

LOCKEDME.COM

PROJECT SPECIFICATION AND SCRUM DETAILS

APPLICATION SCREENSHOT

➤ REGISTRATION LOGIN PAGE

```
***** WELCOME TO LockedMe.com *****
----- Your personal Digital LOCKER -----
=====
1 .Registration
2 .Login
|
=====
***** WELCOME TO REGISTRATION PAGE *****
=====
Enter Username :
BHAGYASHREE
Enter Password :
123
Access Granted! Welcome!
```

➤ MAIN MENU APPLICATION

```
----- WELCOME Main Menu LockedMe.com-----
*****
.Retrieve all files
.create a new file
.Delete a file
.Search a file
.Exit
*****
-----ENTER YOUR CHOICE-----
*****
```

INPUT OUTPUT SCREEN SHOT

```
FileManager.java  LockedMeProject.java X
1 package com.lockedme;
2 import java.io.File;
3 import java.util.ArrayList;
4 import java.util.List;
5 import java.util.Scanner;
6
7 public class LockedMeProject
8 {
9     static final String folderpath="D:\\MYPhase1OppsProject\\LockedMeFiles";
10
11     public static void main(String[] args)
12     {
13         //registration welcome screen
14         registrationLoginMainMenu();
15
16         int proceed=1;
```

Markers Properties Servers Snippets Console X

<terminated> LockedMeProject (Java Application) C:\Users\Ganesh\p2\pool\plugins\org.edipse.j...

```
=====
***** WELCOME TO LockedMe.com *****
----- Your personal Digital LOCKER -----
=====
1 .Registration
2 .Login
=====
***** WELCOME TO REGISTRATION PAGE *****
```

1.0 GENERAL INFORMATION

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.

APPLICATION NAME : LOCKEDME.COM
DEVELOPER GUIDE : MANALI DUABAL
DOCUMENT PURPOSE :APPLICATION SCREENSHOT LOCKEDME.COM

VERSION HISTORY

AUTHOR	BHAGYASHREE SUBHASH PALVE
PURPOSE	SCREENSHOT OF THE APPLICATION
DATE	26 TH NOV 2021
VERSION	2021 ECLIPSE 0.9

PROJECT GITHUB LINK :

REPOSITORY NAME : BHAGYASHREE – PALVE
GITHUB LINK :

Table of Contents

1.0 GENERAL INFORMATION

APPLICATION SCREENSHOT	2
APPLICANT NAME	
GIT HUB LINK	3

2.0 SPRINT PLANNING AND TASK COMPLETION

- i. REGISTRATION LOGIN
- ii. MAIN MENU
- iii. RETRIEVE ALL FILE
- iv. CREATE A NEW FILE
- v. DELETE A FILE
- vi. SEARCH A FILE
- vii. SYSTEM EXIT
- viii. TESTING

DEPLOYMENT

3.0 FUNCTIONAL REQUIREMENT

4.0 TECHNICAL REQUIREMENT

5.0 CORE CONCEPT USE IN PROJECT

6.0 DEMONSTRATE THE WORKING OF THE APPLICATION

8.0 UNIQUE SELLING POINT

9.0 CONCLUSION

10.0 APPENDIX

2.0 SPRINT PLANNING AND TASK COMPLETION :

SPRINT NUMBER	MODULES
1	USER REGISTER AND LOGIN RETRIEVE ALL FILES CREATE A NEW FILE
2	DELETE A FILE SEARCH A FILE TESTTING DEPLOYMENT

3.0 FUNCTIONAL REQUIREMENT :

a. Generic features and three operations:

- **Retrieving the file names in an ascending order**

i. RETRIEVE FILE :

- Retrieve a file is performed to get the all files in a folder path if it selected
- We use the **getAllFiles()**method to operate the file .
- It uses array list to display the list of file name in acesnding order.

- We can use the scanner class object to get the whole information take a user input

b. Modules in the project

- I. Registration Menu
- II. Login Display Main Menu
- III. Create New File
- IV. Delete File
- V. Search File
- VI. Exit

I .Registration Menu:

- When user visits to the page, show the welcome Page and provide two options such as registration/Login.
- If user choose registration, ask for username, password and save the data into a database file.
- Input data should be user input.
- Use a **registrationLoginMainMenu()** method .

ii. Login Display main menu:

- When user choose login, ask the username to provide. Username should be valid if it already exists in the database file.
- Using a **displayMainMenu()** method .

III. CREATE NEW FILE :

- **Create a File** operation is performed to create a new fileWe use the **createFile()** method of file. The
- **CreateFile()** method returns true when it successfully creates a new file
- IT take add content store it them used add() method then to save the content and file in folder .we use list of arrays in folder

- We can use the scanner class object to get the whole information take a user input .

IV .DELETE FILE :

- we can perform on a file is "**deleting a file**".
- In order to delete a file, we will use the **deleteFile()** method of the file.
- We using the boolean object **to delete the file** (is Deleted object because for deleting a file it use return true or false in loop statement

V.SEARCH FILE :

- we can perform on a file is "**searching a file in a given folder path**".
- In order to search a file, we will use the **searchFile()** method of the file.
- We using the boolean object **to search the file** (is found) because for deleting a file it use.
- return true or false in loop statement .and it clse the class FILE exists() method .

vi .SYSTEM EXIT :

When user choice nunmer 5 and end of the application then
method use the program in System.exit(0) method after automatic get close
the program

VII. Testing:

Test 1: PASS (LOGIN WELCOME SCREEN)

- Test first process Run the code java application in eclipse get output display welcome screen REGISTRATION LOGIN get input username and password then display message authentication access.

Test 2: PASS (DISPLAY MAIN MENU SCREEN)

- Test debug the code Next process display the output screen DISPLAY MAIN MENU SCREEN .

Test 3: PASS (RETRIEVE ALL FILE)

- TEST debug the code and user choice the any number 1 shows the all file name and repeat again showing main menu screen.

Test 4: pass (CREATE A NEW ADD CONTENT AND SAVE THE FILE)

- TEST debug the code and user choice the number 2 add a file name and add content and display message show the file save data successfully and either repeat showing main menu screen .

Test 5: PASS (DELETE A FILE)

- TEST debug the code and user choice the number 3 and add a file name delete the file in folder and display the message on screen file deleted after repeat showing again main menu screen.

Test 6: PASS (SEARCH A FILE)

- TEST debug the code and user choice the number 3 and add a file name search the file in folder and display the message on screen file present in the folder after repeat showing again main menu screen exit application the by default .

VIII. DEPLOYMENT:

- INSTALL JDK 8.0 VERSION
- INSTALL ECLIPSE AND DEBUG THE PROJECT CODE JAVA APPLICATION

4.0 TECHNICAL REQUIREMENT :

a.Generic features and three operations (Retrieving the file names in an ascending order)

- I. Provide the path where the files are stored.

folderpath="D:\\MYPhase1OppsProject\\LockedMeFiles";

```
public class LockedMeProject
{
    static final String folderpath="D:\\MYPhase1OppsProject\\LockedMeFiles";

    public static void main(String[] args)
    {//registration welcome screen
        registrationLoginMainMenu();
        int proceed=1;
        do
        {
            //Variable declaration
            Scanner obj = new Scanner(System.in);
            int ch ;
//Menu
            displayMainMenu();

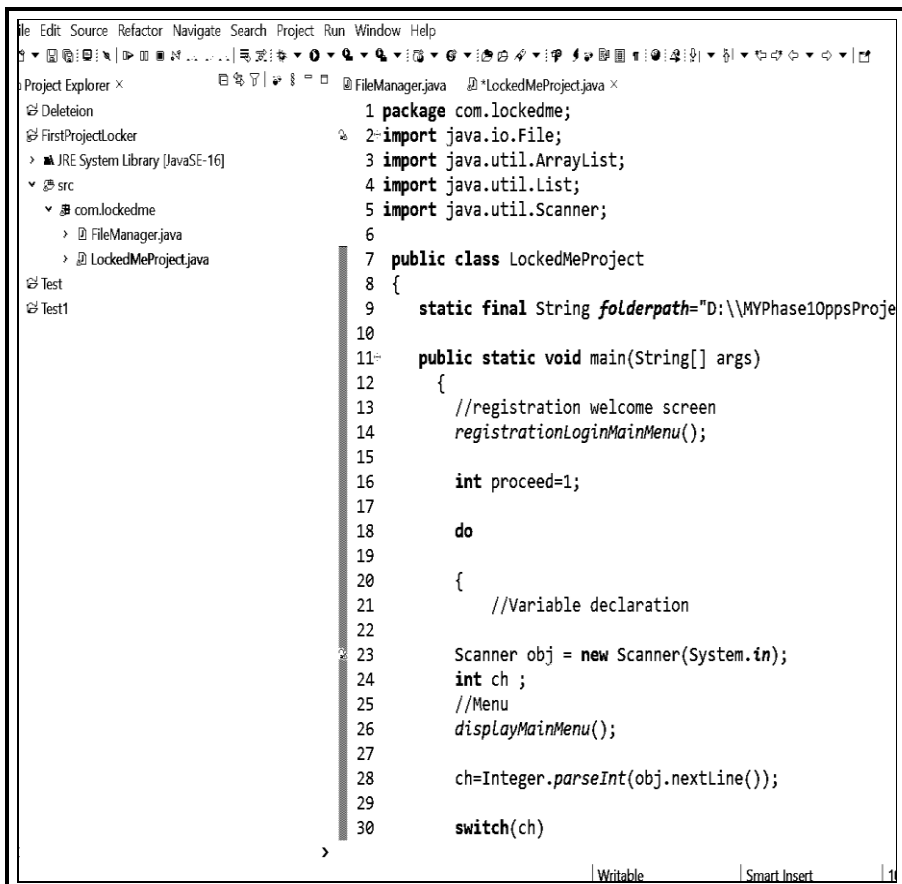
            ch=Integer.parseInt(obj.nextLine());
            switch(ch)
            {
                case 1 : getAllFiles();
                    break;
                case 2 : createFiles();
                    break;
                case 3 : deleteFile();
                    break;
                case 4 : searchFile();
                    break;
                case 5 : System.exit(0);
                    break;
                default :System.out.println("invalid operation");
                    break; }} while(proceed>0);}
```

II. Store the list of file names in a list.

```
//Save and store the content into the file
boolean isSaved = FileManager.createFiles(folderpath, fileName,
content);

if(isSaved)
{
System.out.println("File and data.saved successfully");
}
else
{
System.out.println("some error ocured please contact
admin@meg.com ");
}
//close Scanner object
//obj.close();}
```

PROJECT FOLDER SCREENSHOT



III RETRIEVETHE ALL FILE INPUT CODE

```
// RETRIVE ALL FILE NAME
public static void getAllFiles()
{
//GET THE FILE NAME
List<String> fileNames = FileManager.getAllFiles(folderpath);

for(String f:fileNames)
System.out.println(f);
}
```

OUTPUT:

```
----- WELCOME Main Menu LockedMe.com-----

*****

.Retrive all files
.create a new file
.Delete a file
.Search a file
.Exit
*****

-----ENTER YOUR CHOICE-----

*****
```

b. Business-level operations:

I. Registration Menu input code :

```
public static void registrationLoginMainMenu()
{
    String Username;
    String Password;
    Password = "123";
    Username = "BHAGYASHREE";
    System.out.println(" =====\n");
    System.out.println("***** WELCOME TO LockedMe.com *****\n");
    System.out.println("----- Your personal Digital LOCKER -----\n");
    System.out.println(" =====\n");
    System.out.println(" 1 .Registration \n" );
    System.out.println(" 2 .Login \n" );
    System.out.println(" =====\n");
    System.out.println("***** WELCOME TO REGISTRATION PAGE *****\n");
    System.out.println(" =====\n");

    Scanner input1 = new Scanner(System.in);
    System.out.println("Enter Username : ");
    String username = input1.next();

    Scanner input2 = new Scanner(System.in);
    System.out.println("Enter Password : ");
    String password = input2.next();

    if (username.equals(Username) && password.equals(Password))
    {
        System.out.println("Access Granted! Welcome!");
    }

    else if (username.equals(Username))
    {
        System.out.println("Invalid Password!");
    }
    else if (password.equals(Password))
    {
        System.out.println("Invalid Username!\n");
    }
}
```

OUTPUT :

```
***** WELCOME TO LockedMe.com *****
----- Your personal Digital LOCKER -----
=====
1 .Registration
2 .Login
|
=====
***** WELCOME TO REGISTRATION PAGE *****
=====

Enter Username :
BHAGYASHREE
Enter Password :
123
Access Granted! Welcome!
```

Display Main Menu input code

```
public static void displayMainMenu()  
{  
    System.out.println("*****");  
    System.out.println("----- WELCOME Main Menu LockedMe.com-----\n");  
    System.out.println("*****");  
    System.out.println("1.Retrieve all files");  
    System.out.println("2.create a new file");  
    System.out.println("3.Delete a file");  
    System.out.println("4.Search a file");  
    System.out.println("5.Exit");  
    System.out.println("*****");  
    System.out.println(" -----ENTER YOUR CHOICE-----\n");  
    System.out.println("*****");  
}
```

OUTPUT :

```
----- WELCOME Main Menu LockedMe.com-----  
  
*****  
.Retrieve all files  
.create a new file  
.Delete a file  
.Search a file  
.Exit  
*****  
  
-----ENTER YOUR CHOICE-----  
  
*****
```

||| . Create New File input code

```
//CREATE ALL FILE NAMES
public static void createFiles()
{
    //Variable Declaration
    Scanner obj = new Scanner(System.in);
    String fileName;
    int linesCount;
    List<String> content = new ArrayList<String>();

    //Read file name from user
    System.out.println("Enter The FileName:");
    fileName = obj.nextLine();

    //Read number of lines from user
    System.out.println("Enter how many lines in the file:");
    linesCount = Integer.parseInt(obj.nextLine());

    //Read Lines from user
    for(int i=1; i<=linesCount; i++)
    {
        System.out.println("Enter line" +i+ ":");
        content.add(obj.nextLine());
    }

    //Save and store the content into the file
    boolean isSaved = FileManager.createFiles(folderpath, fileName, content);

    if(isSaved)
    {
        System.out.println("File and data.saved successfully");
    }
    else
    {
        System.out.println("some error occured please contact admin@meg.com ");
    }
    //close Scanner object
}
```

OUTPUT :

```
2
Enter The FileName:
test.txt
Enter how many lines in the file:
1
Enter line1:
GOOD EXAM GOING
File and data.saved successfully
*****
----- WELCOME Main Menu LockedMe.com-----
*****
1.Retrive all files
2.create a new file
3.Delete a file
4.Search a file
5.Exit
*****
-----ENTER YOUR CHOICE-----
*****
```

Iv . Delete File input code Search File

```
/**
 * THIS METHOD IS USED FOR DELETE THE FILE
 */
//DELETE THE FILE
public static void deleteFile()
{
    //code for deleting a file
    String fileName;

    Scanner obj = new Scanner(System.in);
    System.out.println("ENTER FILE NAME TO BE DELETED: ");
    fileName = obj.nextLine();
    boolean isDeleted = FileManager.deleteFile(folderpath ,fileName);
    {
        if(isDeleted)
        {
            System.out.println("FILE DELETED SUCCESSFULLY:");
        }
        else
        {
            System.out.println("EITHER FILE NOT THERE OR SOME ACCESS ISSUE: ");
        }
    }
    //close Scanner object
    //obj.close();
}
```

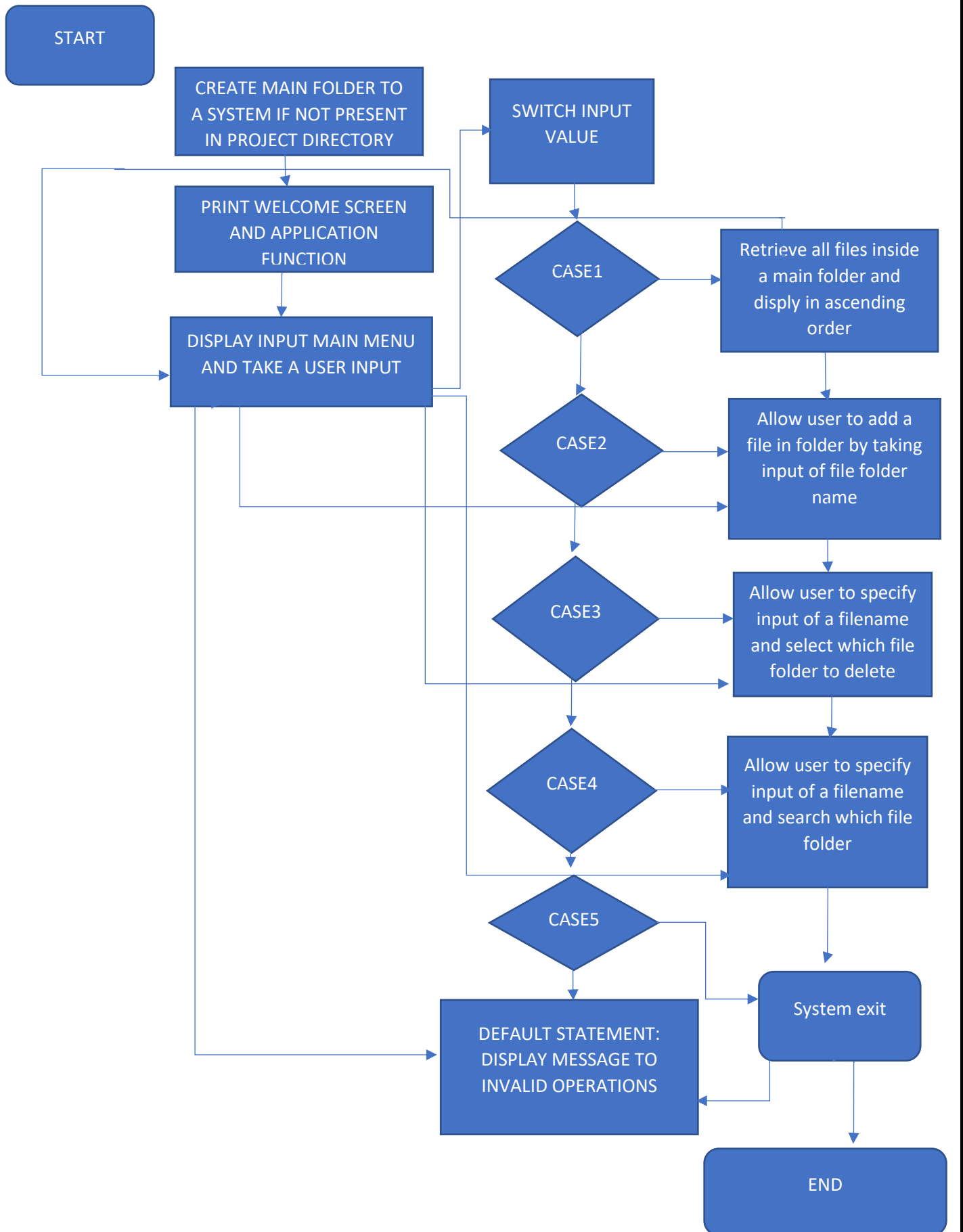
OUTPUT :

```
*****
-----ENTER YOUR CHOICE-----
*****
3
ENTER FILE NAME TO BE DELETED:
JAVA.txt
FILE DELETED SUCCESSFULLY:
*****
----- WELCOME Main Menu LockedMe.com-----
*****
1.Retrive all files
2.create a new file
3.Delete a file
4.Search a file
5.Exit
*****
-----ENTER YOUR CHOICE-----
```

4.0 CORE CONCEPTS USED IN THE PROJECT

- **FILE HANDLING**
- **EXCEPTION HANDLING**
- **NAMING STANDARDS**
- **MODULARITY**
- **OBJECT ORIENTED PROGRAMMING**
- **COLLECTIONS**
- **CONTROL STRUCTURE**
- **DATA STRUCTURE**

5.0 FLOW OF THE APPLICATION



6.0 DEMONSTRATE THE WORKING OF THE APPLICATION :

Step 1: Creating a new project in Eclipse

- Open Eclipse
- Go to File -> New -> Project -> Java Project -> Next.
- Type in any project name and click on "Finish."
- Select your project and go to File -> New -> Class.
- Enter LockedMe in any class name, check the checkbox "public static void main(String[] args)", and click on "Finish."

➤ STEP 1 :RETRIEVE ALL FILE AND CREATE NEW FILE ADD CONTENT AND SAVE IT FILE PROGRAM SOURCE CODE INPUT:

```
FileManager.java  *LockedMeProject.java ×
106 // RETRIVE ALL FILE NAME
107 public static void getAllFiles()
108 {
109 //GET THE FILE NAME
110 List<String> fileNames = FileManager.getAllFiles(folderpath);for(String f:fileNames)
111 System.out.println(f);
112 }
113 /**
114  * THIS METHOD IS USED FOR CREATE THE FILES
115  */
116 //CREATE ALL FILE NAMES
117 public static void createFiles()
118 {
119 //Variable Declaration
120 Scanner obj = new Scanner(System.in);
121 String fileName;
122 int linesCount;
123 List<String> content = new ArrayList<String>();
124 //Read file name from user
125 System.out.println("Enter The FileName:");
126 fileName = obj.nextLine();
127 //Read number of lines from user
128 System.out.println("Enter how many lines in the file:");
129 linesCount =Integer.parseInt(obj.nextLine());
130 //Read Lines from user
131 for(int i=1; i<=linesCount; i++)
132 {
133 System.out.println("Enter line" +i+ ":");
134 content.add(obj.nextLine());
135 }
136 //Save and store the content into the file
137 boolean isSaved = FileManager.createFiles(folderpath, fileName, content);
138 if(isSaved)
139 {
140 System.out.println("File and data.saved successfully");
141 }
142 else
143 {
144 System.out.println("some error occured please contact admin@meg.com ");
145 }
146 //close Scanner object
147 obj.close();
148 }
149
```

➤ OUTPUT :

```
*****
----- WELCOME Main Menu LockedMe.com-----
*****
1.Retrive all files
2.create a new file
3.Delete a file
4.Search a file
5.Exit
*****
-----ENTER YOUR CHOICE-----

1
CLASS.TXT
coding.txt
Exam.txt
Hello.txt
image.txt
java.txt
Notes1.txt
oppsjava.txt
Project.txt
test.txt
*****
----- WELCOME Main Menu LockedMe.com-----
*****
1.Retrive all files
2.create a new file
3.Delete a file
4.Search a file
5.Exit
*****
-----ENTER YOUR CHOICE-----
*****
```

```
2
Enter The FileName:
test.txt
Enter how many lines in the file:
1
Enter line1:
GOOD EXAM GOING
File and data.saved successfully
*****
----- WELCOME Main Menu LockedMe.com-----
*****
1.Retrive all files
2.create a new file
3.Delete a file
4.Search a file
5.Exit
*****
-----ENTER YOUR CHOICE-----
*****
```

➤ STEP 2 : DELETE FILE INPUT PROGRAM SOURCE CODE :

```
sourcecoding - FirstProjectLocker/src/com/lockedme/LockedMeProject.java - Eclipse IDE
Edit Source Refactor Navigate Search Project Run Window Help
FileManager.java  LockedMeProject.java x
163 }
164 /**
165  * THIS METHOD IS USED FOR DELETE THE FILE
166  */
167 //DELETE THE FILE
168 public static void deleteFile()
169 {
170     //code for deleting a file
171     String fileName;
172
173     Scanner obj = new Scanner(System.in);
174     System.out.println("ENTER FILE NAME TO BE DELETED: ");
175     fileName = obj.nextLine();
176     boolean isDeleted = FileManager.deleteFile(folderpath ,fileName);
177 {
178 if(isDeleted)
179 {
180 System.out.println("FILE DELETED SUCCESSFULLY:");
181 }
182 else
183 {
184 System.out.println("EITHER FILE NOT THERE OR SOME ACCESS ISSUE: ");
185 }
186 }
```

OUTPUT :

```
*****
-----ENTER YOUR CHOICE-----
*****
3
ENTER FILE NAME TO BE DELETED:
JAVA.txt
FILE DELETED SUCCESSFULLY:
*****
----- WELCOME Main Menu LockedMe.com-----
*****
1.Retrive all files
2.create a new file
3.Delete a file
4.Search a file
5.Exit
*****
-----ENTER YOUR CHOICE-----
```

➤ STEP 3: SEARCH FILE INPUT PROGRAM SOURCE CODE :

```
FileManager.java  *LockedMeProject.java ×
193 /**
194  * THIS IS METHOD IS USED FOR SEARCH THE FILE
195  */
196 //SEARCH THE FILE
197 public static void searchFile()
198 {
199     //code for searching a file
200
201     String fileName;
202
203     Scanner obj = new Scanner(System.in);
204     System.out.println("ENTER THE FILE NAME TO BE SEARCHED:");
205
206     fileName= obj.nextLine();
207
208
209     boolean isFound =FileManager.searchFile(folderpath,fileName);
210
211     if(isFound)
212     {
213         System.out.println("FILE IS PRESENT IN THE FOLDER:");
214     }
215     else
216     {
217
218         System.out.println("FILE IS NOT PRESENT IN THE FOLDER ");
219     }
220 }
221
---
```

➤ OUTPUT :

```
*****
-----ENTER YOUR CHOICE-----
*****
4
ENTER THE FILE NAME TO BE SEARCHED:
Exam.txt
FILE IS NOT PRESENT IN THE FOLDER
*****
----- WELCOME Main Menu LockedMe.com-----
*****
1.Retrive all files
2.create a new file
3.Delete a file
4.Search a file
5.Exit
*****
-----ENTER YOUR CHOICE-----
*****
5
```

8.0 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, appropriate option needs to be selected.
2. The application doesn't restrict user to specify 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.
3. The application also allows user to delete folders which are not empty.
4. The user is able to seamlessly switch between options or return to previous menu even after any required operation like adding, searching, deleting or retrieving of files is performed.
5. The application is designed with modularity in mind. Even if one wants to update the path, they can change it through the source code.

7.0 Conclusion :

That brings us to the Java projects that we encounter in our daily lives. Be it tracking and saving properly data into a files or folder in daily our task so mainly used in that project file handling operations in java language along with the data structure. Java is a robust and secure language and that is what makes it developers' favourite to carry out such projects. Gaining a fluency to code in Java opens doors to many opportunities. So gear up and try these projects to build up your profile in Java. You could also share your attempt vis GitHub in the comments below. Did you try any other fun projects in Java? Let us know and share it with the community.

8.0 APPENDIX

1. FILE MANAGER CLASS CODEING

```
package com.lockedme;
import java.io.File;
import java.io.FileWriter;
import java.util.ArrayList;
import java.util.List;
public class FileManager
{
    /**
    * This method will be return file names from th folder
    * @param folderpath
    * @return List<String>
    */
    public static List<String> getAllFiles(String folderpath)
    {
        //Creating file object
        File f1= new File(folderpath)
        //Getting all the files into a filearray
        File[] listOfFiles = f1.listFiles();
        //Declare a list to store the file names
        List<String> file
        Names = new ArrayList<String>();
        for(File f:listOfFiles)
        fileNames.add(f.getName());
        //return the list
        return fileNames;
    }
    /**
    * This method will create or append content into the file specified
    * @param folderpath
    * @param fileNames
    * @param content
    * @return boolean
    */
    public static boolean createFiles(String folderpath,String
    fileNames,List<String> content)
    {
```

```

try
{
File f1 = new File( folderpath, fileNames);

FileWriter fw= new FileWriter(f1);
for(String s:content)
{
fw.write(s+"\n");
}
fw.close();
return true ;
}
catch(Exception Ex)
{
return false;
}
}

/**
 * This method will delete the file name if it is exist
 * @param folderpath
 * @param fileName
 * @return
 */
public static boolean deleteFile(String folderpath, String fileName)
{
File file= new File("D:\\MYPhase1OppsProject\\LockedMeFiles" +"\\\\"+
fileName);
try
{
if(file.delete())
return true ;
else
return false ;
}
catch(Exception Ex)
{
return false;
}
}

```



```

/**
 * This method will search the file in folder
 * @param folderpath
 * @param fileName
 * @return
 */
public static boolean searchFile(String folderpath, String fileName)
{
    File file= new File("D:\\MYPhase1OppsProject\\LockedMeFiles" +"\\\\"+
    fileName);
    if(file.exists())
    return true ;
    else
    return false ;
}
}

```

LOCKEDME.COM

2. (MAIN CLASS) CODING

```

package com.lockedme;
import java.io.File;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
public class LockedMeProject
{
    static final String folderpath ="D:\\MYPhase1OppsProject\\LockedMeFiles";
    public static void main(String[] args)
    {
        //registration welcome screen
        registrationLoginMainMenu();
    }
}

```

```

        int proceed=1;
        do{
//Variable declaration
Scanner obj = new Scanner(System.in);
        int ch ;
        //Menu
displayMainMenu();
ch=Integer.parseInt(obj.nextLine());
switch(ch
{
        case 1 : getAllFiles();
                break;
        case 2 : createFiles();
                break;
        case 3 : deleteFile();
                break;
        case 4 : searchFile();
                break;
        case 5 : System.exit(0);
                break;
        default :System.out.println("invalid operation");
                break;
} while(proceed>0);
}
public static void registrationLoginMainMenu()
{
String Username;
String Password;
Password = "123";
Username = "BHAGYASHREE";

System.out.println("
=====\\n");
System.out.println("***** WELCOME TO LockedMe.com
*****\\n");
System.out.println("----- Your personal Digital LOCKER -----\\n");
System.out.println("
=====\\n");
System.out.println( "1 .Registration \\n" );
System.out.println( "2 .Login \\n" );
System.out.println("
=====\\n");

```

```

System.out.println("***** WELCOME TO REGISTRATION PAGE
*****\n");
System.out.println("
=====
");
Scanner input1 = new Scanner(System.in);
System.out.println("Enter Username : ");
String username = input1.next();
input2 = new Scanner(System.in);
System.out.println("Enter Password : ");
String password = input2.next();
if (username.equals(Username) && password.equals>Password))
{
    System.out.println("Access Granted! Welcome!");
}
else if (username.equals(Username))
{
    System.out.println("Invalid Password!");
}
else if (password.equals>Password))
{
    System.out.println("Invalid Username!");
}
}
public static void displayMainMenu()
{
    System.out.println("*****
");
    System.out.println("----- WELCOME Main Menu LockedMe.com-----
\n");
    System.out.println("*****
");
    System.out.println("1.Retrieve all files");
    System.out.println("2.create a new file");
    System.out.println("3.Delete a file");
    System.out.println("4.Search a file");
    System.out.println("5.Exit");
    System.out.println("*****
");
    System.out.println(" -----ENTER YOUR CHOICE-----
\n");
    System.out.println("*****
");

```

```

}
// RETRIVE ALL FILE NAME
public static void getAllFiles()
{
//GET THE FILE NAME
List<String> fileNames = FileManager.getAllFiles(folderpath);
for(String f:fileNames)
System.out.println(f);
}
/**
 * THIS METHOD IS USED FOR CREATE THE FILES
 */
//CREATE ALL FILE NAMES
public static void createFiles()
{
//Variable Declartion
Scanner obj = new Scanner(System.in);
String fileName;
int linesCount;
List<String> content = new ArrayList<String>();
//Read file name from user
System.out.println("Enter The FileName:");
fileName = obj.nextLine();
//Read number of lines from user
System.out.println("Enter how many lines in the file:");
linesCount =Integer.parseInt(obj.nextLine());
//Read Lines from user
for(int i=1; i<=linesCount; i++)
{
System.out.println("Enter line" +i+ ":");
content.add(obj.nextLine());
}
//Save and store the content into the file
boolean isSaved = FileManager.createFiles(folderpath, fileName, content);
if(isSaved)
{
System.out.println("File and data.saved successfully");
}
else
{
System.out.println("some error occured please contact admin@meg.com ");
}
//close Scanner object

```

```

//obj.close();
}
/**
 * THIS METHOD IS USED FOR DELETE THE FILE
 */
//DELETE THE FILE
public static void deleteFile()
{
//code for deleting a file
String fileName;
Scanner obj = new Scanner(System.in);
System.out.println("ENTER FILE NAME TO BE DELETED: ");
fileName = obj.nextLine();
boolean isDeleted =FileManager.deleteFile(folderpath ,fileName);
{
if(isDeleted)
{
System.out.println("FILE DELETED SUCCESSFULLY:");
}
else
{
System.out.println("EITHER FILE NOT THERE OR SOME ACCESS ISSUE:");
}
}
//obj.close();
}

```

```

/**
 * THIS IS METHOD IS USED FOR SEARCH THE FILE
 */
//SEARCH THE FILE
public static void searchFile()
{
//code for searching a file
String fileName;
Scanner obj = new Scanner(System.in);
System.out.println("ENTER THE FILE NAME TO BE SEARCHED:");
fileName = obj.nextLine();
boolean isFound =FileManager.searchFile(folderpath,fileNam);
if(isFound){
System.out.println("FILE IS PRESENT IN THE FOLDER:");}
else{

```

```
System.out.println("FILE IS NOT PRESENT IN THE FOLDER ");
}
} //obj.close();
}
```

OUTPUT :

```
=====

***** WELCOME TO LockedMe.com *****

----- Your personal Digital LOCKER -----

=====

1 .Registration

2 .Login

=====

***** WELCOME TO REGISTRATION PAGE *****

=====

Enter Username :

bhagyashree
Enter Password :

123
Invalid Username!
*****
----- WELCOME Main Menu LockedMe.com-----

*****

1.Retrive all files
2.create a new file
3.Delete a file
4.Search a file
5.Exit
*****
-----ENTER YOUR CHOICE-----

*****

1
```

CLASS.TXT
coding.txt
Exam.txt
Hello.txt
image.txt
java.txt
Notes1.txt
oppsjava.txt
Project.txt
test.txt

----- WELCOME Main Menu LockedMe.com-----

- 1.Retrive all files
- 2.create a new file
- 3.Delete a file
- 4.Search a file
- 5.Exit

-----ENTER YOUR CHOICE-----

2

Enter The FileName:

test.txt

Enter how many lines in the file:

1

Enter line1:

GOOD EXAM GOING

File and data.saved successfully

----- WELCOME Main Menu LockedMe.com-----

- 1.Retrive all files
- 2.create a new file
- 3.Delete a file
- 4.Search a file
- 5.Exit

-----ENTER YOUR CHOICE-----

3

ENTER FILE NAME TO BE DELETED:

JAVA.txt

FILE DELETED SUCCESSFULLY:

----- WELCOME Main Menu LockedMe.com-----

- 1.Retrive all files
- 2.create a new file
- 3.Delete a file
- 4.Search a file
- 5.Exit

-----ENTER YOUR CHOICE-----

4

ENTER THE FILE NAME TO BE SEARCHED:

Exam.txt

FILE IS NOT PRESENT IN THE FOLDER

----- WELCOME Main Menu LockedMe.com-----

- 1.Retrive all files
- 2.create a new file
- 3.Delete a file
- 4.Search a file
- 5.Exit

-----ENTER YOUR CHOICE-----

5

=====

=====