LockedMe.com

17 September 2022 11:30 AM

Introduction:

A Java Application has been developed to digitize the products of Company Lockers Pvt. Ltd. This project is named as LockedMe.com and this prototype has been developed for the project of Company Lockers Pvt. Ltd. This can be presented to the relevant stakeholders for the budget approval.

The code for this project is available in GitHub: https://github.com/bharathnb1999/LockedMe.com_bharath

Concepts used for this application are:

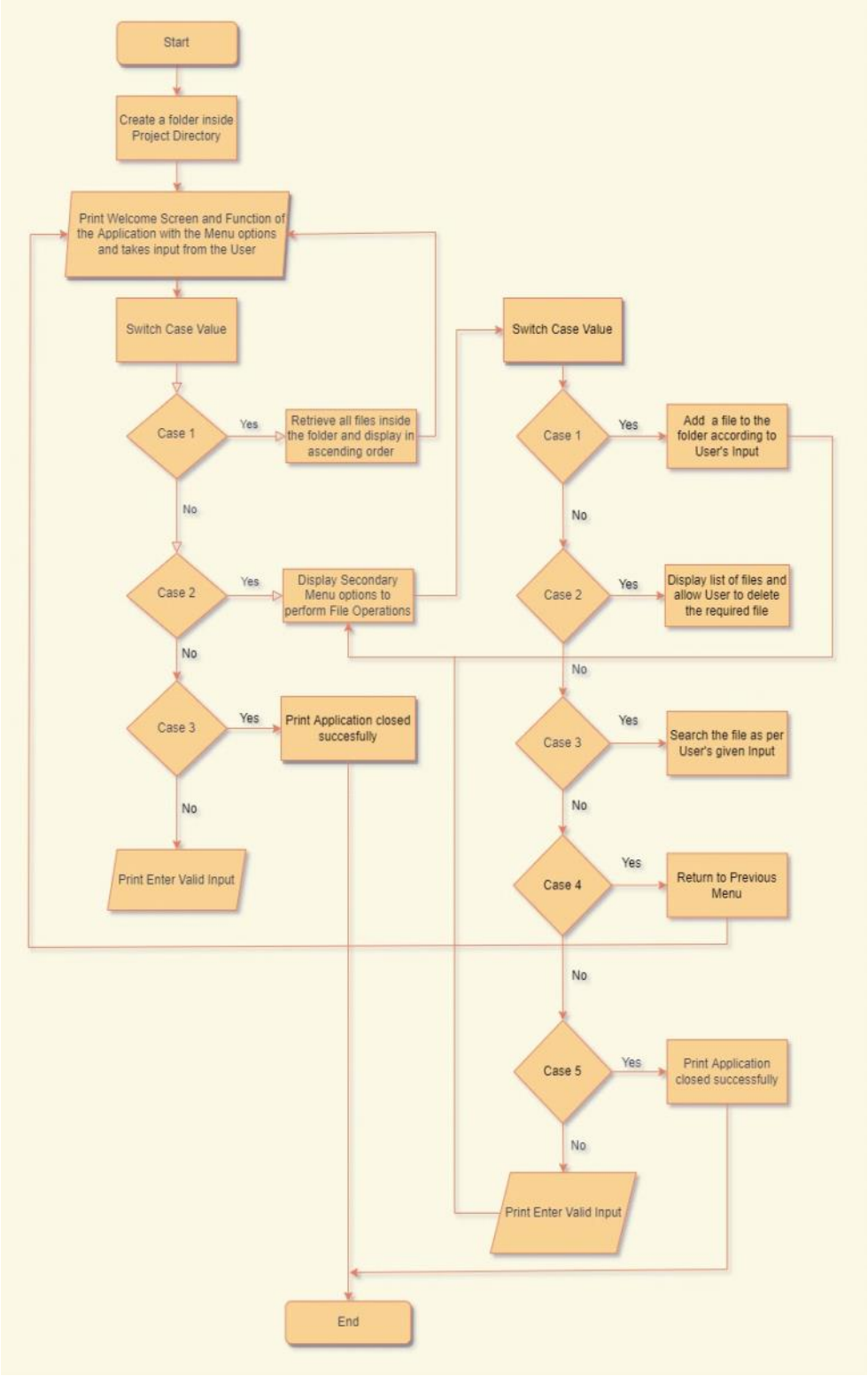
1. Data Structures with Searching and Sorting.
2. File Handling Operations.
3. Exception Handling.
4. Collection Framework.
5. Recursion.

The Java Application has been developed using 4 Classes:

1. Main class: mainlockme.java
2. Class for File Operations: fileops.java
3. Class for Handling the files: handleops.java
4. Class for displaying Menu Options: menuops.java

The Flow of this project is as below:

Flowchart:



Flowchart Link: https://drive.google.com/file/d/1VbuzvI9le13bNcLPkjjuMi4AuBRgK1ov/view?usp=sharing

The code of this project is as below:

mainlockme.java:

```
package packlockme;

public class mainlockme {

	public static void main(String[] args) {

		fileops.createMainFolderIfNotPresent("main");
```

```
                menuops.printWelcomeScreen("LockedMe", "Bharath N B");

                handleops.handleWelcomeScreenInput();
        }

}
```

fileops.java:

```
package packlockme;

import java.io.File;
import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;
import java.util.List;
import java.util.Scanner;
import java.util.stream.Collectors;
import java.util.stream.IntStream;

public class fileops {
        public static void createMainFolderIfNotPresent(String folderName) {
                File file = new File(folderName);
                if (!file.exists()) {
                        file.mkdirs();
                }
        }

        public static void displayAllFiles(String path) {
                fileops.createMainFolderIfNotPresent("main");
                System.out.println("Files with directory structure in ascending order:\n");

                List<String> filesListNames = fileops.listFilesInDirectory(path, 0, new ArrayList<String>());

                System.out.println("Files in ascending order:\n");
                Collections.sort(filesListNames);

                filesListNames.stream().forEach(System.out::println);
        }

        public static List<String> listFilesInDirectory(String path, int indentationCount, List<String> fileListNames) {
                File dir = new File(path);
                File[] files = dir.listFiles();
                List<File> filesList = Arrays.asList(files);

                Collections.sort(filesList);

                if (files != null && files.length > 0) {
                        for (File file : filesList) {

                                System.out.print(" ".repeat(indentationCount * 2));

                                if (file.isDirectory()) {
                                        System.out.println("`-- " + file.getName());


                                        fileListNames.add(file.getName());
                                        listFilesInDirectory(file.getAbsolutePath(), indentationCount + 1, fileListNames);
                                } else {
                                        System.out.println("|-- " + file.getName());
                                        fileListNames.add(file.getName());
                                }
                        }
                } else {
                        System.out.print(" ".repeat(indentationCount * 2));
                        System.out.println("|-- Empty Directory");
                }
                System.out.println();
                return fileListNames;
```

```
}

public static void createFile(String fileToAdd, Scanner sc) {
	fileops.createMainFolderIfNotPresent("main");
	Path pathToFile = Paths.get("./main/" + fileToAdd);
	try {
		Files.createDirectories(pathToFile.getParent());
		Files.createFile(pathToFile);
		System.out.println(fileToAdd + " created successfully");

		System.out.println("Would you like to add some content to the file? (Y/N)");
		String choice = sc.next().toLowerCase();

		sc.nextLine();
		if (choice.equals("y")) {
			System.out.println("\n\nInput content and press enter\n");
			String content = sc.nextLine();
			Files.write(pathToFile, content.getBytes());
			System.out.println("\nContent written to file " + fileToAdd);
			System.out.println("Content can be read using Notepad or Notepad++");
		}

	} catch (IOException e) {
		System.out.println("Failed to create file " + fileToAdd);
		System.out.println(e.getClass().getName());
	}
}

public static List<String> displayFileLocations(String fileName, String path) {
	List<String> fileListNames = new ArrayList<>();
	fileops.searchFileRecursively(path, fileName, fileListNames);

	if (fileListNames.isEmpty()) {
		System.out.println("\nCouldn't find any file with given file name \"" + fileName + "\" \n");
	} else {
		System.out.println("\n\nFound file at below location(s):");

		List<String> files = IntStream.range(0, fileListNames.size())
				.mapToObj(index -> (index + 1) + ": " + fileListNames.get(index)).collect(Collectors.toList());

		files.forEach(System.out::println);
	}

	return fileListNames;
}

public static void searchFileRecursively(String path, String fileName, List<String> fileListNames) {
	File dir = new File(path);
	File[] files = dir.listFiles();
	List<File> filesList = Arrays.asList(files);

	if (files != null && files.length > 0) {
		for (File file : filesList) {

			if (file.getName().startsWith(fileName)) {
				fileListNames.add(file.getAbsolutePath());
			}

			if (file.isDirectory()) {
				searchFileRecursively(file.getAbsolutePath(), fileName, fileListNames);
			}
		}
	}
}

public static void deleteFileRecursively(String path) {

	File currFile = new File(path);
	File[] files = currFile.listFiles();

	if (files != null && files.length > 0) {
		for (File file : files) {

			String fileName = file.getName() + " at " + file.getParent();
```

```
                        if (file.isDirectory()) {
                                deleteFileRecursively(file.getAbsolutePath());
                        }

                        if (file.delete()) {
                                System.out.println(fileName + " deleted successfully");
                        } else {
                                System.out.println("Failed to delete " + fileName);
                        }
                }
        }

        String currFileName = currFile.getName() + " at " + currFile.getParent();
        if (currFile.delete()) {
                System.out.println(currFileName + " deleted successfully");
        } else {
                System.out.println("Failed to delete " + currFileName);
        }
    }
}
```

handleops.java:

```
package packlockme;


import java.util.List;
import java.util.Scanner;

public class handleops {
    public static void handleWelcomeScreenInput() {
        boolean running = true;
        Scanner sc = new Scanner(System.in);
        do {
            try {
                menuops.displayMenu();
                int input = sc.nextInt();

                switch (input) {
                case 1:
                    fileops.displayAllFiles("main");
                    break;
                case 2:
                    handleops.handleFilemenuops();
                    break;
                case 3:
                    System.out.println("Application closed successfully.");
                    running = false;
                    sc.close();
                    System.exit(0);
                    break;
                default:
                    System.out.println("Please select a valid option from above.");
                }
            } catch (Exception e) {
                System.out.println(e.getClass().getName());
                handleWelcomeScreenInput();
            }
        } while (running == true);
    }

    public static void handleFilemenuops() {
        boolean running = true;
        Scanner sc = new Scanner(System.in);
        do {
            try {
                menuops.displayFilemenuops();
                fileops.createMainFolderIfNotPresent("main");

                int input = sc.nextInt();
                switch (input) {
                case 1:
```

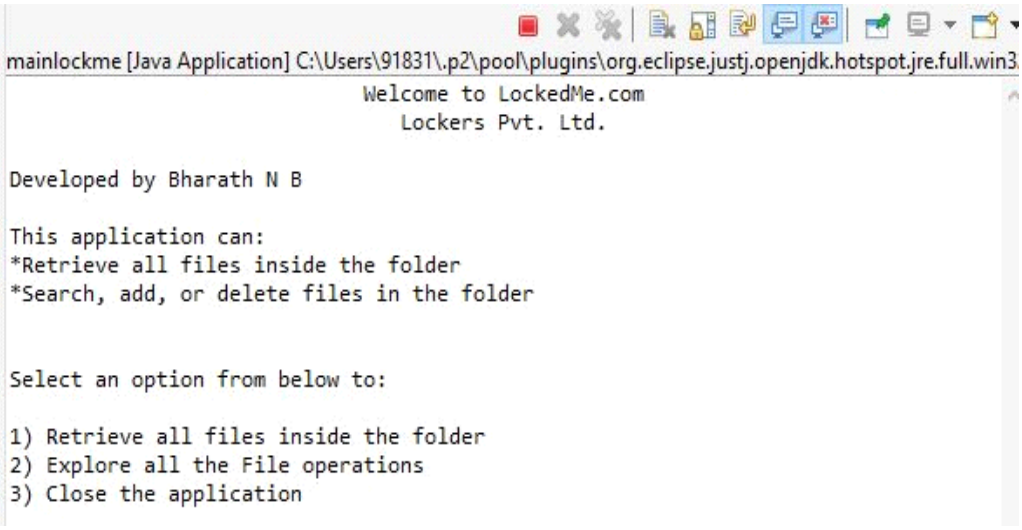
```
                        System.out.println("Enter the name of the file to be added to the folder");
                        String fileToAdd = sc.next();

                        fileops.createFile(fileToAdd, sc);

                        break;
                    case 2:

                        System.out.println("Enter the name of the file to be deleted from folder");
                        String fileToDelete = sc.next();

                        fileops.createMainFolderIfNotPresent("main");
                        List<String> filesToDelete = fileops.displayFileLocations(fileToDelete, "main");

                        String deletionPrompt = "\nSelect index of which file to delete?"
                                + "\n(Enter 0 if you want to delete all elements)";
                        System.out.println(deletionPrompt);

                        int idx = sc.nextInt();

                        if (idx != 0) {
                            fileops.deleteFileRecursively(filesToDelete.get(idx - 1));
                        } else {


                            for (String path : filesToDelete) {
                                fileops.deleteFileRecursively(path);
                            }
                        }


                        break;
                    case 3:

                        System.out.println("Enter the name of the file to be searched from folder");
                        String fileName = sc.next();

                        fileops.createMainFolderIfNotPresent("main");
                        fileops.displayFileLocations(fileName, "main");


                        break;
                    case 4:

                        return;
                    case 5:

                        System.out.println("Application closed successfully.");
                        running = false;
                        sc.close();
                        System.exit(0);
                    default:
                        System.out.println("Please select a valid option from above.");
                    }
                } catch (Exception e) {
                    System.out.println(e.getClass().getName());
                    handleFilemenuops();
                }
            } while (running == true);
        }
}
```

menuops.java:

```
package packlockme;

public class menuops {

    public static void printWelcomeScreen(String appName, String developerName) {
        String companyDetails = String.format("Welcome to %s.com \n" + "Lockers Pvt. Ltd. \n" + "\nDeveloped by %s\n", appName, developerName);
        String appFunction = "This application can: \n"
                + "*Retrieve all files inside the folder\n"
                + "*Search, add, or delete files in the folder\n";
```

```
            System.out.println(companyDetails);

            System.out.println(appFunction);
        }

        public static void displayMenu() {
            String menu = "\nSelect an option from below to: \n\n"
                        + "1) Retrieve all files inside the folder\n" + "2) Explore all the File operations\n"
                        + "3) Close the application\n";
            System.out.println(menu);

        }

        public static void displayFilemenuops() {
            String fileMenu = "\n\nSelect any option from below to:\n\n"
                        + "1) Add a file to the folder\n" + "2) Delete a file from the folder\n"
                        + "3) Search for a file present inside the folder\n" + "4) Show Previous Menu\n" + "5) Close the application\n";

            System.out.println(fileMenu);
        }

}
```

The initial output of the Java Application with Welcome Screen:



Output after selecting option 1:

```
mainlockme [Java Application] C:\Users\91831\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre

Select an option from below to:

1) Retrieve all files inside the folder
2) Explore all the File operations
3) Close the application

1
Files with directory structure in ascending order:

`-- bharath
  |-- addthroughoutput.txt
  |-- bharathtextfile.txt
  |-- output.txt
  |-- sample.txt

|-- New.txt
|-- new1.txt

Files in ascending order:

New.txt
addthroughoutput.txt
bharath
bharathtextfile.txt
new1.txt
output.txt
sample.txt

Select an option from below to:

1) Retrieve all files inside the folder
2) Explore all the File operations
3) Close the application
```

Output after selecting option 2:

```
Select an option from below to:

1) Retrieve all files inside the folder
2) Explore all the File operations
3) Close the application

2


Select any option from below to:

1) Add a file to the folder
2) Delete a file from the folder
3) Search for a file present inside the folder
4) Show Previous Menu
5) Close the application
```

Output after selecting option 3:

```
Select an option from below to:

1) Retrieve all files inside the folder
2) Explore all the File operations
3) Close the application

3
Application closed successfully.
```

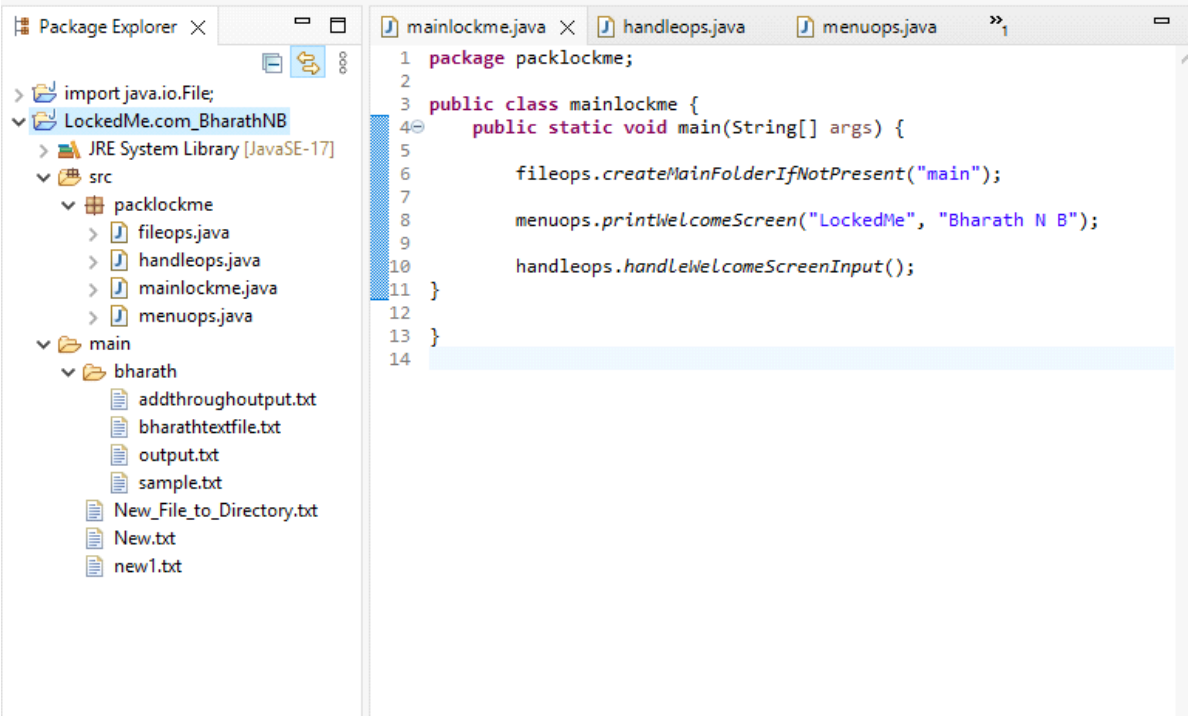
After selecting Option 2 from main menu, it gives the output of secondary menu to explore all the file operations included in this project as shown below:

```
Select any option from below to:

1) Add a file to the folder
2) Delete a file from the folder
3) Search for a file present inside the folder
4) Show Previous Menu
5) Close the application
```

User can add a file by selecting option 1 from Sub Menu as shown below:

```
Select any option from below to:

1) Add a file to the folder
2) Delete a file from the folder
3) Search for a file present inside the folder
4) Show Previous Menu
5) Close the application

1
Enter the name of the file to be added to the folder
New_File_to_Directory.txt
New_File_to_Directory.txt created successfully
Would you like to add some content to the file? (Y/N)
y


Input content and press enter

This is the new sample text file added to directory

Content written to file New_File_to_Directory.txt
Content can be read using Notepad or Notepad++
```

New File has been added successfully:



User can delete a file by selecting the option 2 from Sub Menu as shown below:

```
Select any option from below to:

1) Add a file to the folder
2) Delete a file from the folder
3) Search for a file present inside the folder
4) Show Previous Menu
5) Close the application

2
Enter the name of the file to be deleted from folder
New_File_to_directory.txt


Found file at below location(s):
1: C:\Users\91831\eclipse-workspace\VirtualKeyBharath\main\New_File_to_directory.txt

Select index of which file to delete?
(Enter 0 if you want to delete all elements)
0
New_File_to_directory.txt at C:\Users\91831\eclipse-workspace\VirtualKeyBharath\main deleted successfully
```

User can search for a file by selecting option 3 from Sub Menu by giving the correct filename or part of the filename which is present inside the folder as shown below:

```
Select any option from below to:

1) Add a file to the folder
2) Delete a file from the folder
3) Search for a file present inside the folder
4) Show Previous Menu
5) Close the application

3
Enter the name of the file to be searched from folder
new file


Found file at below location(s):
1: C:\Users\91831\eclipse-workspace\VirtualKeyBharath\main\new1.txt
```

User can go back to the previous menu by selecting the option 4 from the Sub Menu as shown below:

```
Select any option from below to:

1) Add a file to the folder
2) Delete a file from the folder
3) Search for a file present inside the folder
4) Show Previous Menu
5) Close the application

4

Select an option from below to:

1) Retrieve all files inside the folder
2) Explore all the File operations
3) Close the application
```

User can close the application by selecting option 5 from the Sub Menu as shown below:

```
Select any option from below to:

1) Add a file to the folder
2) Delete a file from the folder
3) Search for a file present inside the folder
4) Show Previous Menu
5) Close the application

5
Application closed successfully.
```

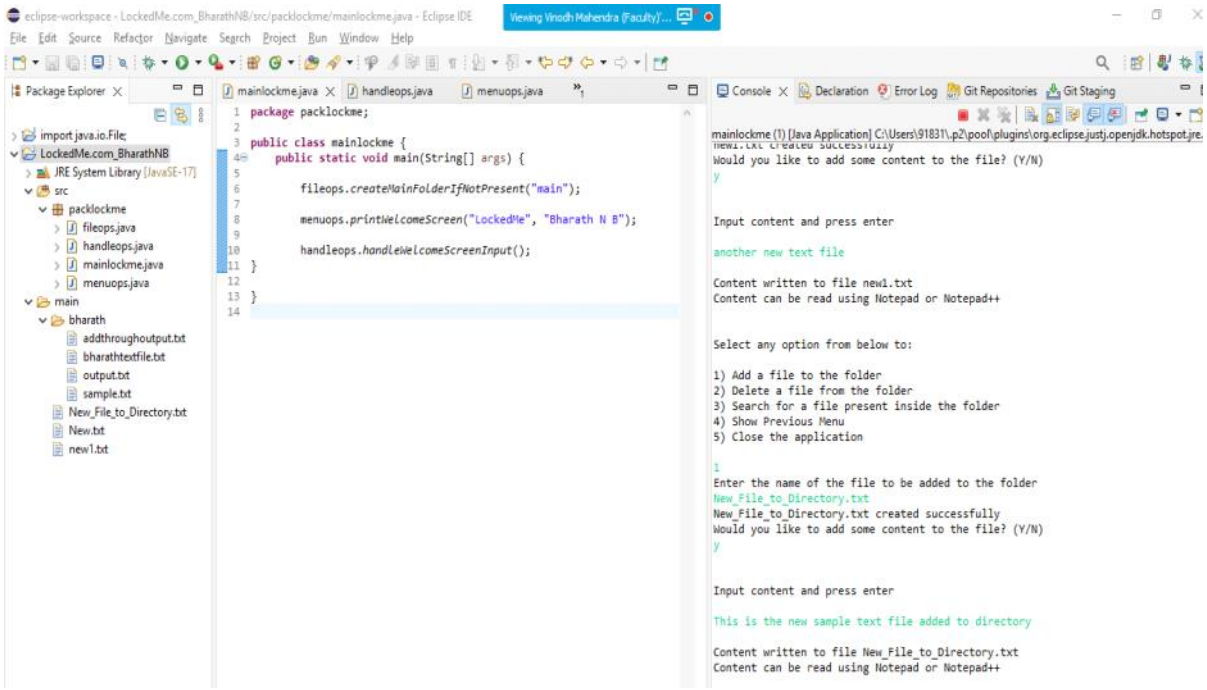
The application asks to enter a valid input if an invalid input is given as shown below:

```
                          Welcome to LockedMe.com
                             Lockers Pvt. Ltd.

Developed by Bharath N B

This application can:
*Retrieve all files inside the folder
*Search, add, or delete files in the folder


Select an option from below to:

1) Retrieve all files inside the folder
2) Explore all the File operations
3) Close the application

4
Please select a valid option from above.

Select an option from below to:

1) Retrieve all files inside the folder
2) Explore all the File operations
3) Close the application

Select an option from below to:

1) Retrieve all files inside the folder
2) Explore all the File operations
3) Close the application

2


Select any option from below to:

1) Add a file to the folder
2) Delete a file from the folder
3) Search for a file present inside the folder
4) Show Previous Menu
5) Close the application

6
Please select a valid option from above.


Select any option from below to:

1) Add a file to the folder
2) Delete a file from the folder
3) Search for a file present inside the folder
4) Show Previous Menu
5) Close the application
```

The application gives the following output if the file is not found in the folder:

```
Select an option from below to:

1) Retrieve all files inside the folder
2) Explore all the File operations
3) Close the application

2


Select any option from below to:

1) Add a file to the folder
2) Delete a file from the folder
3) Search for a file present inside the folder
4) Show Previous Menu
5) Close the application

2
Enter the name of the file to be deleted from folder
hf

Couldn't find any file with given file name "hf"
```

Snap from an Eclipse Workspace:



Snap of GitHub repository in which the current project has been uploaded:

Link: https://github.com/bharathnb1999/LockedMe.com_bharath