



## Proyecto final de Sistemas Operativos

El objetivo de este proyecto, es el desarrollo de una aplicación de monitoreo de un sistema operativo para ser ejecutado en la consola, junto su respectiva aplicación gráfica (utilizando cualquier librería de interfaz gráfica multiplataforma como GTK, Qt, Fltk, etc) para facilitar el uso por parte de las personas sin conocimientos de la consola.

### ***PARTE I. Desarrollo de la aplicación de consola***

La aplicación de consola simplemente requiere el paso de **un argumento** para indicar la tarea a realizar, este parámetro, es escrito al **momento de ejecutar la aplicación** y en **ningún momento se debe ser solicitado**, por ejemplo:

```
$ miProyectoFinal -kernel-version  
3.6.1-1-ARCH
```

Aquí el programa recibe el parámetro **-kernel-version** e imprime la versión de Linux (en este caso **3.6.1-1-ARCH**).

La lista de parámetros que debe estar disponible en la aplicación y su respectiva salida es la siguiente:

1. **--kernel-version:** Muestra la versión del kernel linux actual.
2. **--running-processes:** Número de procesos ejecutándose en el sistema.
3. **--mem-total:** Total de memoria RAM.
4. **--mem-total-free:** Total de memoria RAM libre.

5. **--mem-swap:** Total de memoria SWAP.
6. **--mem-swap-free:** Total de memoria SWAP libre en el sistema.
7. **--disk-list:** Lista los discos existentes en el sistema.
8. **--partitions-list:** Lista las particiones existentes en todas las unidades del sistema.
9. **--net-list:** Lista las interfaces de red.
10. **--net-list-ip:** Lista las interfaces de red junto a su IP (si tiene asignada).
11. **--disk-space:** Muestra el espacio usado en los discos montados.
12. **--current-user:** Muestra el nombre usuario con el cual fue ejecutado el programa.
13. **--date-time:** Muestra la hora y fecha actual del sistema (formato y hora de Venezuela).
14. **--uptime:** Muestra el tiempo que ha estado encendida la computadora.

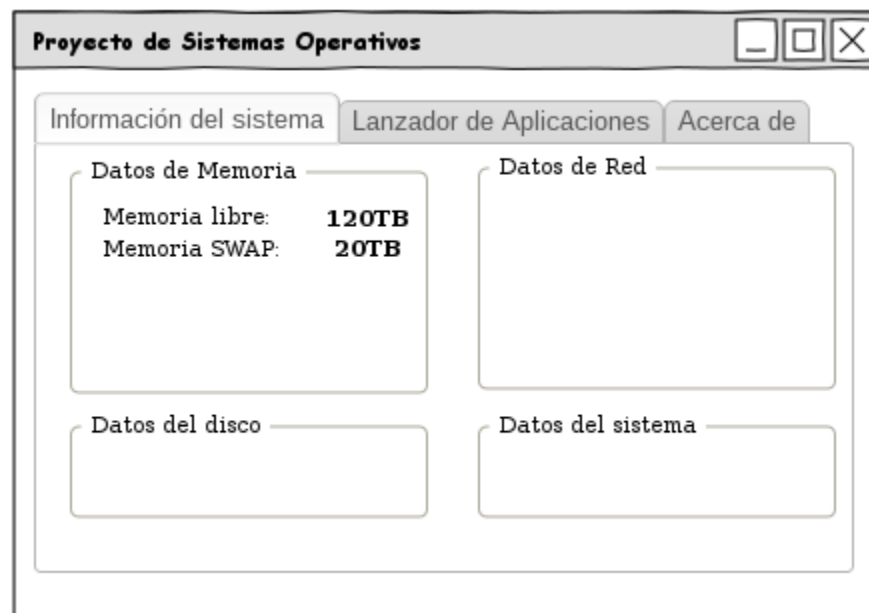
### Limitaciones:

- La salida solicitada debe ser simplemente el valor(es) solicitado(s) **no agregue texto no pedido o use formatos personalizados.**
- Para el caso de las unidades de memoria principal y secundaria la salida debe estar en GB.
- Puede realizar la aplicación en lenguaje C/C++ o en una secuencia de comandos (script).
- El parámetro debe pasarse al momento de ejecutar la aplicación, **no se puede solicitar en tiempo de ejecución.**
- Debe existir compatibilidad para al menos 2 sistemas operativos, se recomienda Windows y Linux, pero puede agregar soporte para cualquier otro, si lo desea, puede usar el mismo código fuente y utilizar directivas del compilador para generar, para mayor facilidad de desarrollo, se recomiendan versiones independientes para cada sistema.

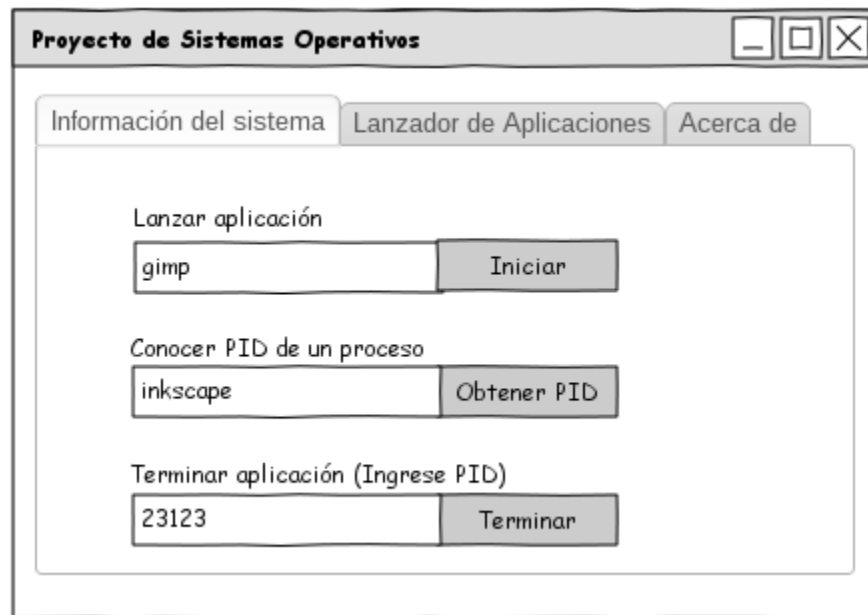
## PARTE II. Desarrollo de una interfaz gráfica

Se debe elaborar una interfaz gráfica para facilitar el uso de la aplicación, para ahorrar tiempo y esfuerzo, cada vez que se requiera la información del sistema **se ejecutará la aplicación de la Parte I** y se obtiene su salida. Es decir, la aplicación gráfica **será un lanzador de comandos de la aplicación de la primera parte.**

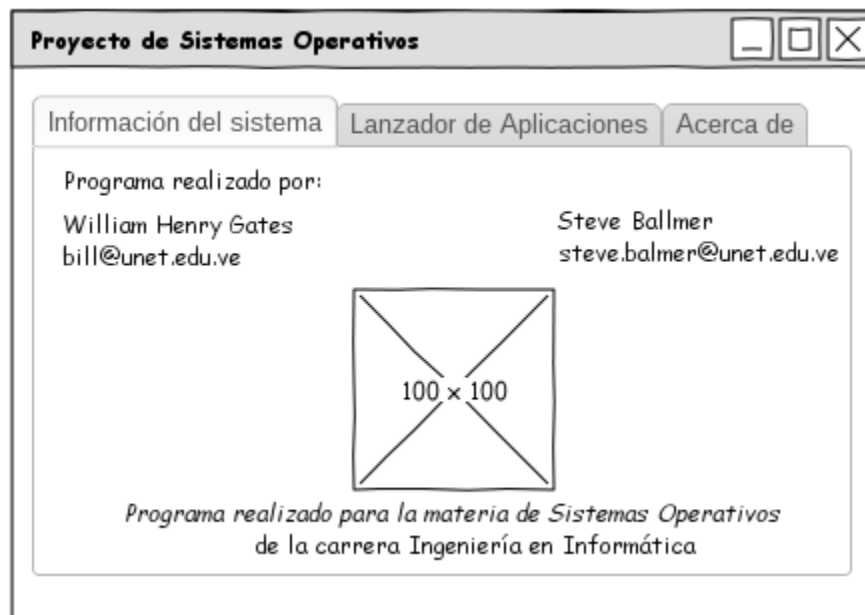
Este programa estará compuesto de 3 secciones, la primera se denomina **“Información del sistema”** donde se mostrará cada una de la información que el programa de consola puede mostrar, para facilitar su lectura, organice la información de manera ordenada, por ejemplo crear una sección dentro de la ventana llamada Datos de Memoria y mostrar debajo de ella los datos de memoria total, libre, intercambio, etc. Toda la información mostrada aquí debe actualizarse cada 5 segundos para ver siempre la información mas actualizada y recuerde mostrar las unidades correctas para indicar el valor correcto al usuario.



La segunda sección llamada **“Lanzador de aplicaciones”** permite lanzar y cerrar aplicaciones, todo mediante su PID. En esta parte es opcional el uso programa de la parte anterior, lo importante es que permita lanzar aplicaciones sin bloquear el programa gráfico, permita obtener el PID de un programa en específico con el fin de utilizarlo para matarlo (a través del nombre del programa), si existen varias instancias, tome solo el PID de una de ellas. Utilice la siguiente imagen para obtener una idea del requisito solicitado.



La última sección será llamada **“Acerca de los autores”** donde se escribirá el nombre de los integrantes, junto a su cédula y correo electrónico de la universidad. A continuación se debe mostrar el logotipo de la UNET centrado respecto a la ventana y con el texto *“Programa realizado para la materia de Sistemas Operativos de la carrera Ingeniería en Informática”*. Un diseño de ésta ventana puede observarse en la siguiente captura de pantalla.



### Características a tomar en cuenta:

- La aplicación servirá de monitor de varios aspectos del sistema operativo, por ello, cada 5 segundos se debe refrescar toda la información mostrada en la sección actual. Puede utilizar hebras / hilos o cualquier método de ejecución por tiempo.
- La aplicación gráfica **debe ser independiente**, no debe realizar llamadas al sistema ni ejecutar comandos de la shell de Linux para obtener la información, **sino tomar la información a través de la salida del programa de consola** (realizado en la parte I).
- Es libre usar cualquier disposición y formato para mostrar la información, se evaluará organización, simplicidad y presentación agradable de los datos en la ventana.
- Puede utilizar cualquier librería gráfica de C/C++.
- Al momento de mostrar información, tenga como prioridad la facilidad de lectura por parte del usuario: utilice un tamaño de letra adecuada a la lectura, muestre correctamente las unidades de cada información

mostrada (si tienen) y utilice decimales cuando sea necesario para evitar pérdida de información.

- No se permite el uso de lenguajes interpretados, pues no interactúan con el sistema operativo directamente, ejemplo: Java, Python, Ruby, Javascript, PHP, etc.
- La cantidad máxima de integrantes es de 3 personas, se sugiere dividir el trabajo entre el desarrollo de la interfaz gráfica, aplicación de consola para Windows y Linux respectivamente.
- Al momento de la entrega deben estar los desarrolladores del proyecto para comentar sobre el trabajo realizado.