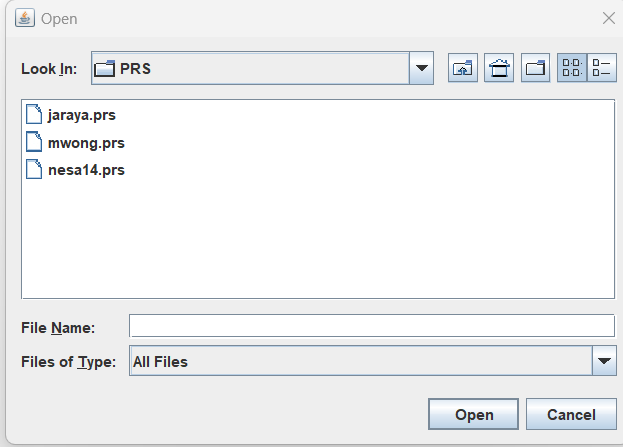
* Un problema recurrente es el de la coordinación virtual. Por el tipo de fechas y horas de desarrollo, el ponerse de acuerdo para las horas limites fue complicado.
* Hubo confusiones con el manejo de los archivos ‘.prs’ los cuales confundimos y creímos que tenían que ser ‘.json’.
* El diagrama de clases quedaba muy incómodo generarlo automáticamente desde NetBeans.

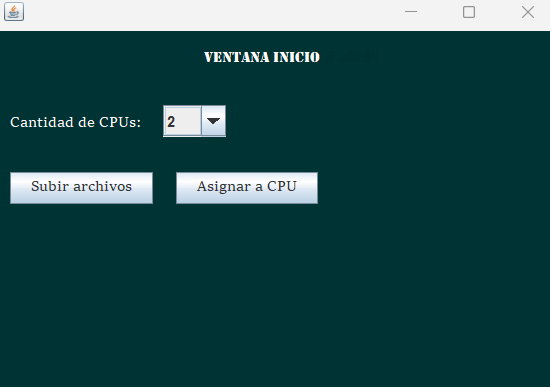
# Diseño del Programa

**VentanaInicio:**

#1 

#2 

#3 

#4 

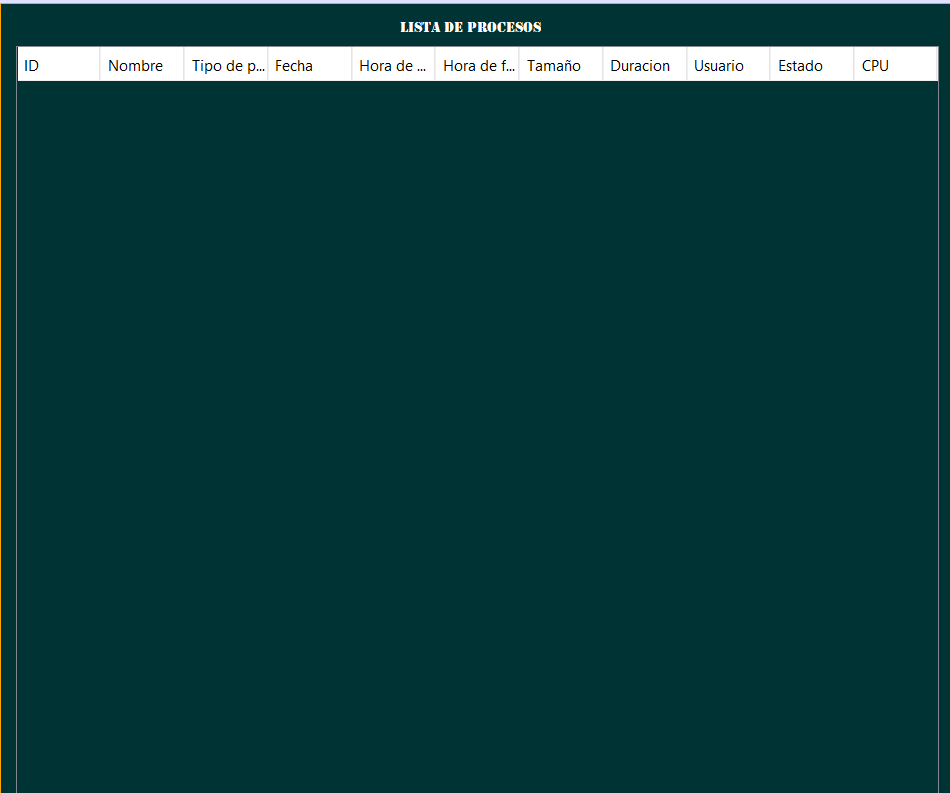
#5 

**Explicación:**

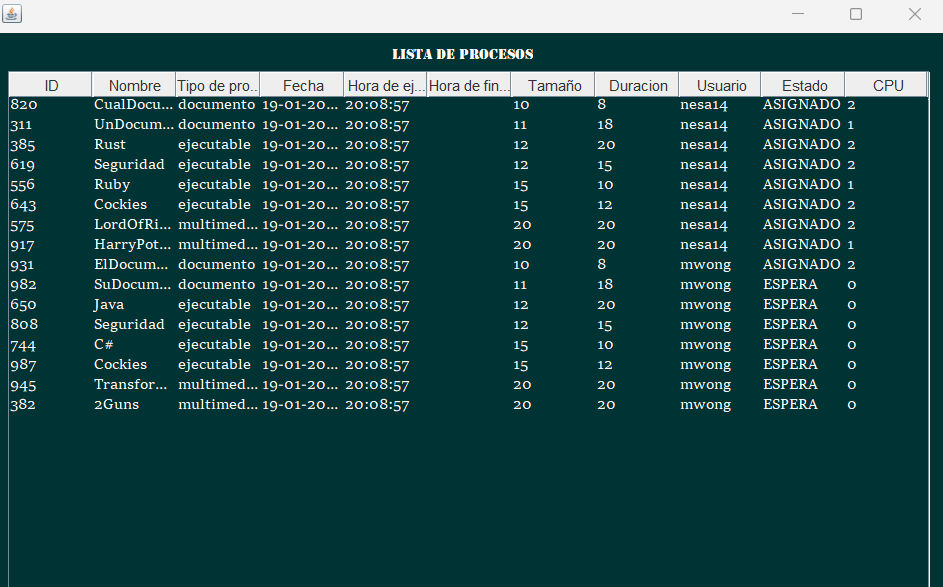
En esta clase se llama VentanaInicio , se comienza la ejecución del programa, en la primera imagen se muestra el diseño de la ventana, a la hora de dar inicio al programa, al usuario se le presentará la segunda imagen donde tendrá que presionar el botón de “Subir Archivos”, para que se le despliegue una venta, la cual es la tercera imagen, ahí escogerá el archivo o varios de ellos con extensión “prs”, después de subir los archivos se le desplegará una pantalla con la lista de los procesos que hay dentro de estos archivos, luego de esto el usuario escogerá cuantos CPUs desea utilizar, podrá escoger de 1 a 4 CPUs. Una vez que escogió la cantidad deseada, deberá seleccionar el botón de “Asignar a CPU” y con ello se le desplegarán la cantidad de ventanas de CPU dependiendo de la cantidad de CPUs que escogió, después de esto se le habilitan las opciones de “Ejecutar” o “Cancelar”, en caso de elegir “Ejecutar”, se dará inicio al programa y en caso de escoger “Cancelar”, no se comienza el programa.

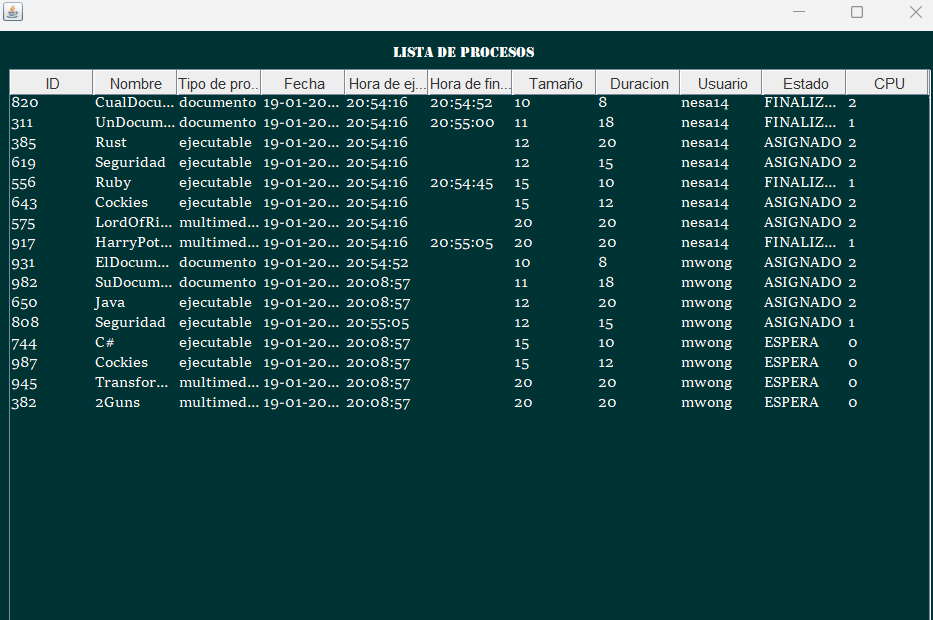
**VentanaListaProcesos:**

#1:



#2:



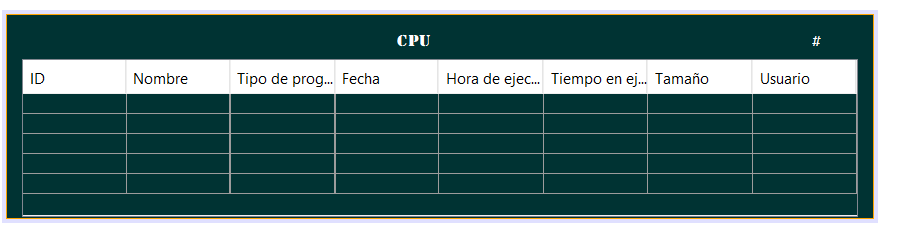
#3:  


**Explicación:**

En la primera imagen se muestra el diseño de la VentanaListaProcesos, esta clase se encarga de mostrar los procesos dentro de los archivos PRS, en la segunda se muestra los archivos ya subidos desde la **VentanaInicio,** ya por último en la tercera imagen, se muestran los procesos en ejecución, se puede notar ya que en algunos procesos se cambió el estado de “Asignado” a “Finalizado.

**VentanaCPU:**

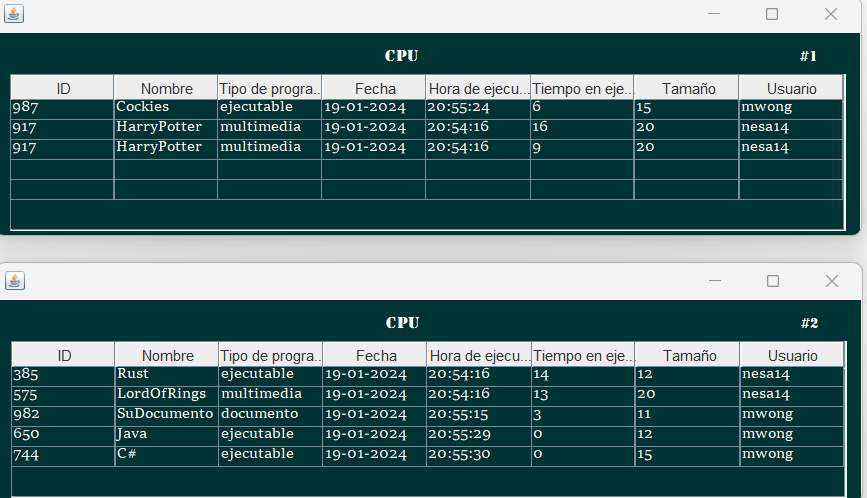
#1:



#2:



#3



**Explicación:**

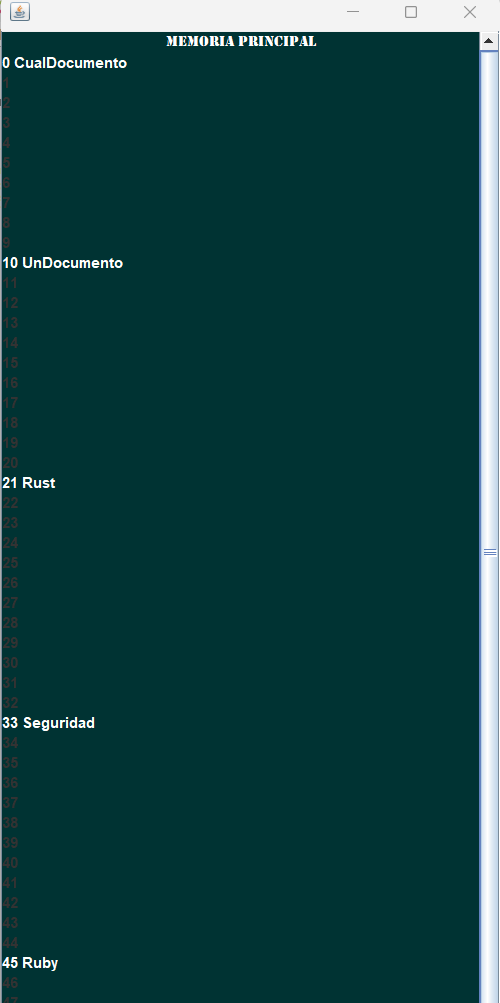
En la primera imagen se muestra el diseño de la clase CPU, tomando referencia a la clase VentanaInicio, dependiendo de los CPUs escogidos en esa clase, se mostrarán la misma cantidad de ventanas de CPUs, en este caso solo dos y van numeradas como se muestra en la segunda imagen, a la hora de dar ejecución al programa aparecen los procesos dentro de estas ventanas de CPUs, como se muestra en la tercera imagen

**VentanaRAM:**

#1:



#2:

  
**Explicación:**  
 Esta clase está encargada mostrar los procesos dentro de la memoria, respetando el tamaño de esta, en la primera imagen se muestra el diseño de la ventana, en la segunda imagen ya se muestran los procesos asignados dentro de la memoria, el tamaño de cada proceso se puede notar porque están marcados de blanco, por ejemplo, el proceso llamado “CualDocumento” tiene un tamaño de 10, ya que toma de 0 a 10 espacios de memoria. En caso de que la memoria esté llena, cuando finaliza un proceso se añade el siguiente que esté en ejecución.

# Librerías Usadas

No usamos librerías externas

# Análisis de resultados

## Objetivos alcanzados

* Un ejemplo es la mejora progresiva de la coordinación del grupo.
* Manejo de hilos.
* Mejora en UI.

## Objetivos fallidos

* No logramos cerrar la aplicación de manera óptima por falta de tiempo.
* A veces la pestaña de VentanaRam parpadea constantemente si se mueve de posición o cambia algún estado.

# Video explicativo