**LockedMe – Virtual Key for Repositories**

The code for this project is hosted at <https://github.com/aj-create-54/Lockers>

The project is developed by Anshika Jain.

**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.

## **Sprints planning and Task completion**

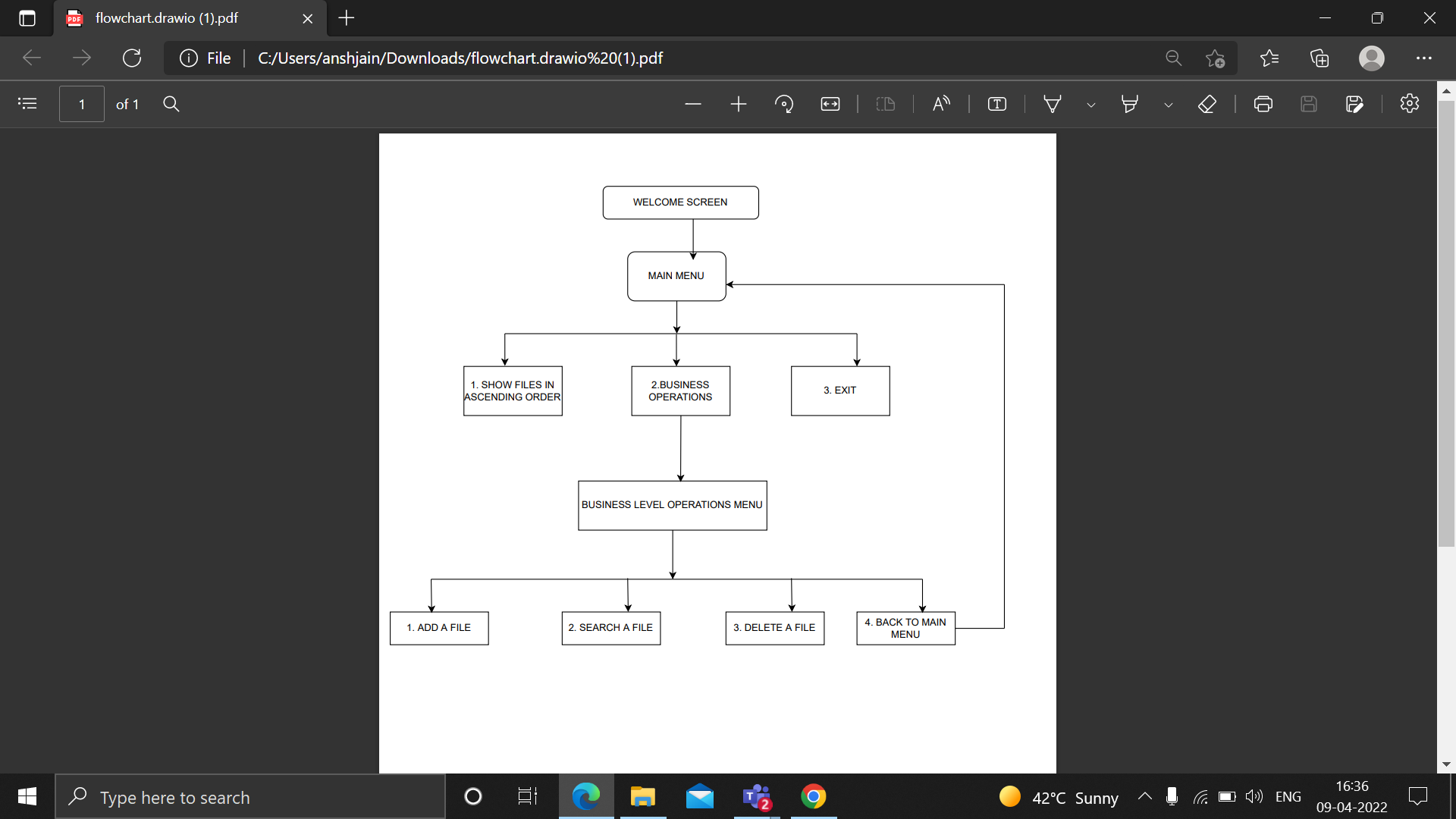
The project is planned to be completed in 1 sprint. Tasks assumed to be completed in the sprint are:

* Creating the flow of the application
* 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.
* Creating this specification document highlighting application capabilities, appearance, and user interactions.

## Core concepts used in the project

Collections framework, File Handling, Sorting, Flow Control, Recursion, Exception Handling.

**Flow of the Application**



**Program Hierarchy**

|  |  |
| --- | --- |
| **FILE** | **DESCRIPTION** |
| OutputFile.java | Entry point of the application contains main method. |
| ChooseOptions.java | Displays the menu options. |
| ShowFiles.java | Displays the list of files in the directory. |
| PerformingOperations.java | Performs business-level-operations, contains methods to add, delete and search for a file. |

## **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, an appropriate option needs to be selected.

1. The application can take any file/folder name as input. Even if the user wants to create nested folder structure, user can specify the relative path, and the application takes care of creating the required folder structure.

1. Users are also provided the option to write content if they want into the newly created file.

1. The application doesn’t restrict the user from specifying the exact filename to search/delete file/folder. They can specify the starting input, and the program searches all files/folder starting with the value and displays it. The user is then provided the option to select all files or to select a specific index to delete.

1. The application also allows the user to delete folders which are not empty.

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

1. When the option to retrieve files in ascending order is selected, the user is displayed with two options for viewing the files.

* 1. Ascending order of folders first which have files sorted in them,
  2. Ascending order of all files and folders inside the “main” folder.

1. 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.

## **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 the 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.
* Allowing user to append data to the file.