

Customized Virtual File System Documentation

- Name of the Project: Customized Virtual File System
- Technology Used: System Programming using c++
- User Interface: Command User Interface
- Platform Required: Windows NT Platform OR Linux Distributions
- Architectural requirement: Intel 32 bit processor
- SDK used : None
- Description of the Project
 - In this project we emulate all data structures which are used by operating system to manage File system oriented tasks.
 - As the name suggest its virtual because we maintain all records in Primary storage.
 - In this project we create all data structures which required for File Subsystems.
 - We provide all implementations of necessary system calls and commands of File subsystem as Open, Close, Read, Write, Lseek, Create, RM, LS, Stat, Fstat etc.
 - While providing the implementations of all above functionality we use our own data structures by referring Algorithms of UNIX operating system.
 - By using this project we can get overview of UFS (UNIX File System) on any platform.
- Data Structures Used in the Project
 - Inode Table,
 - File Table
 - UAREA
 - User File Descriptor Table
 - Super block
 - Disk Inode List Block
 - Data Block
 - Boot Block etc.
- Diagram of Data Structures Used: (Diagram here)
 -
- How to used this Project
 - Commands to start this project
 - ```
1 cd Customized virtual File System- project
2 g++ File-system.cpp -o file-exe
3 ./file-exe
```
  - Now you are in Project Customized VFS : >
    - help command will give all list of commands
      - ```
1 Customized VFS: > help
```
 - Code snippet of terminal after project started

```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\admin\Desktop\Customized Virtual File System - Project> g++ File-system.cpp -o file-exe
PS C:\Users\admin\Desktop\Customized Virtual File System - Project> ./file-exe
DILB created successfully

Customised VFS : > help
ls      : To List out all files
clear   : To clear console
open    : To open the file
close   : To close the file
closeall : To close all opened files
read    : To Read the contents from file
write   : To Write contents into file
exit    : To Terminate file system
stat    : To Display information of file using name
fstat   : To Display information of file using file descriptor
truncate : To Remove all data from file
rm      : To Delete the file
backup  : To Take Backup Of All Created Files

Customised VFS : >

```

- command to used Customized VFS:
 - man : man command will give description of how to used different commands link read, write, create etc

```
1 Customised VFS: > man create
```

```

File Edit Selection View Go Run ...
EXPLORER CUSTOMIZED VIRTUAL FIL...
  file-exe.exe
  File-system.cpp 2
  hi.txt
  myfile.txt
  README.md
  > OUTLINE
  > TIMELINE
  2 0 0

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\admin\Desktop\Customized Virtual File System -
DILB created successfully

Customised VFS : > man create
Description : Used To Create New Regular File
Usage : create File_Name, Permission

Customised VFS : > man read
Description : Used to read data from regular file
Usage : read File_name, No_Of_Bytes_To_Read

Customised VFS : > man write
Description : Used to write data into regular file
Usage : write File name
After this enter the data that we want to write

Customised VFS : > man ls
Description : Used to list all information of files
Usage : ls

Customised VFS : > man stat
Description : Used to display information of file
Usage : stat File_name

Customised VFS : > man fstat
Description : Used to display information of file
Usage : stat File_Descriptor

Customised VFS : > man closeall
Description : Used to close all opened file
Usage : closeall

Customised VFS : > man close
Description : Used to close opened file
Usage : close File_name

Customised VFS : >

```

-
- clear : clear command will used to clear the console
 - 1 Customised VFS: > clear
- create: create command is used to create new file with permission 1 (read only), 2(write only) and 3(read and write).

```
1 Customised VFS : > create myfile.txt 3
2 File is successfully created with file descriptor : 0
```

```
1 Customized VFS : > create myfile.txt 1 // read only file
```

```
1 Customized VFS : > create myfile.txt 2 // write only file
```

```
1 Customized VFS : > create myfile.txt 3 // read and write only file
```

- write: write command is used to write in file

```
1 Customised VFS : > write myfile.txt
2 Enter the data :
3 Hello world
4 Sucessfully : 11 bytes written
5 Customized VFS: >
```

- read : read command is used to read file and we have to specified how many bytes we want to read.

```
1 Customized VFS: > read myfile.txt
2 Hello world
3 Customized VFS : >
```

- open : open command is used to open file in mode 1 (read mode), 2(write mode), 3 (read and write mode).

```
1 Customized VFS: > open myfile.txt 3
```

- close : close command is used to close file

```
1 Customized VFS: > close myfile.txt
```

- ls : ls command will give list of all command which are available.

```
1 Customised VFS : > ls
2
3 File Name      Inode number   File size      Link count
4 -----
5 myfile.txt      1              0              1
6 -----
7
8 Customised VFS : >
```

- closeall : closeall command is used to close all open file.

```
1 Customized VFS: > closeall
```

- stat: stat command give statistical information about file that is properties.

```
1 Customised VFS : > stat myfile.txt
2
3 -----Statistical Information about file-----
4 File name : myfile.txt
5 Inode Number 1
6 File size : 1024
7 Actual File size : 0
8 Link count : 1
9 Reference count : 3
10 File Permission : Read & Write
11 -----
12
13
14 Customised VFS : >
15
```

- truncate: truncate command is used to delete data from file

```
1 Customised VFS : > truncate myfile.txt
2 Data Succesfully Removed
```

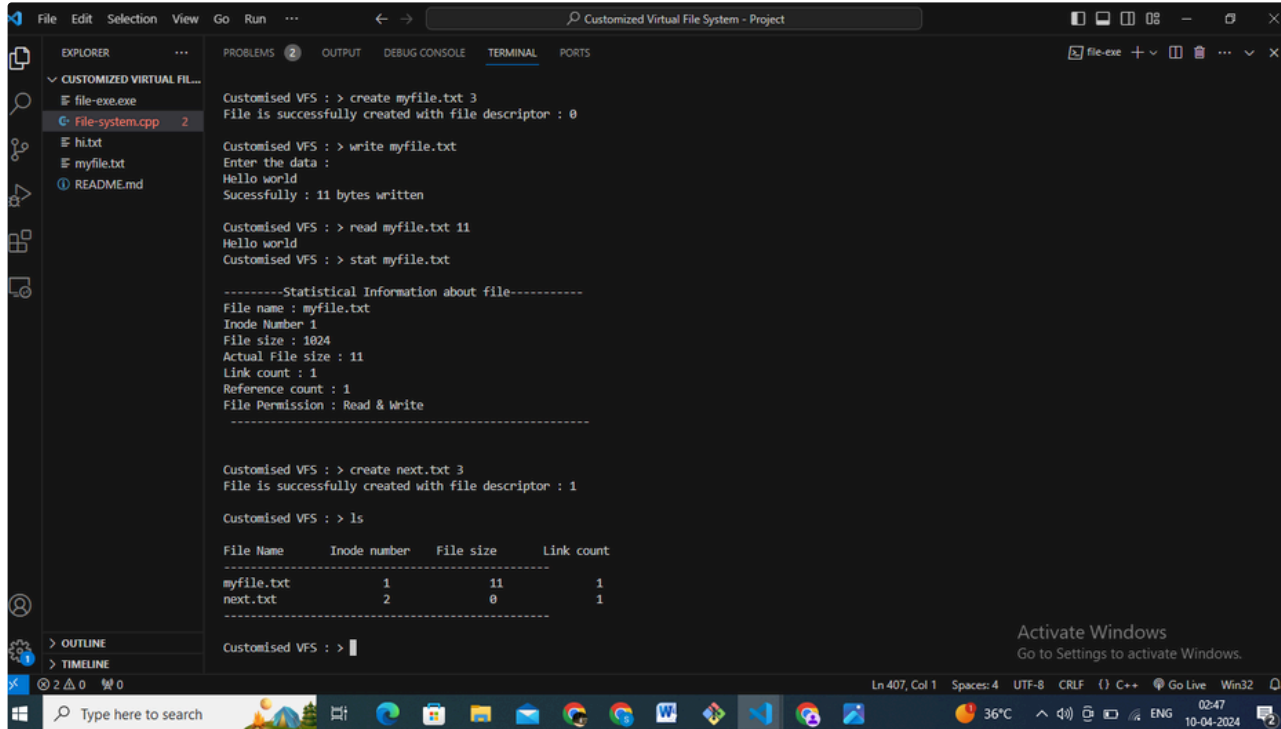
- rm: rm command used to remove the file.

- 1 Customised VFS : > rm giraff.txt
- 2 File Successfully Deleted
- 3

◦ exit: exit command is used to terminate file system

- 1 Customized VFS : > exit

- Code snippet of terminal



```
Customised VFS : > create myfile.txt 3
File is successfully created with file descriptor : 0

Customised VFS : > write myfile.txt
Enter the data :
Hello world
Successfully : 11 bytes written

Customised VFS : > read myfile.txt 11
Hello world
Customised VFS : > stat myfile.txt

-----Statistical Information about file-----
File name : myfile.txt
Inode Number 1
File size : 1024
Actual File size : 11
Link count : 1
Reference count : 1
File Permission : Read & Write

Customised VFS : > create next.txt 3
File is successfully created with file descriptor : 1

Customised VFS : > ls

File Name      Inode number   File size     Link count
-----
myfile.txt      1              11            1
next.txt        2              0             1

Customised VFS : >
```

Ln 407, Col 1 Spaces: 4 UTF-8 CRLF {} C++ Go Live Win32

36°C 02:47 10-04-2024

- Thank you for using this documentation.