LockedMe – Virtual Key for Repositories

1. Project and developer details

The project is developed by Nidhi Lunagariya. The code for this project is hosted at https://github.com/Nidhi16797/Phase1Assignment

2. Sprints planned, and the task achieved in them

The project is planned to be completed in 2 sprints.

In first sprint,

- Creating the flow of the application.
- Creating the new Repository name Phase1Assignment in GitHub.
- Creating the Algorithm of the application.
- Creating this specification document.

In Second sprint,

- Initializing git repository to track changes as development progresses.
- Writing the Java program to fulfill the requirements of the project.
- Testing the Java program with different kinds of User input.
- Pushing code to GitHub.

3. Algorithms and flowcharts of the application

- Algorithm for Proj.java File

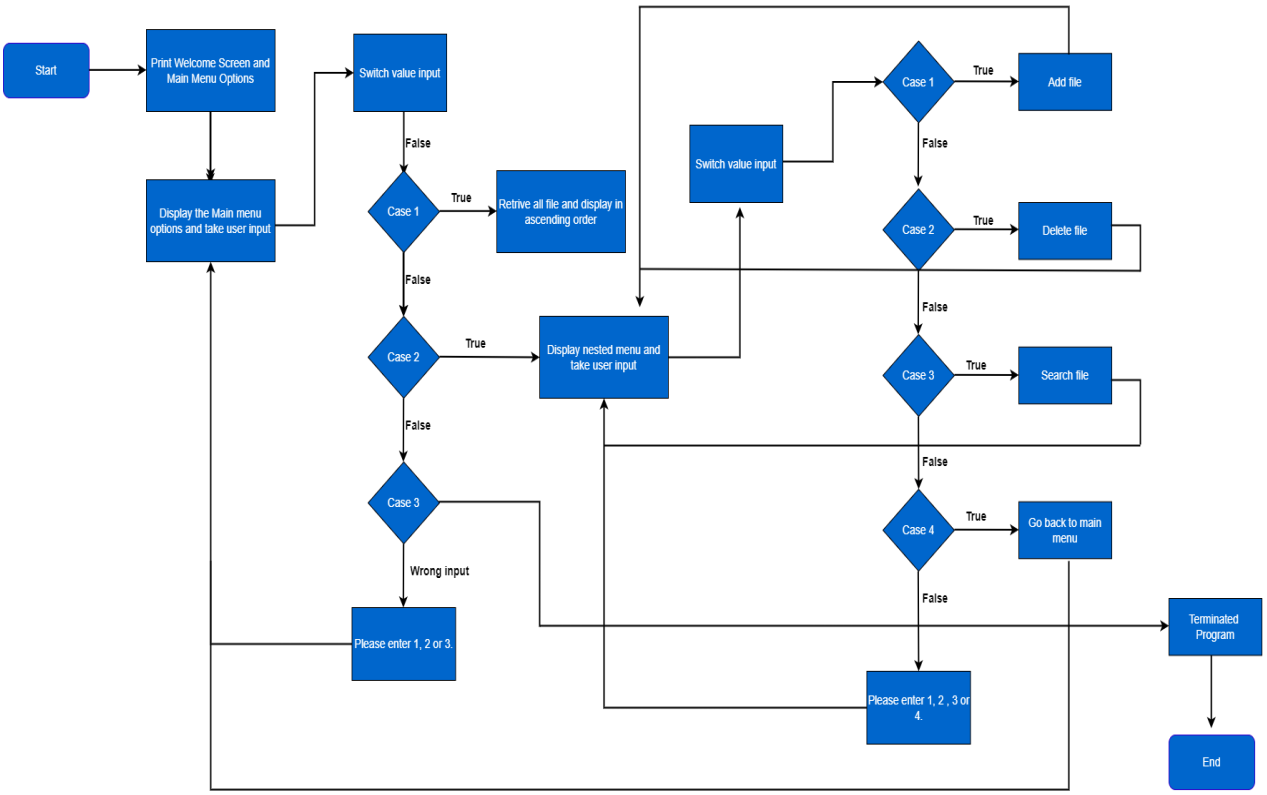
Step 1: Start
Step 2: Declare the string variable WELCOME_STRING, MAIN_MENU_STRING, NESTED_MENU_STRING and file object variable file_name.
Step 3: Declare the function for fullfill the requirement of the project. Function names are showNestedMenu(),

showFilesAscendingOrder, addFile(String fname), deleteFile(String fname), searchFile(String fname), showMainMenu().

Step 4: Create the Proj.java file object in main method and call the function name showMainMenu() .

Step 5: End.

- Flowchart of the application



4. Core concepts used in the project

- Collections framework
- File Handling
- Flow Control
- Sorting and searching techniques
- Exception Handling

5. Links to the GitHub repository to verify the project completion

or create a new repository on the command line

```
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Nidhi16797/Phase1Assignment.git
git push -u origin main
```

or push an existing repository from the command line

```
git remote add origin https://github.com/Nidhi16797/Phase1Assignment.git
git branch -M main
git push -u origin main
```

6. Unique Selling Points of the Application

1. The application is designed to keep on running and taking user inputs even after exceptions occur. To terminate the application, appropriate option needs to be selected.

2. User is also provided the option to write content if they want into the newly created file.

3. The user is able to seamlessly switch between options or return to previous menu even after any required operation like adding, searching, deleting or retrieving of files is performed.

4. The application is designed with modularity in mind. Even if one wants to update the path, they can change it through the source code. Application has been developed keeping in mind that there should be very less “hardcoding” of data.

7. Conclusions

Further enhancements to the application can be made which may include:

- Conditions to check if user is allowed to delete the file or add the file at the specific locations.
- Asking user to verify if they really want to delete the selected directory if it’s not empty.
- Retrieving files/folders by different criteria like Last Modified, Type, etc.