Green University of Bangladesh

Department of Computer Science and Engineering (CSE)
Semester: (Spring, Year: 2024), B.Sc. in CSE (Day)

PROJECT REPORT

File Controlling System

Course Title: Operating System Lab
Course Code: CSE 310
Section: 221 D3

Students Details

Name	ID
MD Rakib Sarder	221902377
Sazzad Hossain Shihab	221902182

Submission Date: 13-06-2024
Course Teacher's Name: Md. Rajibul Palas

[For teachers use only: Don't write anything inside this box]

Project Report Status	
Marks:	**Signature:**
Comments:	**Date:**

Contents

Chapter 1

1.1 Introduction

The OS - File Controlling System is a comprehensive command-line utility designed to facilitate efficient file management using a user-friendly interface built with whiptail. This project allows users to perform various file and directory operations through a menu-driven interface, enhancing the ease of file management tasks on Unix-like operating systems.

1.2 Objectives:

The primary objectives of the OS - File Controlling System project are as follows:

- To provide a user-friendly interface for managing files and directories using a menu-driven approach.
- To simplify the process of creating, editing, renaming, and deleting files and directories.
- To ensure secure access to the file management system through user authentication, including sign-up and sign-in functionalities.
- To enable efficient file searching and viewing of file details and contents.
- To offer file deletion options, including temporary deletion with the ability to restore files and permanent deletion with logging.
- To counting and listing of files and directories within the current directory.
- To enhance user experience by providing progress indicators and clear feedback on operations.
- To support sorting of file content, making it easier to organize and access information within files.
- To continuously improve the system based on user feedback and usage patterns.

1.3 Features:

The OS - File Controlling System project offers the following key features:

1. **User Authentication:** Secure sign-up and sign-in functionalities to authenticate users and protect system access.
2. **File Management Menu:** A menu-driven interface for easy management of files and directories.
3. **File Operations:** Options to create, edit, rename, and delete files and directories directly from the menu.
4. **File Search:** Ability to search for files by name within the current directory.
5. **File Details:** Display of detailed information about a selected file, including permissions and timestamps.
6. **File Counting:** Count and display the number of files in the current directory.
7. **File Sorting:** Sort the contents of a file alphabetically.
8. **Temporary Delete:** Move files to a temporary trash directory with the option to restore.
9. **Permanent Delete:** Permanently delete files with logging of deletions for record-keeping.
10. **Trash Management:** View and restore files from the temporary trash directory.
11. **Progress Indicators:** Visual progress indicators during file operations for user feedback.

Chapter 2

2.1 Methodology

- **Project Planning and Requirements Gathering**: The initial phase involved defining the scope, objectives, and requirements of the OS - File Controlling System project. This included identifying key functionalities such as file management, user authentication, and interactive menu design.
- **Design and Architecture**: Following requirements gathering, the system architecture was designed to ensure scalability, modularity, and user-friendly interaction. This phase included creating flowcharts and outlining the menu structure for seamless navigation.
- **Implementation**: With the architecture finalized, the project moved to the implementation stage. Bash scripting was employed to develop functionalities such as file creation, editing, renaming, searching, and deletion. Each function was meticulously coded to ensure reliability and adherence to Unix shell scripting standards.
- **Testing**: Comprehensive testing was conducted to validate the functionality and performance of the implemented features. This included unit testing of individual script components, integration testing of the entire system, and user acceptance testing to ensure ease of use and reliability.
- **Deployment**: After successful testing, the OS - File Controlling System was deployed on the target environment. This involved configuring the system on a Linux server environment and ensuring compatibility with different terminal emulators.
- **User Training and Documentation**: User manuals and documentation were prepared to guide users on system navigation, functionality usage, and troubleshooting tips. Training sessions were conducted to familiarize users with the features and operations of the system.
- **Maintenance and Support**: Ongoing maintenance involves monitoring system performance, addressing user feedback, and updating scripts to accommodate new requirements or address issues. Regular backups of user data and system configurations are maintained to ensure data integrity and system reliability.

2.2 Results

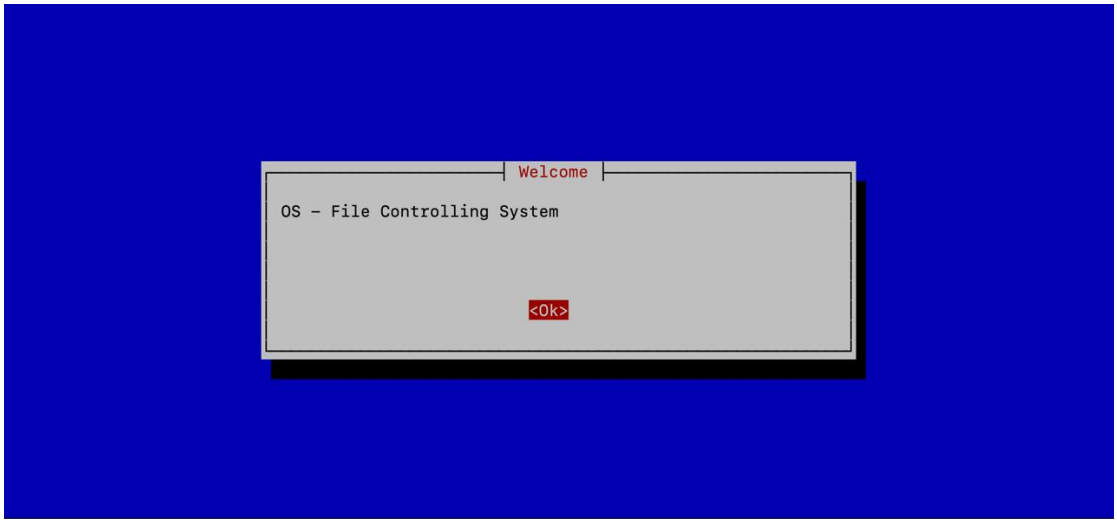

Figure 1: Welcome Screen

.

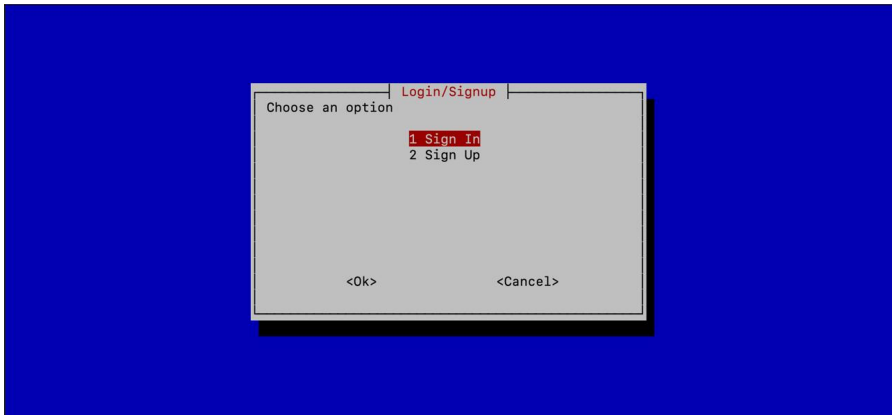

Figure 2: Signin or Signup

.

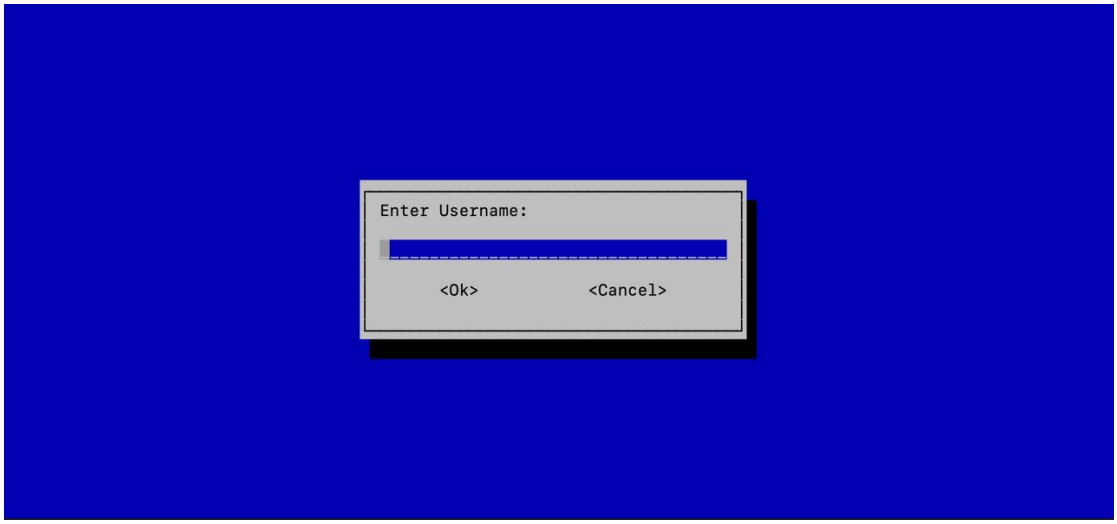

Figure 3: Adding Username

.

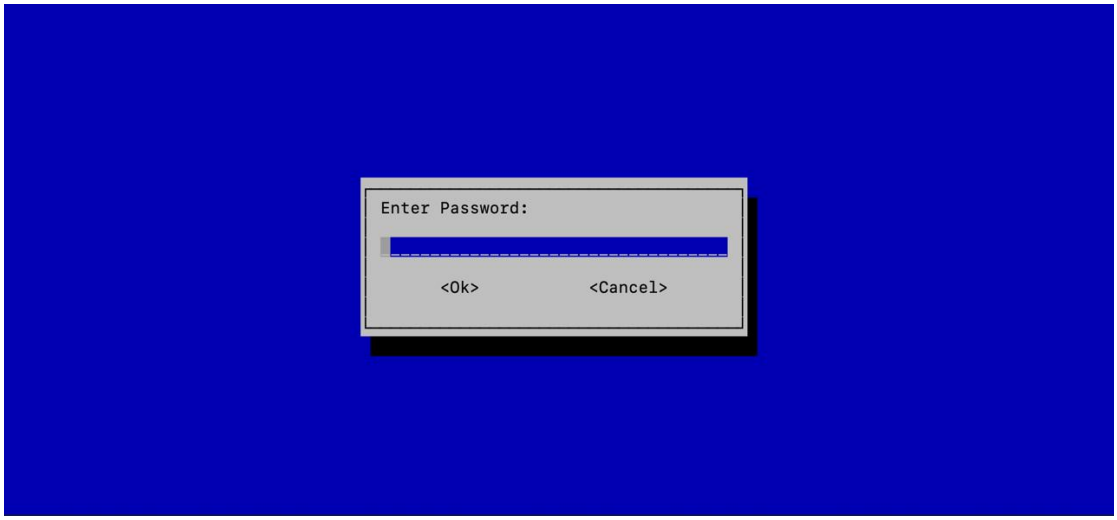

Figure 4: Adding Password

.

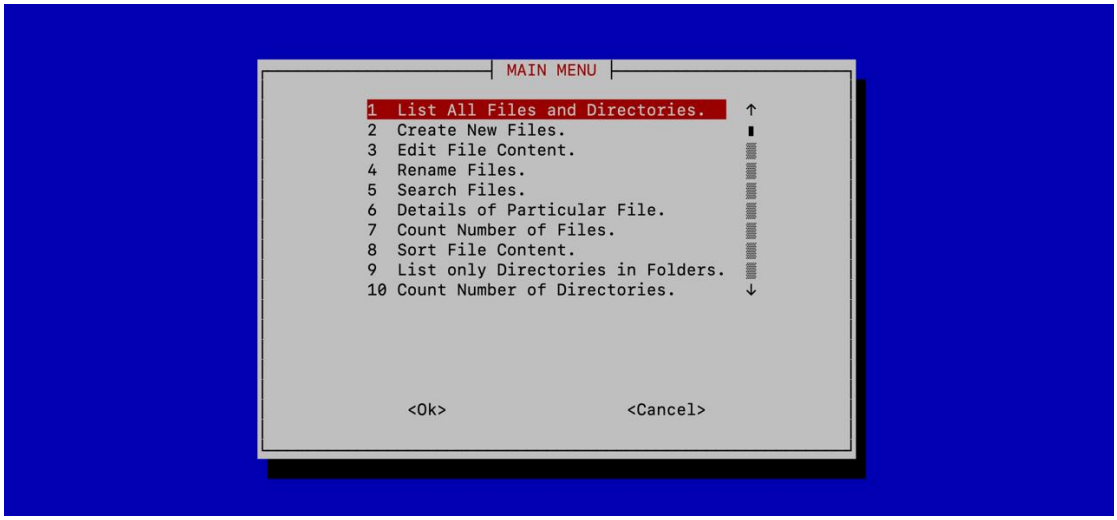

Figure 5: Menu Screen

.

Figure 6: Ending Screen

.

Chapter 3

3.1 Conclusion

The OS - File Controlling System addresses the challenges and requirements of efficient file management in a Unix/Linux environment. In today's world, where data organization and system administration are critical, the demand for such a system has become a pressing concern. It is essential to have a well-organized tool that enables users to manage files and directories effectively, ensuring a robust solution for those in need.

By utilizing Bash scripting and Unix shell commands, the system streamlines file operations such as creation, deletion, renaming, and searching, thereby enhancing overall file management efficiency. The integration of user authentication through a login and signup mechanism adds a layer of security, ensuring that only authorized users can access and manipulate files. This project demonstrates how leveraging the capabilities of shell scripting can lead to a significant improvement in managing file-related operations, providing a user-friendly and reliable solution for users in a Unix/Linux environment.

3.2 References:

- W3School: https://www.w3schools.io/terminal/bash-tutorials/
- ChatGPT: https://chat.openai.com/
- Google: www.google.com
- Youtube: www.youtube.com
- Whiptail: https://www.redhat.com/sysadmin/use-whiptail
- Javapoint: https://www.javatpoint.com/shell-scripting-tutorial