**SPRINTS**

This prototype project (LockedMe.com) will be delivered in

**Product Backlog**

Setting up Git and GitHub accounts to store and track project enhancements

To display a welcome screen

Display sub-menu options for file operation option from main menu

To add a user specified file to the application

To delete a user specified file from the application

To search a user specified file on the application

Option to return to main menu

**Release 1**

Sprint 1: 1 week

User story 1: Setting up Git and GitHub accounts to store and track project enhancements

Task 1: Set up a dedicated remote repository to store and track enhancements

Task 2: Set up a dedicated local repository to store and trace enhancements

User story 2: To display a welcome screen

Task 1: Show application name and developer details with print out statements.

Task 2: Print text to show 3 main menu options for user interaction.

Task 3: Prompt for user for input with Scanner object

Task 4: Write a switch block to handle main menu options

Task 5: Write simple print out statements for switch cases

Task 5: Loop main menu options, switch block and user input.

Task 6: Commit project to GitHub

Sprint 2: 1 week

User story 1: Display sub-menu options for file operation option from main menu

Task 1: Print text to show 4 sub-menu options for file operation

Task 2: Prompt user for input with Scanner object.

Task 3: Write a switch block to handle sub-menu options

Task 4: Write simple print out statements for switch cases

Task 5: Loop main menu options, switch block and user input.

Task 6: Create a local repository for tracking and storing LockedMe.java and FileOperations.java files

User story 2: To add a user specified file to the application

Task 1: Add prompt for user to type file name to be added.

Task 2: Implement file creation with java.io.File class.

Sprint 3: 1 week

User story 1: To delete a user specified file from the application

Task 1: Add prompt for user to type file name to be deleted.

Task 2: Implement file deletion with java.io.File class.

User story 2: To search a user specified file on the application

Task 1: Add prompt for user to type file name to be searched.

Task 2: Implement file search with java.io.File class.

User story 3: Option to return to main menu

Task 1: Implement break out from sub-menu to main menu

**Definition of Done**

Algorithm is complete

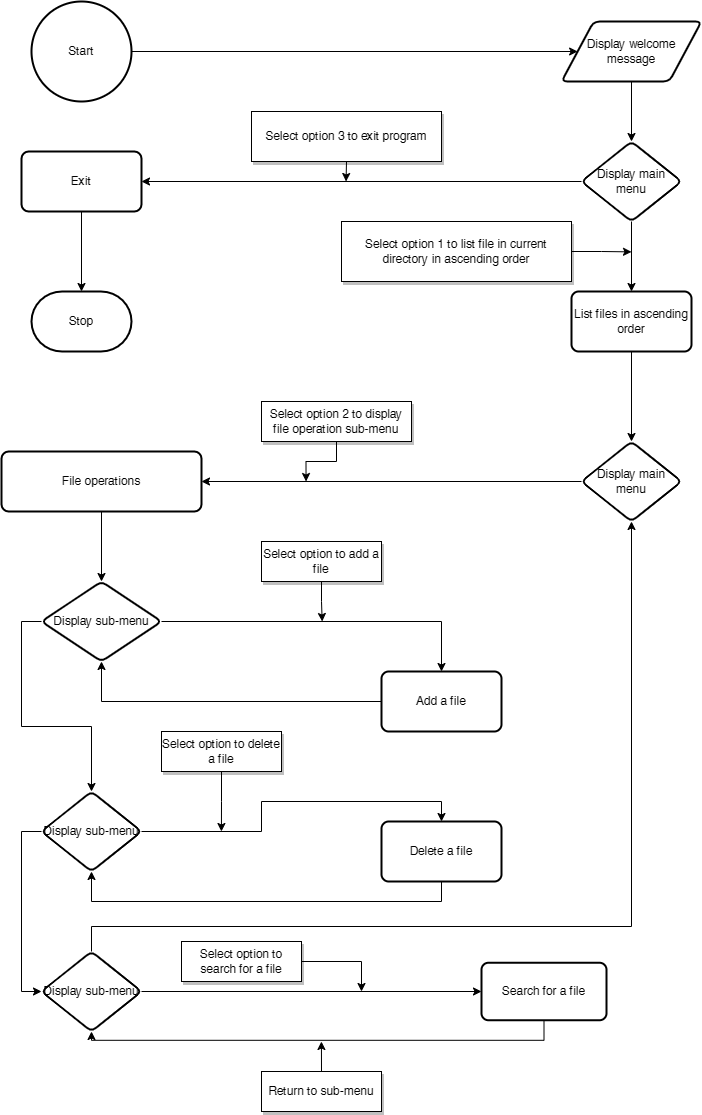
Documentation is complete

Testing is complete

Source code pushed to server

Code has been reviewed and Product owner has approved

**FLOW CHART**



**JAVA CORE CONCEPTS**

Encapsulation: LockedMe has private static property. Public static getSc() method to get scanner object.

Using control/loop blocks: if, else, while, enhanced for and switch

Data structures: variables(local, static and instance variables) and arrays

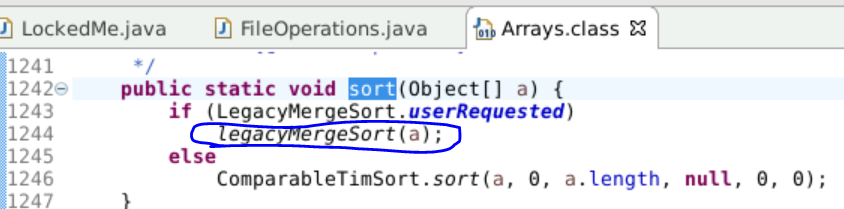
Exception handling: with try and catch blocks

Generic classes: using ArrayList<String> from FileOperations.java

**ALGORITHMS**

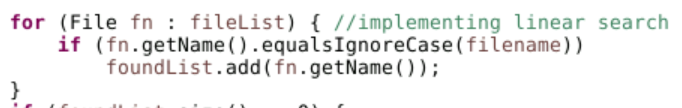
**Sorting Methods**

Merge sort( from java.util.Array.sort()



**Searching methods**

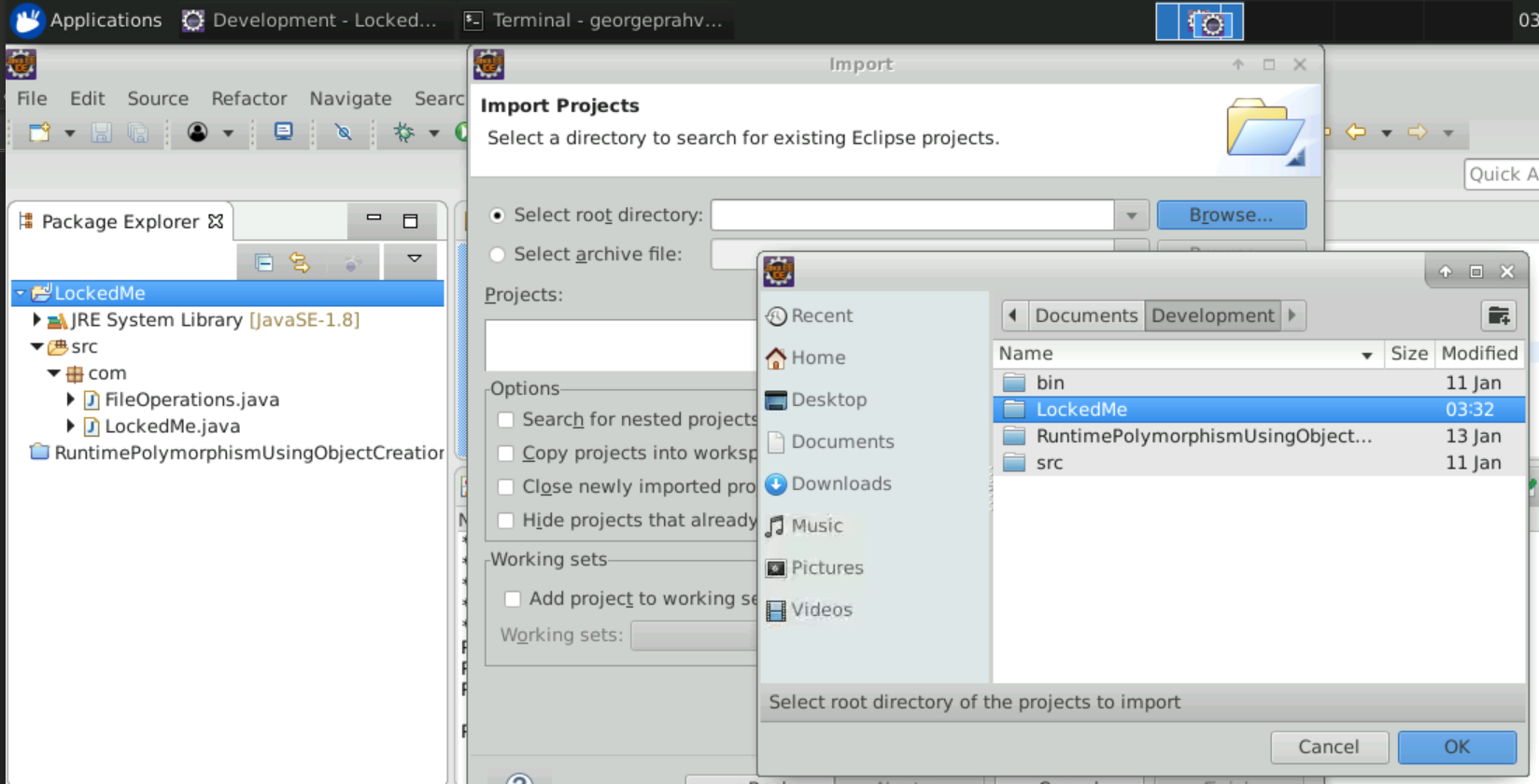
Linear search



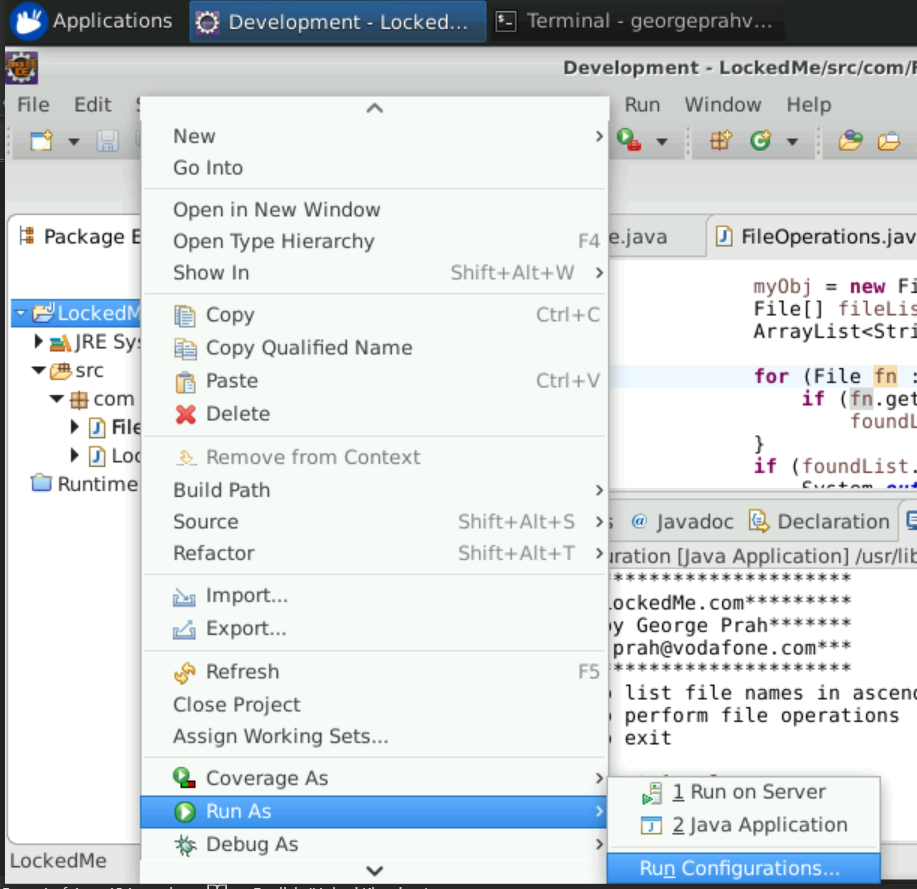
**How to run program**

**From Eclipse IDE**

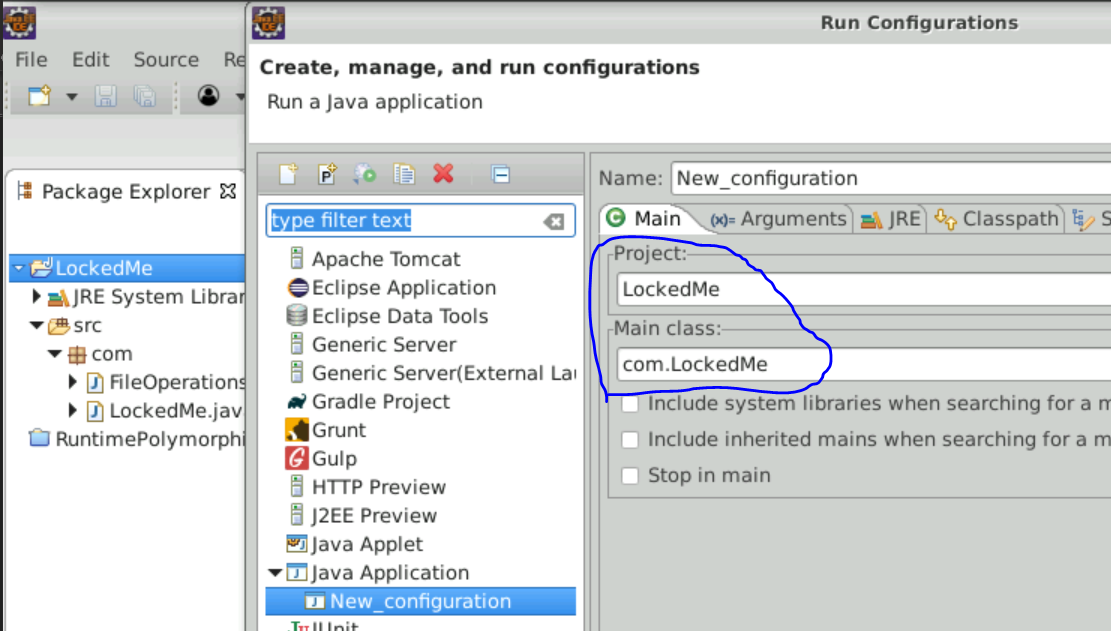
From File menu, select import to import existing Eclipse project and choose LockedMe project folder.

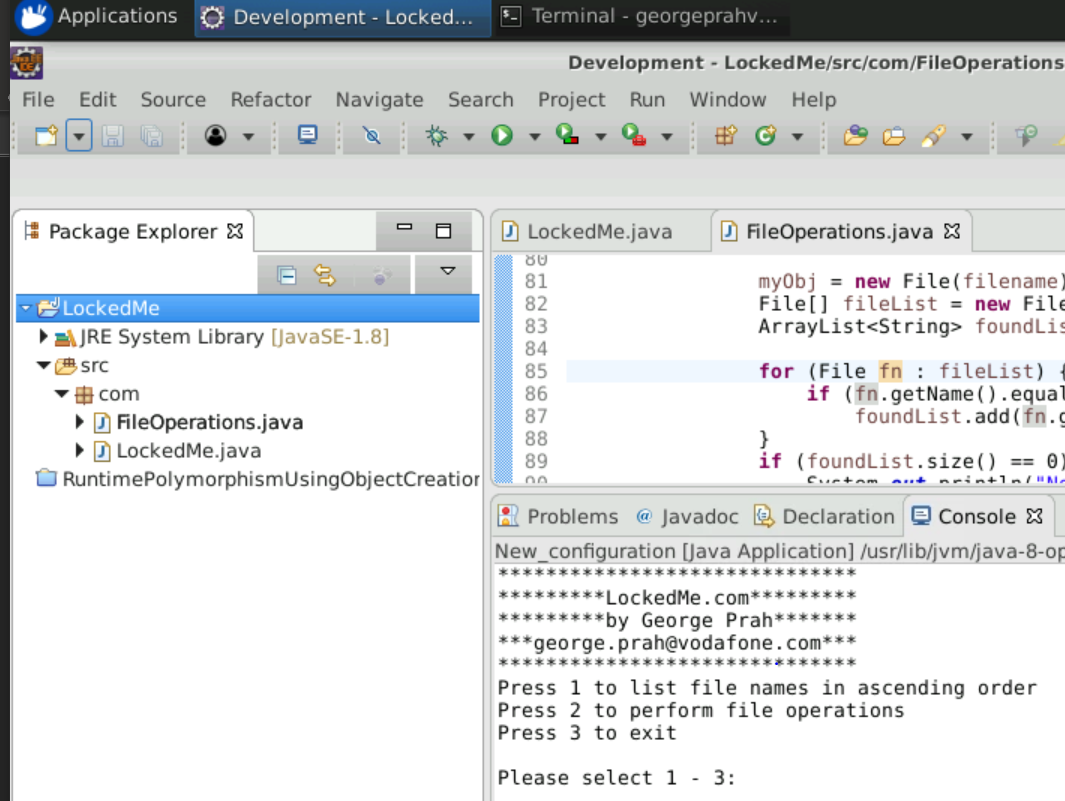


**Setting run configuration**



**Set main class com.LockedMe, apply and run to execute LockedMe project.**

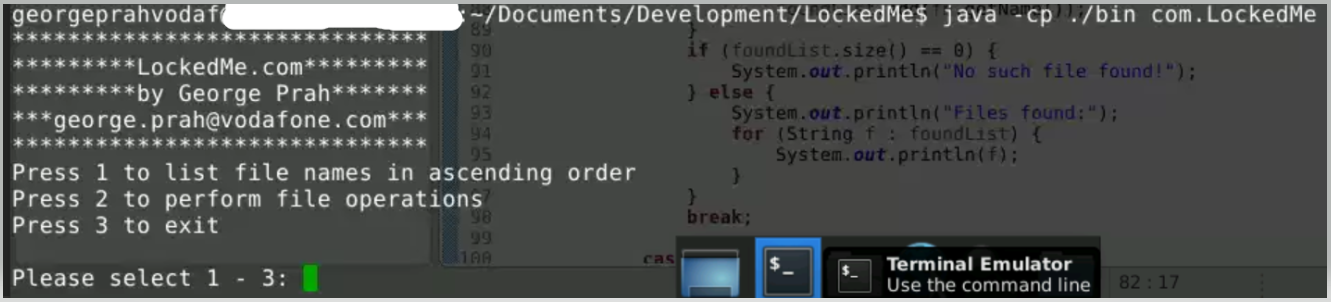




**From command line**

Navigate to LockedMe downloaded folder from GitHub.

Run java from within LockedMe folder as shown below.



GitHub repository

https://github.com/geprah/FullStackJavaDeveloper.git