

# **Ejercicio 1 Y 3**

Rubén Hita Morales

14/09/2022

Versión 1.0

IES Francisco de Quevedo 2DAM Desarrollo de Interfaces

## Índice

Introducción.....	3
Requisitos que debería cumplir.....	3
Dibujo de la apariencia de la aplicación.....	4
Prototipo.....	4
Diseño Final:.....	5
Proceso de ejecución y uso.....	6
Evaluación de la usabilidad.....	7
Referencias.....	8
Conclusiones.....	8
Mi valoración personal.....	9

# Introducción

El objetivo del trabajo es describir como debería hacerse y que debería tener una interfaz gráfica para la ejecución de comandos.

Dividiéndolo en tres partes:

1. Requisitos que debería cumplir: En esta parte explico que requisitos tendría que cumplir la interfaz para que sea lo mas sencillo posible para la persona que lo vaya a usar.
2. Dibujo de la apariencia de la aplicación: Un dibujo de como se vería la interfaz.
3. Proceso de ejecución y uso: El proceso para que el usuario interactúe con la interfaz de manera correcta.
4. Evaluación y usabilidad: Aquí muestro el proceso necesario para evaluar la la aplicación de manera correcta.

## Requisitos que debería cumplir

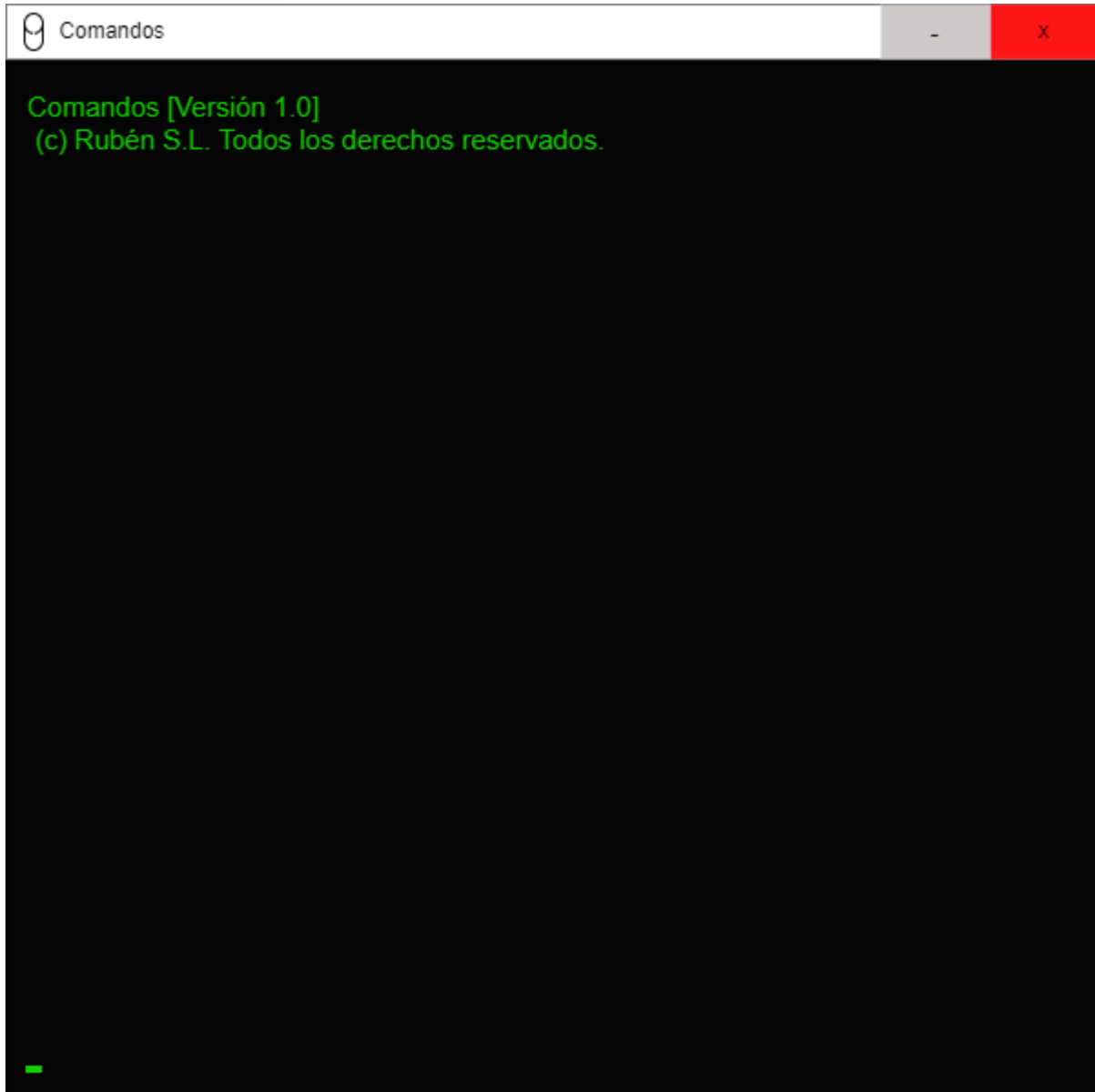
Teniendo en cuenta que la función principal de esta interfaz es simplificar la comunicación entre una máquina o un sistema operativo y un usuario y que su uso sea lo mas sencillo posible para el tendríamos que tener en cuenta estos requisitos:

- Una interfaz fácil de ver, utilizando una gama de colores cuyas combinaciones sean intuitivas para el usuario. Por ejemplo el fondo negro y el texto en verde o blanco hace que se vea mucho mejor.
- Una interfaz que responda al usuario cuando cada vez que se introduzca un comando indicando lo que se ha ejecutado. En caso de que ocurra cualquier error (un comando mal escrito), la interfaz respondería al usuario indicando cual es el error.
- Una interfaz en la que hayan indicaciones claras de donde y como el usuario debe introducir los comandos.
- En la interfaz se va a poder usar tanto el teclado como el ratón. El teclado para escribir los comandos y el ratón para seleccionar texto.
- Facilitar y predecir las acciones más comunes del usuario: La propia interfaz debería de facilitar las acciones al usuario prediciendo el comando que va a escribir.
- La curva de aprendizaje es acelerada y es fácil recordar su funcionamiento: La interfaz debería estar hecha de manera que el usuario se adecuó fácilmente a su funcionamiento.

# Dibujo de la apariencia de la aplicación

## Prototipo

Este es un prototipo de como se vería la interfaz de una manera básica para solo comprobar el funcionamiento



Primero haría este prototipo solo para probar si la interfaz funciona.

- Se puede ver que abajo a la izquierda hay un rectángulo verde en el que se podrán escribir los comandos y si pulsas enter se ejecutan. Todavía no está puesto el path porque al principio solo comprobaría si se pueden ejecutar bien los comandos.
- Las letras verdes hacen contraste con el fondo negro para facilitar la vista de las mismas, dentro de la interfaz se podría escoger distintos colores para el texto cumpliendo una gama de colores que siempre hagan contraste con el fondo.

## Diseño Final:

Este sería el diseño final de la interfaz

```
C:\Users\2dam> python3 trenecito
```

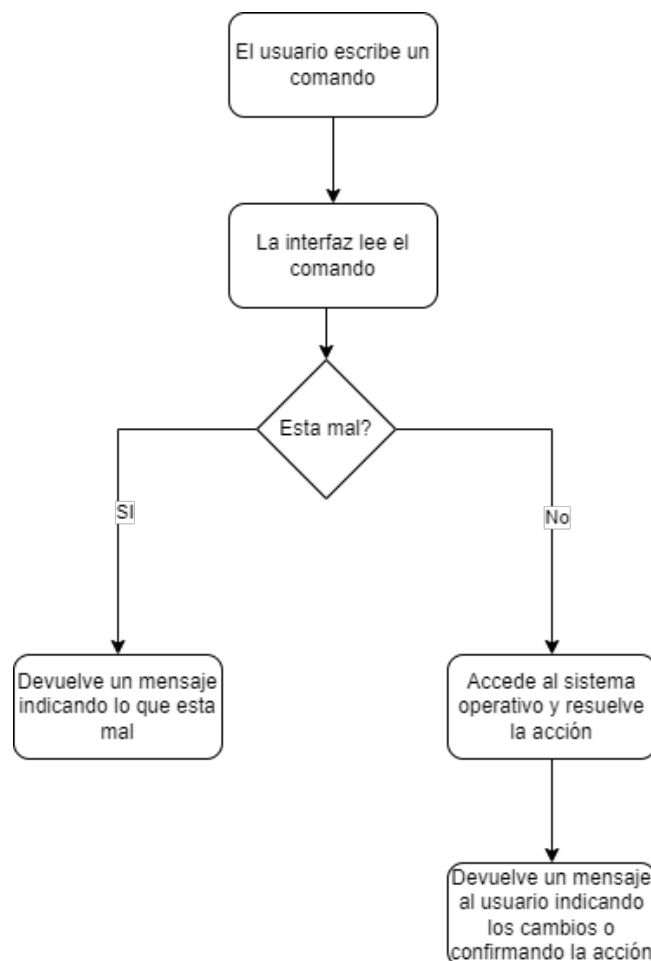
The ASCII art depicts a steam locomotive pulling several passenger cars. The locomotive has a large smokestack emitting smoke, a boiler with rivet details, and a multi-spoked flywheel. It is followed by three rectangular passenger cars with windows and doors. The entire scene is composed of various symbols like underscores, pipes, hyphens, equals signs, and letters.

- Arriba a la derecha hay tres botones: Minimizar, maximizar y cerrar, la de cerrar esta en rojo para que se sepa que es la de cerrar de manera intuitiva. Esto indica claramente al usuario que es cada cosa
- Abajo a la izquierda esta la parte donde se escribe indicada con un rectángulo que siempre esta a la derecha de lo escrito. Ejemplo: `C:\Users\2dam> cd ..`. Siempre se debe escribir ahí y si el usuario ya ha usado mas de una vez el programa debería de ser fácil de recordar.

## Proceso de ejecución y uso

En principio esta interfaz se trata de la administración del sistema a través de la línea de comandos, así que el proceso de ejecución se basa en 4 pasos:

1. El usuario escribe un comando y pulsa “enter” para ejecutarlo.
  - a) El usuario debe escribir el comando con la sintaxis correspondiente: comando [parámetro]. Ej. cd Desktop/2DAM.
2. La interfaz resuelve el comando y ejecuta una acción.
  - a) En caso de que el comando sea erróneo devuelve un mensaje indicando cual es el error para que el usuario pueda resolverlo.
3. En caso de que el comando este bien escrito la interfaz ejecuta el comando.
4. La interfaz después de ejecutar el comando devuelve un mensaje confirmando la acción o los cambios provocados en el equipo.
5. Finalmente la interfaz vuelve al estado inicial en el que el usuario podrá escribir otro comando.



# Evaluación de la usabilidad

Aquí explico el plan de evaluación de la usabilidad al analizar la interfaz.

1. **Definición de los usuarios:** El tipo de usuario que utiliza esta interfaz es un usuario con conocimientos informáticos intermedios que necesita utilizar los comandos para ejecutar procesos un poco mas complejos en su ordenador.
2. **Definición de las tareas a evaluar:** Las tareas que se llevan a cabo en esta interfaz son la comunicación entre el usuario y la maquina mediante los comandos ejecutados por la interfaz.
3. **Definición de las condiciones de evaluación:**
  1. Duración de la observación: El tiempo de análisis será el suficiente y necesario hasta comprobar que todas las tareas funcionan perfectamente y el usuario podría utilizar la interfaz sin ningún problema
  2. Estados inicial y final al cual se quiere llegar de la tareas a evaluar: Al principio se evaluaría con un prototipo de una interfaz no muy avanzada, solo para comprobar el correcto funcionamiento, cuando se haya terminado la observación y se compruebe que todo funciona bien, se probaría con una interfaz mas avanzada, que sería la interfaz final para el cliente.
  3. Trabajar sobre un sistema real o simulado (laboratorio): Las pruebas se harían con una interfaz con el mismo funcionamiento que la interfaz final para el usuario, pero menos avanzada para hacer todas las pruebas mucho mas rápido.
  4. Establecer la capacidad de intervención del observador: El observador debe intervenir en el funcionamiento lo suficiente para comprobar que ciertas cosas funcionan y arreglar todos los fallos.
4. **Establecer la metodologías:**
  1. Un mismo usuario para distintas aplicaciones: La aplicación se podrá abrir en varias ventanas y se podrán usar a la vez (ejecutar procesos al mismo tiempo). Pero solo podrán ser usadas por un mismo usuario
  2. Observación del usuario mientras realiza la tarea: Mientras el comando se ejecuta la interfaz deberá mostrar al usuario en todo momento el proceso de la ejecución, para que el usuario pueda ver lo que esta pasando.
  3. Cuestionarios y entrevistas: El usuario podrá hacer una valoración de la interfaz, indicando que se puede mejorar y errores que se puedan solucionar.
5. **Valorar aspectos éticos- sociales:**
  1. El usuario no debe sentir presión indebida: Dejar que el usuario libremente lo pruebe, lo use y pueda dar su propia opinión.
  2. El usuario tiene el derecho de saber de que se trata el estudio: Explicar al usuario de que se trata todo antes de empezar a hacer pruebas.

3. No confundir al usuario: Ser claro sobre el funcionamiento para que el usuario no se confunda.
  4. Abortar el estudio si el usuario lo pide: Si el usuario no quiere seguir con las pruebas que pare cuando quiera.
- 6. Establecer que se va a medir cuantitativamente (observación objetiva):**
1. Tiempo en completar la tarea: Que la interfaz este optimizada de tal manera que tarde el menor tiempo posible en ejecutar un comando
  2. Número de tareas logradas en un tiempo dado: Que en el menor tiempo posible pueda resolver el mayor numero de tareas en ese mismo tiempo
  3. Tasa de errores: Procurar que la tasa de errores sea la mínima posible.
  4. Tiempo de recuperación de errores: Que los errores se puedan solucionar en el menor tiempo posible.
- 7. Presupuesto:**
1. Usuarios no productivos mientras dure la prueba (dejan de hacer su trabajo para hacer las pruebas): Se pararía la producción para poder hacer las pruebas, eso conllevaría un parón para poder seguir desarrollando.
  2. Personal necesario para recabar y tratar los datos: Se necesitará personal para poder hacer todas las pruebas, lo cual conllevará un gasto.
  3. Soporte para hacer las modificaciones que salgan: En caso de que haya que modificar algo, solucionar ese problema es otro gasto.
- 8. Realizar pruebas piloto:**

## Referencias

1. [Requisitos que debería cumplir](#)

## Conclusiones

Este trabajo se podría resumir en varias partes:

El objetivo principal de la interfaz: Que el usuario pueda utilizarla de manera sencilla y sin mucho esfuerzo.

Hacer una interfaz intuitiva, es decir, de manera que se vea todo claro y que siempre devuelva algún mensaje para informar al usuario de lo que esta pasando (independientemente de que el comando sea erróneo o no).



## **Mi valoración personal**

Mi valoración personal es que después de haber reunido toda esta información y de tener un diseño creo que esta interfaz es viable y en caso de que algo este mal se podría solucionar de manera sencilla.