**Project and Developer Details**

**Project Name: LockedMe.com File Systems**

**Project Version 1.0**

**Project Details:**

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.

It is a prototype of an application for file management, which generates a root directory by creating a bin folder in the file application. It gives user the capability to list files/folder present in this root directory as well as add and remove files from this root directory. It also gives the user a capability to search all the files with a particular name in the directory.

**Sprint Details for the project**

**No. of Sprints Planned:** 2

**Sprint 1 Details:**

In the first sprint, the basic structure of the application was developed, with the simple capability of being able to list files present in the root directory.

The application will have a three-tier structure. The presentation layer, the business validation layer and the data access layer.

1. The presentation layer is the console, and the void main of the application corresponds to the presentation layer. It is responsible for taking the inputs from the user as well as showing the outcome/output to the user. Thus, it is the layer which communicates with the user.
2. The Business Validation layer is responsible for validating the input from the user.
3. The data access layer processes this user input and performs the operation as per the user input.

This basic structure was created using multiple classes and interfaces. The logic to list the files present in the root directory was developed.

**Sprint 2 Details:**

The above application was enhanced to incorporate additional features like addition/ removal/ search of a file.

This included creating a submenu. And adding the above options. New methods had to be created for this purpose.

**Application Logic/Algorithm**

The Application has a main menu, and a submenu. The main menu gives you an option to list all the files in the directory, navigate to the sub-menu and exit the application.

Text

Description automatically generated with medium confidence

The sub-menu gives you an option to add a file, delete a file, search files from a directory.

A picture containing graphical user interface

Description automatically generated

Classes & Interface used to build the application:

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Name | Class/interface | Remarks |
| 1 | LockedmeDAO | Interface | Data access object interface for the application |
| 2. | LockedmeDAOImpl | Class | Class that implements the LockedmeDAO interface. This has the core logic of the program |
| 3 | LockedmeBO | Interface | Business Object interface for the application |
| 4 | LockedmeBOImpl | Class | Class that implements the LockedmeBO interface. This has the logic for business validation layer. |
| 5 | FileObj | Class | Use to create the basic file objects and store data corresponding to a file. It is the model class of the application |
| 6 | Main | Class | It is the presentation layer of the application it has the Void Main function. |

**Core concepts used in the project**

Java concepts implemented in this application:

* Classes
* Interface
* Inheritance
* Constructors
* LinkedList
* Constructors
* Comparable Interface
* Data Encapsulation (using private along with setters and getters)
* Exception Handling
* File class
* For, While, Switch Case, If-Else ladder, and other logic related functions.

**Screenshot of program:**

Text

Description automatically generated

**Links to the GitHub repository to verify the project completion**

<https://github.com/rahul-mendes/LockedMe>

**Conclusion**

The present program has been conceived as a prototype to prove the concept of a simple file system structure to the client. The USP of this project as it is right now, is that is a clear implementation of the core file management service and can be expanded or modified easily. The scope for improvements to this program is extremely huge and includes functionality for creating new directories and navigating between them, editing, moving and renaming the files, writing within the files, and implementing a security interface using passwords, to protect misuse or mismanagement of these files.