***Practica ud1- ejer 2***

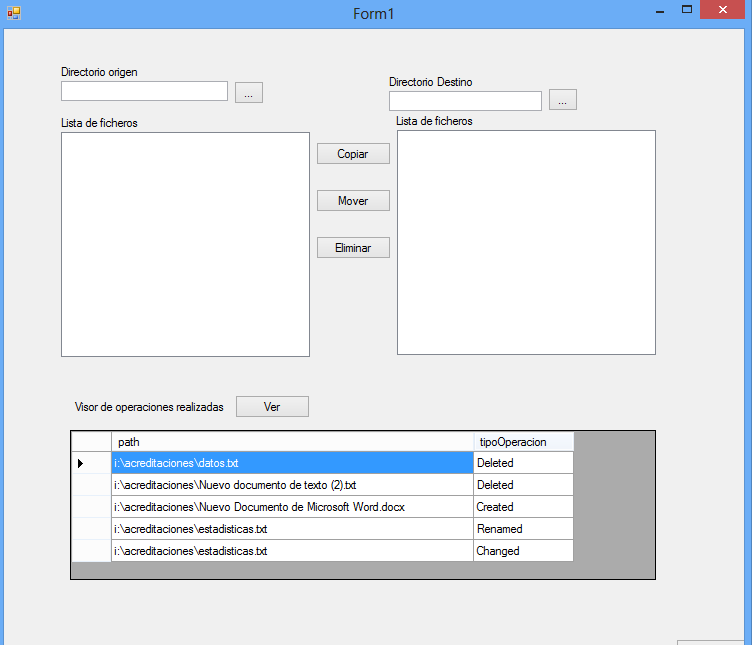
Desarrollar una aplicación que visualice los archivos del directorio origen y del directorio destino que se seleccionaron.

La lista de archivos que pertenecen al directorio origen permitirá selección múltiple, dicha selección será copiada o movida al directorio destino según se use copiar o mover. La función eliminar será aplicable únicamente a los archivos del directorio origen que fuesen seleccionados.

Todas estas operaciones se realizarán con vigilancia de modo que se registrara en un ArrayList de objetos de la clase Vigilancia .

|  |
| --- |
| Vigilancia |
| + path:string  + operación:string |

La opción Ver mostrará el log de las operaciones realizadas con los archivos del directorio origen



La clase ***FileSystemWatcher***  proporciona un servicio de vigilancia del estado del sistema de archivos mediante el que es posible detectar cualquier cambio que se produzca en él (creación y borrado de archivos, modificación de sus atributos, etc.) (Ver pag. 31 apuntes ficheros.pdf)

Ejemplo del manejo de la clase  ***FileSystemWatcher***

|  |
| --- |
| / ver los cambios en el directorio de la aplicación y sobre todos los archivos  FileSystemWatcher watcher = new FileSystemWatcher(@"i:\acreditaciones", "\*.\*");  // ver el nombre del archivo y tamaño cambiado  watcher.NotifyFilter = NotifyFilters.FileName | NotifyFilters.Size|NotifyFilters.Attributes;  watcher.Changed += new FileSystemEventHandler(OnChange);  watcher.Created += new FileSystemEventHandler(OnChange);  watcher.Deleted += new FileSystemEventHandler(OnChange);  watcher.Renamed += new RenamedEventHandler(OnChange);  watcher.EnableRaisingEvents = true;    private void OnChange(object sender, FileSystemEventArgs e)  {  // e.ChangeType.ToString()🡪 se obtiene la operación  // e.FullPath🡪 se obtiene el path  } |