

Lockedme.com

## (Sprint Work and Project Specification)

### Version History:

Author	Liladhar Patil
Purpose	Sprint Work and Project Specification.
Date	12th Aug 2021
Version	1.0

## Contents:

1. Modules in the Project:.....	3
2. Java Technologies Used: .....	3
3. Sprint wise Work:.....	3
4. Project GitHub Link: .....	3
5. Project Code:.....	4

## 1. Modules in the Project:

- ❖ **Display all files.**
- ❖ **Add a file.**
- ❖ **Delete a file.**
- ❖ **Search a file.**

## 2. Java Technologies Used:

- Exception Handling:
- Working with Files
- Naming Standards
- Modularity
- Object Oriented Programming
- Collections
- Data Structures
- Control Structures

## 3. Sprint wise Work:

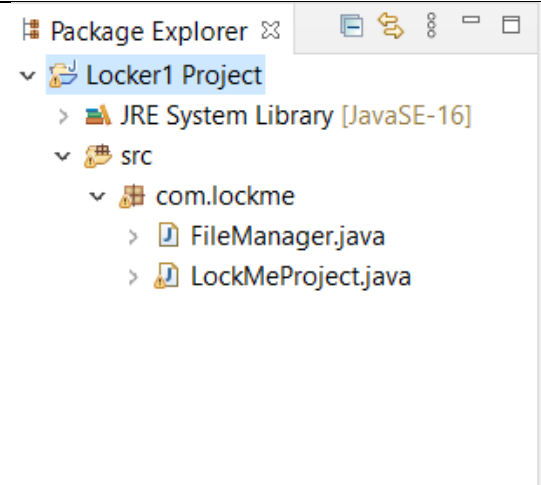
Sprint Number	Module
1	<ul style="list-style-type: none"><li>• Display all Files: Displays all the files in the folder path.</li><li>• Add new File: Adds a new file in the folder path.</li></ul>
2	<ul style="list-style-type: none"><li>• Delete a File: Deletes a file from the folder path.</li><li>• Search File: Searches a file in the folder path.</li><li>• Testing: Running the program code for modules.</li><li>• Deployment (Creating a jar file): Running the code through .jar file.</li></ul>

## 4. Project GitHub Link:

Repository Name:
GitHub Link:

## 5. Project Code:

### Folder Structure:



### FileManager.java:

```
package com.lockme;
import java.io.File;
import java.io.FileWriter;
import java.util.ArrayList;
import java.util.List;

public class FileManager
{
    public static List<String> getALLFiles(String folderpath)
    {
        /**
         * This method will return all file name from the folder
         * @param folderpath
         * @return List<String>
         */
        //Creating File Object
        File fl = new File(folderpath);

        //Getting all files into FileArray
        File[] listOfFiles= fl.listFiles();

        //Declare a list to store file names
        List<String> fileNames = 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 folder specified
     * @param folderpath
     * @param fileName
     * @param content
     * @return boolean
     */
}
```

```

public static boolean createFiles(String folderpath,String fileName, List<String> content)
{
    try
    {
        File fl = new File(folderpath, fileName);
        FileWriter fw = new FileWriter(fl);

        for(String s:content)
        {
            fw.write(s+"\n");
        }
        fw.close();
        return true;
    }
    catch(Exception Ex)
    {
        return false;
    }
}
/**
 * This method will delete the file from the folder.
 * @param folderpath
 * @param fileName
 * @return boolean
 */
public static boolean deleteFile(String folderpath, String fileName)
{
    //adding folder path with file name and creating object
    File file = new File(folderpath+"\\ "+fileName);
    try
    {
        if(file.delete())
            return true;
        else
            return false;
    }
    catch(Exception Ex)
    {
        return false;
    }
}
/**
 * This method will search the file from the folder.
 * @param folderpath
 * @param fileName
 * @return boolean
 */
public static boolean searchFile(String folderpath, String fileName)
{
    //adding folder path with file name and creating object
    File file = new File(folderpath+"\\ "+fileName);
    if(file.exists())
        return true;
    else
        return false;
}
}

```

LockmeProject.java:

```
package com.lockme;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class LockMeProject
{
    static final String folderpath="D:\\New
Volume\\MyPhase1Project\\LockMeFiles";
    public static void main(String[] args)
    {
        int proceed=1;
        do
        {
            //Variable Declaration
            int ch;

            //Menu
            ch=displayMenu();
            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 Option");
                    break;
            }
        } while(proceed>0);
    }

    public static int displayMenu()
    {
        Scanner obj = new Scanner(System.in);
        int ch;

        System.out.println("*****");
        System.out.println("\t\tLockedMe.com");

        System.out.println("*****");
        System.out.println("1. Display all files");
        System.out.println("2. Add 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");
        ch=Integer.parseInt(obj.nextLine());
        //obj.close();
        return ch;
    }
}
```

```

public static void getAllFiles()
{
    // Getting File Name
    List<String> fileNames = FileManager.getALLFiles(folderpath);

    if(fileNames.size()==0)
        System.out.println("No files in the directory");
    else
    {
        System.out.println(" File List is below:\n");

        for (String f:fileNames)
            System.out.println(f);
    }
}
/**
 * Adding files from the folderpath
 */
public static void createFiles()
{
    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 file name:");
    fileName=obj.nextLine();

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

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

    //save 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 occurred. Please contact
admin@abc.com");

    //close scanner object
    obj.close();
}
/**
 * Deleting files in the folderpath
 */
public static void deleteFile()
{
    String fileName;

```

```

Scanner obj = new Scanner(System.in);
System.out.println("Enter the file name to be deleted:");
fileName=obj.nextLine();

//Deleting the file
boolean isDeleted = FileManager.deleteFile(folderpath,
fileName);

if(isDeleted)
    System.out.println("File deleted successfully");
else
    System.out.println("Either file not present or access
issue");
}
/**
 * Searching files in the folderpath
 */
public static void searchFile()
{
    String fileName;
    Scanner obj = new Scanner(System.in);
    System.out.println("Enter the file name to be
searched:");

    fileName=obj.nextLine();

    //Searching the file
    boolean isFound = FileManager.searchFile(folderpath,
fileName);

    if(isFound)
        System.out.println("File is present");
    else
        System.out.println("File is not present in the
folder");
}
}

```