Project 4: File System Manager

Austin Johns

Grand Canyon University: ENT-446

26 April 2020

**Previous Project**

The previous projects this semester have been surrounding creating a Unix shell command line interpreter. In these projects, we have created a command line interpreter that is capable of executing commands, interpreting multiple lines, and multithreading. Deadlock avoidance was also included in these projects to ensure that processes had the resources that they needed. In the latest project, a process scheduler was created that specifically handled multiple processes simultaneously.

**Project Description**

In this project, a file system hierarchy will be implemented as a directory structure. The shell will utilize 13 commands to take advantage of this new filesystem. The file system manager maintains good storage space and intelligently organizes information. It manages the file lifecycle and directory lifecycle as well.

**Methodology Explanation**

In this project, file directories and files are organized and analyzed to make the best of the shell and file system. Changes to files can be made once the file is located and verified. This is done by tracking paths, and this is where the directory comes in. To test this, commands such as -d to delete a file are used to check if this works. After deleting, the interpreter’s job is to find the file. It will follow the path and be unable to find the file.

**Assumptions**

For this project, we will assume that a user is able to and fluent in their understanding of the file system. It can be assumed that the user understands the crucial parts of the filesystems path in order to not corrupt the filesystem.

**Commands in the Manager**

1. Deletion of directories with handling for non-empty directories
   1. Deletes unnecessary directories in the filesystem. If a directory has any data stored along its path, the user will be alerted to prevent any accidentally lost data.
2. Move file to another directory
   1. Change the path of a file
3. Copy a directory
   1. Copies files and stores them along another path in bulk.
4. Copy a file
   1. Copies files to change or add to another path.
5. Create, rename, delete a file
   1. Creates files, renames, deletes, alters the path of a file in the directory.
6. View a directory
   1. Given a starting node, the user can view all paths of the directory and its contents.
7. Acquire information about a file
   1. Gives file information (size, type, metadata)
8. Acquire information about a directory
   1. Information about a directory (size, type, metadata, contents, path)

**Screenshot of Successful Execution**

**A screenshot of a cell phone

Description automatically generated**