

Objetivos del proyecto

- Implementar un File System.
- Tener una experiencia en el uso de Sistemas Distribuidos

Definición general

Este proyecto tiene como finalidad implementar un File System. Para esto se implementaran las funciones del File System que se desglosan posteriormente. La idea es que se tenga una serie de comandos y el sistema responda a cada uno de ellos. El File System se implementará de una forma distribuida. Esto implica que debe haber un servidor, que es el que sabe que hacer con cada comando y un cliente que es la aplicación que solicita los comandos. Con cada comando que se introduce el cliente debe ir y ejecutar el método remoto correspondiente que ha sido publicado por el servidor (preferiblemente Web Services). Pueden utilizar las estructuras de datos que ustedes diseñen para mantener la información del sistema. El cliente del File System no debe tener una interfaz complicada, basta con que el usuario pueda ingresar los diferentes comandos y pueda observar la respuesta del File System. El proyecto será implementado en Java.

Descripción Detallada

- File System: Deben simular un File System. Esto implica que:
 - debe mantener una estructura de archivos
 - saber en que nivel de la estructura se encuentra uno en determinado momento. Debe ser claro la ruta actual, y esta debe estar disponible al usuario en todo momento.
 - debe permitir navegar entre los directorios del File System.
 - debe tener la seguridad correspondiente relacionada a no crear dos directorios o archivos con el mismo nombre en un directorio.
 - si se va a crear un archivo o directorio con el nombre de uno que ya existe debe preguntar si le desea "caer encima" al archivo/directorio ya existente.
 - debe avisar en el caso que el disco este lleno
- Funciones del File System: Los comandos o funciones a implementar son las siguientes:
 - CREATE: Este comando lo utilizaremos para crear un disco virtual. Los parámetros serán la cantidad de sectores y el tamaño del sector.
 - FILE: Crear un Archivo. Se le debe definir el contenido del archivo y el nombre y extensión de este.
 - MKDIR: Este comando crea un directorio en el directorio Actual. El parámetro es el nombre del directorio.
 - CambiarDIR: Permite cambiar el directorio actual. Me debe permitir irme a un directorio cualquiera de la estructura de directorios actual
 - ListarDIR: Lista los archivos y directorios dentro del directorio actual. Debe mostrar una diferencia clara entre los directorios y archivos.
 - ModFILE: Se puede seleccionar un archivo y cambiarle el contenido.
 - VerPropiedades: Permite ver las propiedades de un archivo. Nombre, Extensión, Fecha de Creación, Fecha de Modificación y tamaño.
 - ContFile: Para un determinado archivo se debe poder ver el contenido del archivo.
 - CoPY: Se implementarán 3 tipos de copias. (aplica a archivos o directorios)
 - un archivo con ruta "real" será copiado a una ruta "virtual" de MI File System.
 - un archivo con ruta "virtual" de MI File System será copiado a una ruta "real"
 - un archivo con ruta "virtual" de MI File System será copiado a otra ruta "virtual" de MI File System.
 - MoVer: Mover un archivo o directorio. Nótese que el MV sirve como rename, pues se puede mover al mismo directorio con otro nombre.

- ReMove: Con este comando se eliminan archivos. La eliminación puede ser normal para uno o varios archivos o recursiva en el caso de directorios.
- FIND: Recibe un nombre de archivo o directorio y lista todas las rutas de MI File System con archivos de ese nombre. (Nótese que puede recibir de parámetro algo como "*.doc")
- TREE: Despliega, simulando un árbol, la estructura de archivos del File System. Debe estar visible siempre.
- Clientes: Recordar que pueden haber varios clientes utilizando los servicios del File System. Cada cliente debe ver solo la estructura del File System que creó. Los "discos" virtuales no deben ser borrados, pero no se podrán acceder una vez que un cliente terminó su "sesión"
- Parte Distribuida: Se recomienda utilizar una aplicación que publique Web Services. Pero lo importante es que el servidor esté remoto a los clientes.

Documentación

No se va a realizar una documentación para calificar, debido a la fecha en que se entregará el proyecto pero si es necesario que entreguen lo siguiente para la revisión:

- Un resumen con el resultado de la programación, qué sirve y qué no sirve.
- Cómo compilar y ejecutar los programas.

Aspectos Administrativos

- El desarrollo de este programa debe de realizarse en grupos de exactamente dos personas salvo acuerdo con el profesor.
- El trabajo se debe de entregar el día 19 de Junio de 2016 a las 12:00m.n por correo.
- Las revisiones se realizaran el día 20 de Junio.
- Recuerde que oficialmente no se recibirán trabajos con entrega tardía.