LockedMe Java program

Development by : Amal alyahya

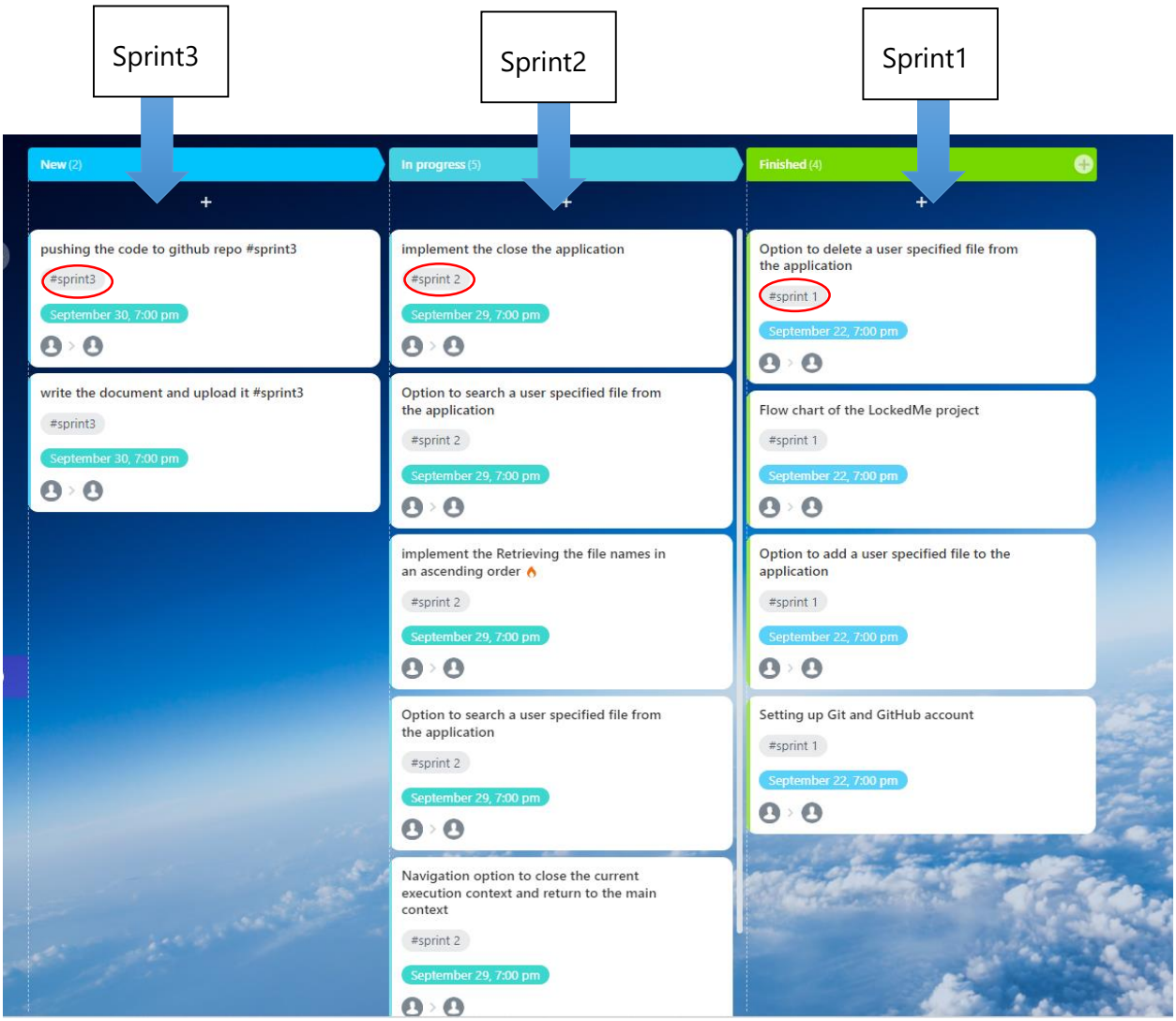
Table of Contents

Core concepts used in the project

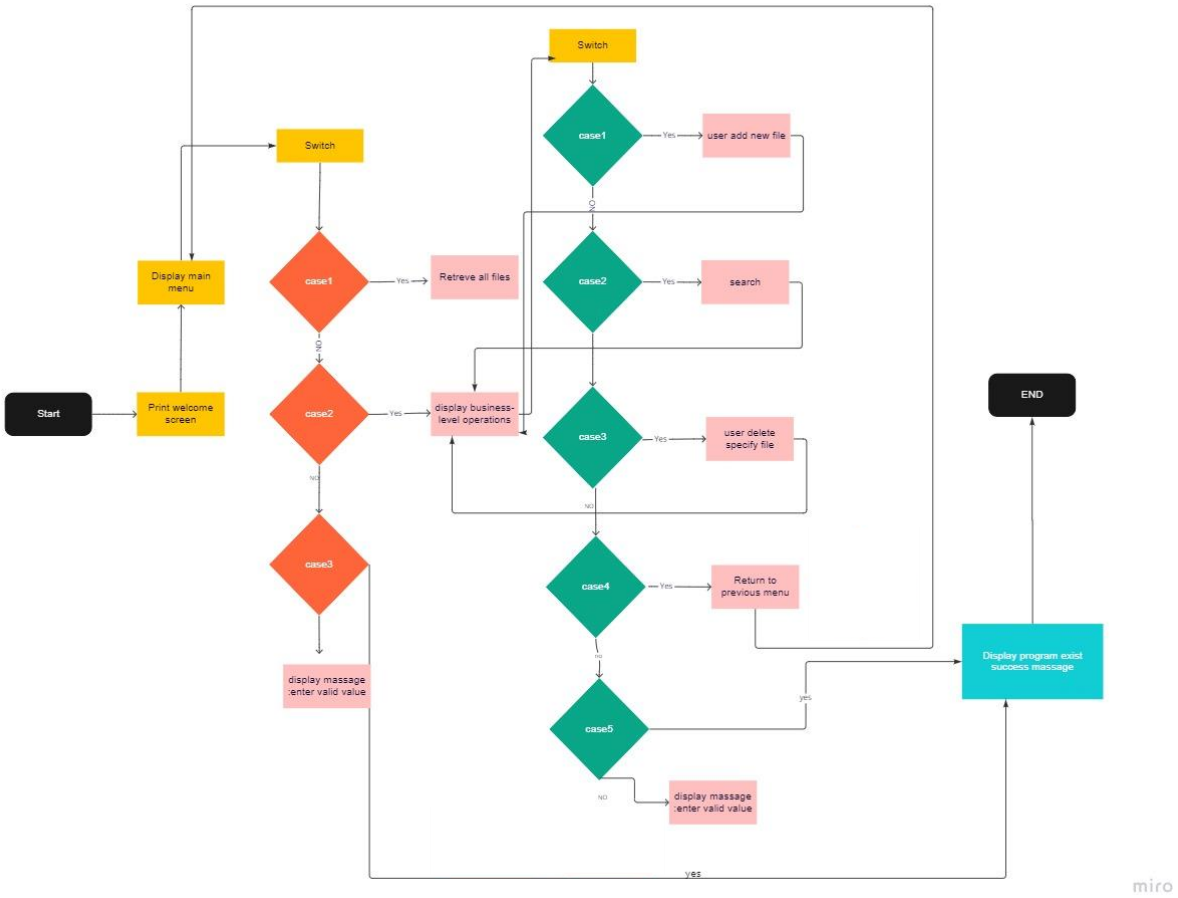
- OOPs Concepts
- Collections
- Exception Handling
- Java File Handling,
- Sorting
- Flow Control
- Java Conditions and loop
- Java Arrays
- Java Methods
- Java Classes.

Sprints planning and Task

The project is planned to be completed in 3 sprints as below:



Flow chart


Start
Print welcome screen
Display main menu
Switch
case1
Yes
Retreve all files
NO
case2
Yes
display business-level operations
NO
case3
display massage :enter valid value
Switch
case1
Yes
user add new file
NO
case2
Yes
search
case3
Yes
user delete specify file
NO
case4
Yes
Return to previous menu
no
case5
yes
NO
display massage :enter valid value
yes
Display program exist success massage
END
miro

Implement the project code

1- **Creating a new project :**For these project I use the STS (Spring Tool Suite) to create the Java program:

Step 1: Open the Spring Tool Suite.

Step 2: Click on the File menu -> New -> JAVA Project

It shows the New Java Project wizard. On the first page −

1. Enter the Project Name: `lockedMeProject`
2. Select the Java Runtime Environment (JRE) or leave it at the default
3. Select the Project Layout which determines whether there would be a separate folder for the source codes and class files. The recommended option is to create separate folders for sources and class files.

Step 3: Click finish

2- Create a new class `main.java` :

- contain the main method:

```
package lockedMeProject;

public class main {
 public static void main(String[] args) {

        handlingOption bo = new handlingOption();
        System.out.println("-----------------------------------\n \n"+
              " Wellcome LockedMe.com \n " +
              "This application developed by Amal Alyahya \n \n" +
              "-----------------------------------\n");
              bo.dispalyMainMenu();

       }

 }
```

3- Create new class: handlingOption.java

```
package lockedMeProject;

import java.io.File;
import java.io.IOException;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.InputMismatchException;
import java.util.List;
import java.util.Scanner;


public class handlingOption {

        Scanner sc= new Scanner(System.in);
        public final static String ROOT_DIRECTORY_PATH="./files/";


      public static void dispalyMainMenu()
      {

   try {
            String msgPrint=
                        "\n \n please enter your chois number :\n \n" +
                  "1-Retrieving the file\n"+
                         "2-Display File operations menu \n"+
                  "3-Exit the application";

                   System.out.println(msgPrint);
                   Scanner sc= new Scanner(System.in);



               int option=sc.nextInt();

                  switch (option) {

                  case 1:
                   retriveAllFiles();
                   break;

                  case 2:
                   dispalyMenu();
                   break;

                  case 3:
                  System.out.println("the porgram is EXIT");
```

```
System.exit(0);
                        break;

                    default :
                     System.out.println("PLEAS ENTER VAID OPTION NUMBER");



                      }} catch (InputMismatchException e)
   {

                   System.out.println("PLEAS ENTER VAID OPTION  \n");
                   dispalyMainMenu();


}

                      }

       // Retrieve all files
       public static void retriveAllFiles() {

       File folder = new File(ROOT_DIRECTORY_PATH);
       File[] listOfFiles = folder.listFiles();
       if (listOfFiles != null && listOfFiles.length > 0) {
    System.out.println("The file in directory : \n "+"---------------------------" );
       for (int i = 0; i < listOfFiles.length; i++) {
         if (listOfFiles[i].isFile()) {
           System.out.println("File :" + listOfFiles[i].getName());
         } else if (listOfFiles[i].isDirectory()) {
           System.out.println("Directory " + listOfFiles[i].getName());
         }
       }
       }else{
             System.out.println("---- Empty Directory ---");
       }
       dispalyMainMenu();
       }



       public static void dispalyMenu()

       {
             try{
                   String s= "1-Add file to the Directory \n"+
                    "2-Search file from the in Directory\n" +
             "3-Delete file from the Directory\n"+
                    "4-Return to main menu \n"+
                    "5-Exit file from the application \n";


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

```
Scanner sc= new Scanner(System.in);
       String fileName;
    int option=sc.nextInt();

       switch (option) {

       case 1:
            System.out.println("please enter the file name: \n");
            fileName=sc.next();
            creatFile(fileName);
             break;

       case 2:
            System.out.println("please enter the file name are you want to search: \n");
            fileName=sc.next();
            findFile( fileName);
                dispalyMenu();
                break;

       case 3:
            System.out.println("please enter the file name to delete: \n");
            fileName=sc.next();
            deleteFile(fileName);

             break;

       case 4:
            dispalyMainMenu();
             break;

       case 5:
            System.out.println("the porgram is EXIT");
            System.exit(0);
             break;
     default :
           System.out.println("PLEAS ENTER VAID OPTION NUMBER \n");
           System.exit(0);


       }
     }   catch (InputMismatchException e)
              {

           System.out.println("PLEAS ENTER VAID OPTION NUMBER \n");
           dispalyMenu();


       }


     }
```

```
        // delete function
         public static void deleteFile(String name)
           {
                boolean isExist=false;
                  try {
                     File folder = new File(ROOT_DIRECTORY_PATH);
                     File[] Files = folder.listFiles();
                           for(File fileEntry : Files) {
                              //System.out.println(fileEntry.getName());
                              if(fileEntry.isFile()) {
                              if (fileEntry.getName().equals(name+".txt")) {
                                    isExist=true;
                                    fileEntry.delete();

                              }

                           }}
                              if(isExist) {
                                    System.out.println("File is delete \n");
                                    dispalyMenu();
                              }
                              else{
                                    System.out.println("File not found or check the case
sensitivity\n");
                                    dispalyMenu();

                              }

                        } catch (Exception e) {
                          System.out.println("An error occurred. \n");
                          e.printStackTrace();
                          dispalyMenu();
                        }
                      }
```

```
// search function
         public static void findFile(String name)
           {
                boolean isExist=false;
                  try {
                     File folder = new File(ROOT_DIRECTORY_PATH);
                     File[] Files = folder.listFiles();
                           for(File fileEntry : Files) {
                              if(fileEntry.isFile()) {
                              if (fileEntry.getName().equals(name+".txt")) {
                                    isExist=true;
                              }

                           }}
                              if(isExist) {
                                    System.out.println("File is existed \n");
                              }
                              else{
                                    System.out.println("File is not existed in the
directory\n");

                              }

                        } catch (Exception e) {
                          System.out.println("An error occurred. \n");
                          e.printStackTrace();
                          dispalyMenu();
                        }
                      }

      }
```

Git and pushing code to GitHub Repository

1. **create a new repository in GitHub website**

   in my case, the name of repo is : LockedMeProject ,and Fill up all the required detail .

2. **Initializin git**

   select my project & open the terminal in my project's root directory, by use : **"git init"** command create a new empty repository or directory consisting of files with the hidden directory. '.git' is created at the top level of your project, which places all of the revision information in one place.

3. **Add the file to the new local repository**
   - Use git add . to add all the files to the given folder.
   - Use git status to view all the files which are going to be staged to the first commit.
4. **Commit the files staged in local repository**
   - create a commit message by: git commit -m 'the message', which adds the change to the local repository.
5. **Add the remote repository's URL from GitHub using**
   - git remote add origin https://github.com/Amal-A-Y/LockedMeProject.git
6. **Push the code in the local repository to GitHub using**
   - git push -u origin master
7. **GitHub page to access the source code**
   - https://github.com/Amal-A-Y/LockedMeProject.git

screenshot of the project

1. welcome screen and retrieving all files:

```
main [Java Application] C:\Program Files\Java\jre1.8.0_311\bin\javaw.exe (Sep 26, 2022, 11:
-----------------------------------

 Wellcome LockedMe.com
 This application developed by Amal Alyahya

---------------------------------------


 please enter your chois number :

1-Retrieving the file
2-Display File operations menu
3-Exit the application
1
The file in directory :
 --------------------------
File :aaa.txt
File :aba.txt
File :abb.txt
File :abc.txt
File :amal1.txt
File :amal2.txt
File :bbbb.txt
File :sds.txt


 please enter your chois number :

1-Retrieving the file
2-Display File operations menu
3-Exit the application
```

2. Chose display file operation menu and create new file name test2:

```
 please enter your chois number :

1-Retrieving the file
2-Display File operations menu
3-Exit the application
2
1-Add file to the Directory
2-Search file from the in Directory
3-Delete file from the Directory
4-Return to main menu
5-Exit file from the application

1
please enter the file name:

test2
File created: test2.txt
```

4. **Create file with existing name**

```
1-Add file to the Directory
2-Search file from the in Directory
3-Delete file from the Directory
4-Return to main menu
5-Exit file from the application

1
please enter the file name:

test1
File already exists.
```

5. **SEARCH:**

1. By file is exist
2. Enter INVALID choice
3. by file not exist
4. test search by case sensitive

```
please enter the file name are you want to search:

test2
File is existed

1-Add file to the Directory
2-Search file from the in Directory
3-Delete file from the Directory
4-Return to main menu
5-Exit file from the application

TEST2
PLEAS ENTER FAID OPTION NUMBER

1-Add file to the Directory
2-Search file from the in Directory
3-Delete file from the Directory
4-Return to main menu
5-Exit file from the application

2
please enter the file name are you want to search:

Test2
File is not existed in the directory

1-Add file to the Directory
2-Search file from the in Directory
3-Delete file from the Directory
4-Return to main menu
5-Exit file from the application

2
please enter the file name are you want to search:

asd
File is not existed in the directory
```

1.

2.

3.

4.

6. **delete and test with case sensitive :**

```
1-Add file to the Directory
2-Search file from the in Directory
3-Delete file from the Directory
4-Return to main menu
5-Exit file from the application

3
please enter the file name to delete:

Test1
File not found or check the case sensitivity

1-Add file to the Directory
2-Search file from the in Directory
3-Delete file from the Directory
4-Return to main menu
5-Exit file from the application

3
please enter the file name to delete:

test1
File is delete
```

7. **Return to previous menu and exit the program**

```
1-Add file to the Directory
2-Search file from the in Directory
3-Delete file from the Directory
4-Return to main menu
5-Exit file from the application

4


 please enter your chois number :

1-Retrieving the file
2-Display File operations menu
3-Exit the application
3
the porgram is EXIT
```

Conclusions

Further enhancements to the application:

- Allowing user to add data to the file.
- Allowing user to read the data on file.
- Retrieving files by different criteria like empty file, type, etc.
- ask the user to verify before delete the file : are you sure for delete “file name”?

References and tools

Tools

https://miro.com

https://www.bitrix24.com

References

https://stackoverflow.com

https://www.w3schools.com

My Github repo

https://github.com/Amal-A-Y/LockedMeProject