

Java FSD with Angular: Course End Project 1

1. Project Details

Project Title: LockedMe.com

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.

2. Developer Details

Name: Manoj M

Designation: Technical Trainee

Email: manm@teksystems.com

Phone: 9741626527

3. Project Background and 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. The application developed using Java programming language, Git and GitHub for version control, and Scrum for Agile methodology.

- 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.

4. Sprints Planned

- **Sprint 1:** Planning and Creating UI
- **Sprint 2:** Fetching Files from Directory
- **Sprint 3:** Creating and Deleting Files
- **Sprint 4:** Searching and Final Touch
- **Sprint 5:** Testing and Documentation

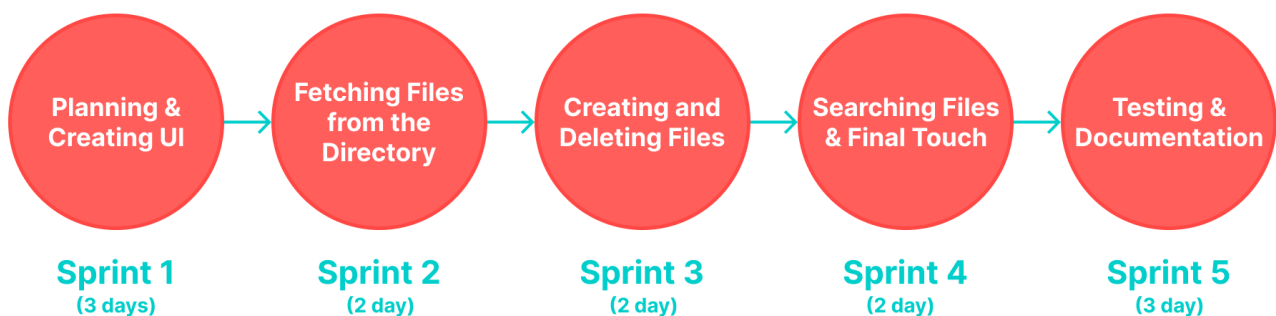


Fig: Sprints Flow diagram

- **Sprint 1: Planning and Creating Welcome Screen (Duration: 3 Days):**

Understanding the requirements and functionalities of the application, Setting up the development environment (Eclipse, Git, GitHub). Started developing welcome of the application.

- **Sprint 2: Fetching Files from Directory (Duration: 2 Day):**

Creating a class for doing operations on the directory, created a method to retrieve the filenames in ascending order and created a method to get directory path.

- **Sprint 3: Creating and Deleting Files: (Duration: 2 Day):**

Creating a class for all file operations, created a method to create and deletion a file from the directory.

- **Sprint 4: Searching and Final Touch (Duration: 2 Day):**

Created a method to search a file in the directory and Improving the user interface and adding exception handling if any missing before.

- **Sprint 5: Testing and Documentation (Duration: 3 Days):**

Testing an application and fixing any bugs if present and preparing the final documentation of the application.

5. Algorithms and Flowchart

Flowchart of overall application:

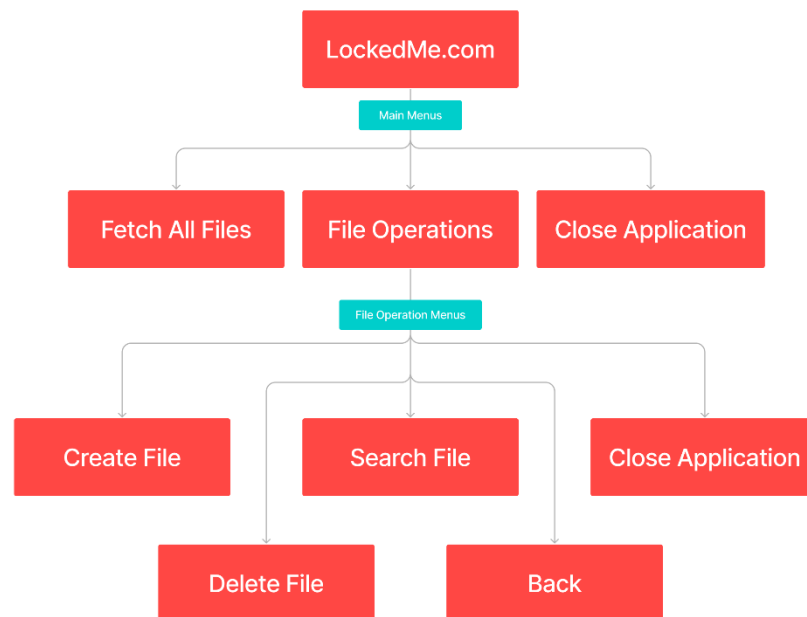


Fig: Flowchart

Algorithms:

1. Algorithm to retrieve all files:

- Get a path of the directory.
- Creating a File object using the directory path.
- Check whether the directory is present, if present get all the files using `listFiles()` method of File Class and store it in List.
- Sort a list of files using `sort()` method of File by passing Comparator.
- Return the sorted file list.

2. Algorithm to search file:

- Get a path of the directory and name of the file.
- Create a File object using the root directory path and user-specified file name.
- Check whether the file is present in the list of files.
- If present returns true else false.

3. Algorithm to retrieve create a file:

- Create a File object using the root directory path and user-specified file name.
- Check if the file already exists using `searchFile()` method of same class.
- If the file does not exist, create the file using `createNewFile()` method of File class.
- Print the status of the operation.

4. Algorithm to retrieve delete a file:

- Create a File object using the root directory path and user-specified file name.
- Check if the file already exists using searchFile() method of same class.
- If the file exists, delete the file using delete() method of File class.
- Print the status of the operation.

6. Core concepts used in the project

- **Collections:** Used to store and manipulate the files in the directory
- **File:** Used for creating and deleting the file.
- **Comparator:** Used to sort the filenames in the collection.
- **Exception:** Used to handle errors and invalid user input.

7. GitHub Repository

The source code for the LockedMe.com prototype can be found in the following GitHub repository: <https://github.com/Manoj-14/lockedMe>

8. Conclusion

The LockedMe.com prototype is a command-line application that allows users to interact with a directory of files. The prototype was developed using Java and implemented various concepts such as Collections, File, Exceptions, and Comparator. The application was developed in two sprints, with the first sprint focused on implementing the generic features and the second sprint focused on implementing the business-level operations. The application was successfully completed, and the source code was pushed to a GitHub repository for version control. Overall, the prototype demonstrates the ability to deliver a high-end quality product incrementally using an agile framework such as Scrum. And unique selling points of the application are it have a user-friendly interface, ability to perform basic file operations and it gives fast and effective performance.