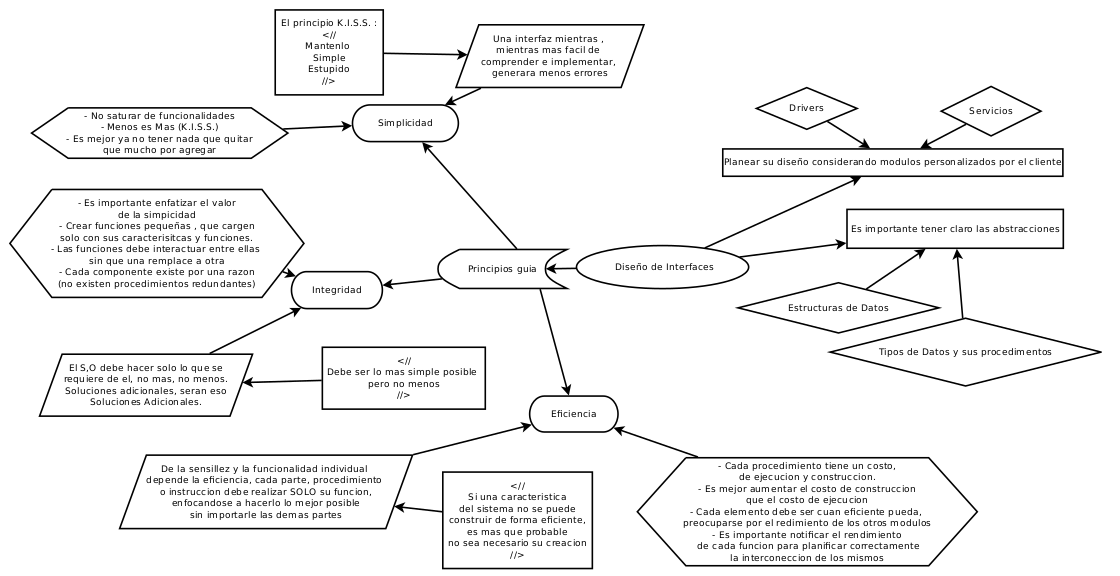
Integrantes :

* Medina Carbajal , Diego
* Medina Lopez, Jahir
* Marquina Gonazales, Alessandra
* Mendoza Campos , Renato

1. **Crear un mapa conceptual del tema.**

1. **Explicar cada principio.**

* **Simplicidad :**

Este principio busca garantizar que durante las etapas de desarrollo se planee cumplir con las funciones minimas necesarias, de forma que se reduzca lo maximo posible un debug futuro.

***Ejm. Visual***

*En este ejemplo se ejecuta el comando* **tree** en linux, este comando genera una vista de arbol de todos el contenido de un directorio, con solo 3 parametros :

-a : All , muestra todo el contenido

-L: Level, identifica hasta que profundida se debe generar el arbol

-d: Dir, especifica solo se debe mostrar los directorios.

Se consigue generar un listado de todos las carpetas que existen en el directorio actual , incluidas las carpetas ocultas.

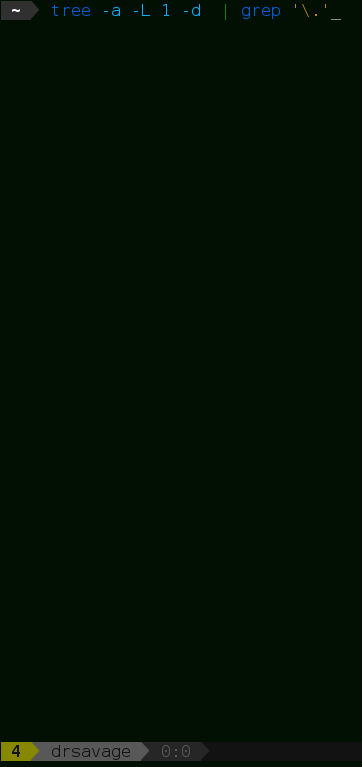
******

* **Integridad :**

Cada componente del sistema operativo debe cumplir su funcion, de forma que al interactuar con las diferentes partes o modulos del sistema , garantice un correcto funcionamiento.

***Ejm. Visual***

*Se conecta dos procesos :* **tree y grep** *, estos microprocedimientos permiten, el primero generar un arbol del contenido de una carpeta, mientras el segundo filtra la salida según un formato de verificacion, este caso ,* **grep** *busca todas las carpetas que tienen un* **“.”** *al comienzo de su nombre (carpetas ocultas)*

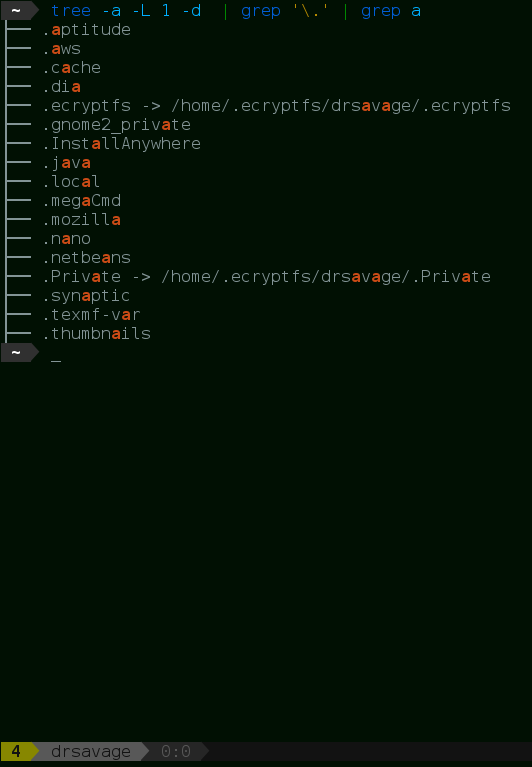
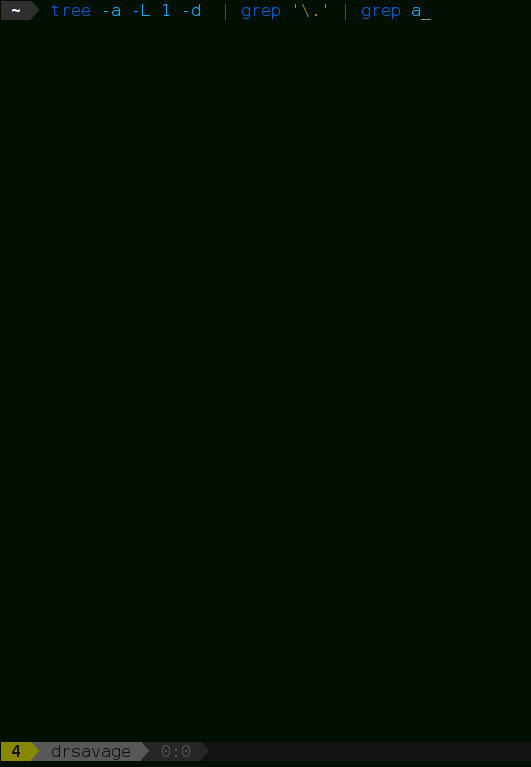
**

* **Eficiencia :**

Tal cual se intuye por simplicidad e integridad, la eficiencia debe garantizar que cada modulo , de forma independiente se ejecute lo mas rapido posible sin importar la eficiencia de los demas.

***Ejm. Visual***

*En esta parte final se juntan 3 procedimientos : tree, grep y nuevamente grep, siendo este ultimo el encargado del filtrado final:* **Buscar todas las carpetas dentro un directorio (solo el primer nivel) que esten ocultas (comienzen con .) y contengan una letra “a” en su nombre***. Estos 3 modulos simples, que individualmente se ejecutan rapidamente, construyeron un preocedimiento complejo en tiempo record. La construccion de la operación no tomo mucho tiempo y retorono el resultado deseado correctamente.*

******

1. **Explicar 3 ejemplos de la aplicacion o uso de estos principios.**

* **Ejemplo 1:**

La construccion los servicios AWS (amazon cloud) , cada “producto” , aun que reside sobre una misma estructura de servidores, esta planificado de tal forma que puedan comunicarse entre si como si de entidades independientes se trataran, esto implica, que de fallar uno, no se lleva por delante los demas servicios.

Cada servicio garantiza su propia eficiencia y calidad de servicio.

* **Ejemplo 2:**

La implementacion de un servidor de almacenamiento de correo electronico, si bien es cierto, esto reside en un S.O. y no es como tal uno, su construccion hace uso de los 3 principios, pues debe garantizar una alta confiabilidad.

Cada modulo (recepcion de email, edicion, filtrado, etc) debe operar sin afectar al otro pero si apoyando y constituyendo en conjunto un servicio confiable.

* **Ejemplo 3:**

El sistema operativo **Open BSD** , este sistema operativo es una variante de la arquitectura MS-DOS , el cual busca la ejecucion de programas asociados a esta arquitectura pero sin ser privativo (como el caso de windows, creador de la arquitectura MS-DOS).

En la creacion de este sistema operativo fue importante trabajar usando los 3 principios, pues no solo se debe garantizar su funcionamiento sino tambien la interoperatibilidad con otro S.O. : Microsoft Windows, pues este S.O. se presenta como una alternativa libre a Windows.