**Locked Me – Virtual Key for Repositories**

This document contains sections for:

1. [Project Details and Problem Statement](#DESCRIPTION)
2. [Sprint planning and Task completion](#Sprint_plan)
3. [Algorithm and flowchart](#Algorithm)

The code for this project is hosted at <https://github.com/amsdhs36/VirtualRepository>

The project is developed by Amruta Singh.

**1.Project Details**

**Project objective:**

As a Full Stack Developer, complete the features of the application by planning the development in terms of sprints and then push the source code to the GitHub repository. As this is a prototyped application, the user interaction will be via a command line.

**Background of the problem statement:**

Company Lockers Pvt. Ltd. hired you as a Full Stack Developer. They aim to digitize their products and chose LockedMe.com as their first project to start with. You’re asked to develop a prototype of the application. The prototype of the application will be then presented to the relevant stakeholders for the budget approval. Your manager has set up a meeting where you’re asked to present the following in the next 15 working days (3 weeks):

* Specification document - Product’s capabilities, appearance, and user interactions
* Number and duration of sprints required
* Setting up Git and GitHub account to store and track your enhancements of the prototype
* Java concepts being used in the project
* Data Structures where sorting and searching techniques are used.
* Generic features and three operations:
  + Retrieving the file names in an ascending order
  + Business-level operations:
    - Option to add a user specified file to the application
    - Option to delete a user specified file from the application
    - Option to search a user specified file from the application
    - Navigation option to close the current execution context and return to the main context
  + Option to close the application

The goal of the company is to deliver a high-end quality product as early as possible.

## **2. Project planning and Task completion**

* Initializing git repository to track changes as development progresses.
* Prepare flowchart and algorithm
* Writing the Java program to fulfill the requirements of the project.
* Pushing code to GitHub.

The project is planned to be completed in 2 sprints. Tasks assumed to be completed in the sprint are:

|  |  |  |  |
| --- | --- | --- | --- |
| Sprint | User Story | Tasks | Duration |
| 1 | As a user, I need to see the application welcome screen with Menu options. Menu First Option: User should able to display the files in ascending order from the directory. The root directory can be either empty or contain few files or folders in it.Second Option Exit | Create the Welcome screen | 1 |
| Create Check if directory exist or create new | 1 |
| If exist and choice enter by user to display the files in ascending order | 1 |
| 2 | Second option: Another menu inside Second option with choices -User should be able to 1.add, 2. delete files, 3. search file from the directory.  4.User should be able to get back to previous menu  Third Option: Exit the menu | Add methods to add the file in the directory, delete method to delete file | 3 |
| Search the user specified file from directory and give error message if not found | 2 |
| Go back to previous menu or exit the menu | 1 |

**3.Algorithm and Flowchart**

**User Story1**

**Steps:**

1. Start
2. Check if directory exist else create a new directory in the project
3. Print the welcome screen
4. Display the menu
5. Take user choice from menu and switch to user choice
6. If user choice 1 then display the files inside directory in ascending order
7. If user choice 2 then exit
8. Default statement: Please enter correct input

**Algorithm to display the files inside the directory in ascending order**

**Steps:**

1. Create a File object for the main directory.
2. Get an array of files for the main directory.
3. If array[i] is a directory: Print out directory name
4. If array[i] is a file: Print out the file name.(recursively)

Flowchart:

Start

Create “Main” folder if not exist

Take Input and Switch

Take Input and Switch

Print Welcome Screen and app details

Case 1se 1

Add the file and display submenu

Case 1se 1

Display all files from the Main folder and then display menu again

Display Menu

1. Display the files
2. Display the sub menu
3. Exit

**Case 2** 222

Display Sub menu:

1.Add file 2. Delete File 3.Search File

4.Goto main menu 5.Exit

Case 2 222

Delete the file and display submenu

Search the file and display submenu

**Case 3** 3

**Case 3** 3

**Case 4** 3

Go to main menu

Default : Display message Please enter correct input.

**Case 5** 3

End

Default : Display message Please enter correct input.

**User Story2**

**Steps:**

1. Start
2. Check if directory exist else create a new directory in the project
3. Print the welcome screen
4. Display the menu
5. Take user choice from menu and switch to user choice
6. If user choice 1 then display the files inside directory in ascending order
7. If user choice 2 then display the second menu
8. 1.Add: If user choice 1 then add the file as specified by user
9. 2. Delete: If user choice 2 then delete the file as specified by user
10. 3. Search: If user choice 3 then search the file as specified by user
11. 4. If user choice 4 then go to step 4
12. If user choice 5 then exit
13. Default statement: Please enter correct input
14. If user choice 3 then exit
15. Default statement: Please enter correct input