

# Project Documentation:

## LockedMeApp

## Project and Developer Details:

- **Project Name:** LockedMeApp
- **Developer:** André Bessa
- **Email:** [andre.bessa@vodafone.com](mailto:andre.bessa@vodafone.com)

## GitHub Repository:

- **GitHub Repository:** <https://github.com/AndreBessa-cmd/Phase1EndProject.git>
- **Token:** ghp\_FkurPiMCxmO8UeLLF7gGRKZCCf6kif1T0Kuj

## **Sprints Planned and Achieved Tasks:**

Duration of Sprints: 1 week

### **Sprint 1:**

- Task 1: Set up the project structure.
- Task 2: Implement the main menu.
- Task 3: Implement listing files in ascending order.

### **Sprint 2:**

- Task 1: Implement the operations menu.
- Task 2: Implement file addition functionality.
- Task 3: Implement file deletion functionality.

### **Sprint 3:**

- Task 1: Implement file search functionality.
- Task 2: Implement error handling and user input validation.
- Task 3: Finalize testing and documentation.

## **Algorithms of the application:**

- **List Files in Ascending Order:**

Get the root directory path.

Create a File object for the root directory.

Get a list of files and directories in the root directory.

Sort the list of files in ascending order.

Iterate through the sorted list and display each file's name.

## • Add a File:

Get the user's input for the file name to add.

Construct the full file path using the provided name.

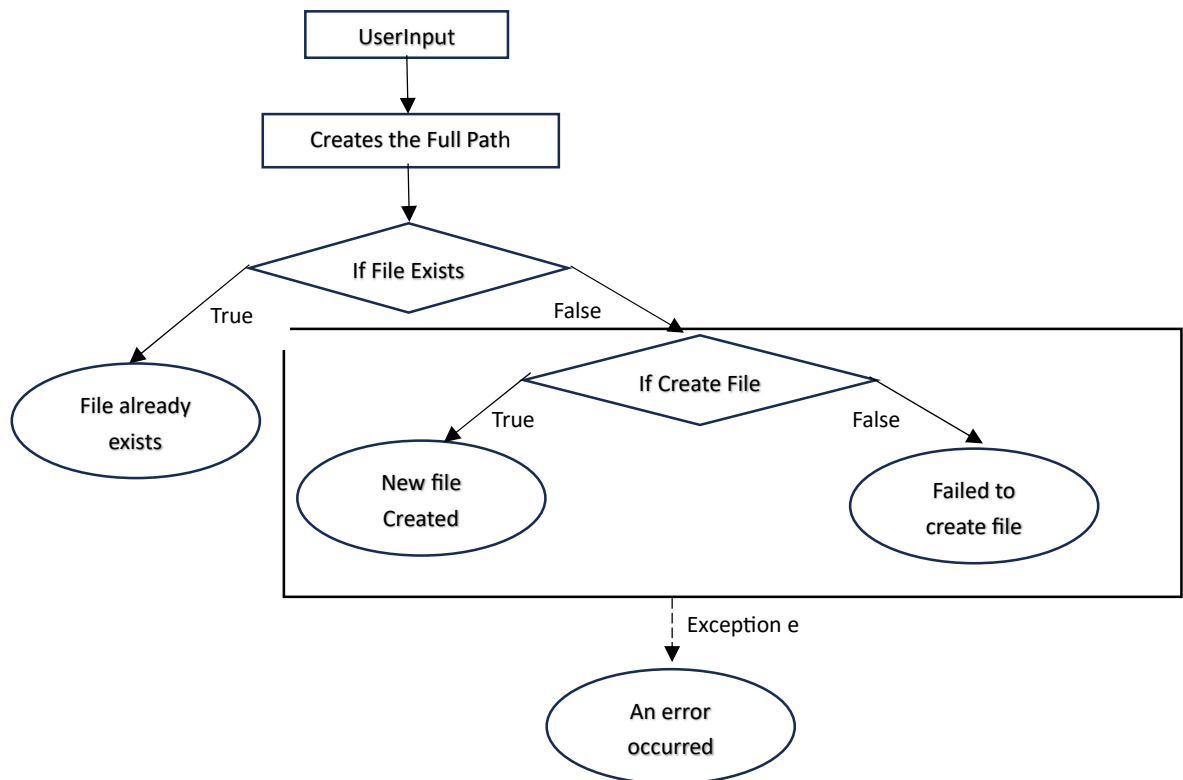
Check if the file already exists at the specified path.

If the file exists, display "File already exists" message.

If the file doesn't exist, attempt to create a new file.

If successful, display "New file created with name: [filename]" message.

If not successful, display "Failed to create the file" message.



## • Delete a File:

Get the user's input for the file name to delete.

Construct the full file path using the provided name.

Check if the file exists at the specified path.

If the file exists, attempt to delete the file.

If successful, display "File deleted" message.

If not successful, display "Failed to delete the file" message.

If the file doesn't exist, display "File not found: [filename]" message.

## • Search for a File:

Get the user's input for the file name to search.

Construct the full file path using the provided name.

Check if the file exists at the specified path.

If the file exists, display "File found: [filename]" message.

If the file doesn't exist, display "File not found: [filename]" message.

## **Core Concepts Used:**

- File I/O: Reading and writing files in the specified directory.
- Exception Handling: Handling user input errors and file-related exceptions.
- User Interface: Creating a command-line interface for user interaction.

## **Conclusion:**

LockedMeApp is a command-line file management application. It provides users with essential file operations such as listing files in ascending order, adding files to the directory, deleting files, and searching for files.

### **Unique Selling Points (USPs):**

1. **User-Friendly Interface:** LockedMeApp offers an intuitive command-line interface for users to manage files easily.
2. **Error Handling:** The application includes robust error handling to ensure a smooth user experience, even in cases of invalid input or file-related issues.
3. **Case Sensitivity Control:** File operations consider case sensitivity, ensuring precise actions on files with matching names.

The project's GitHub repository contains the source code and documentation, providing transparency and accessibility to interested parties.

LockedMeApp is a versatile tool for file management, suitable for various use cases and environments.