ASSESSMENT – 1
Virtual Key for Your Repositories

PROJECT AND DEVELOPER DETAILS

Locker Pvt. Ltd. has hired a Full Stack Developer. They aim to digitalize their products and chose LockedMe.com as their digitalized locker. This project is launched in 2021.

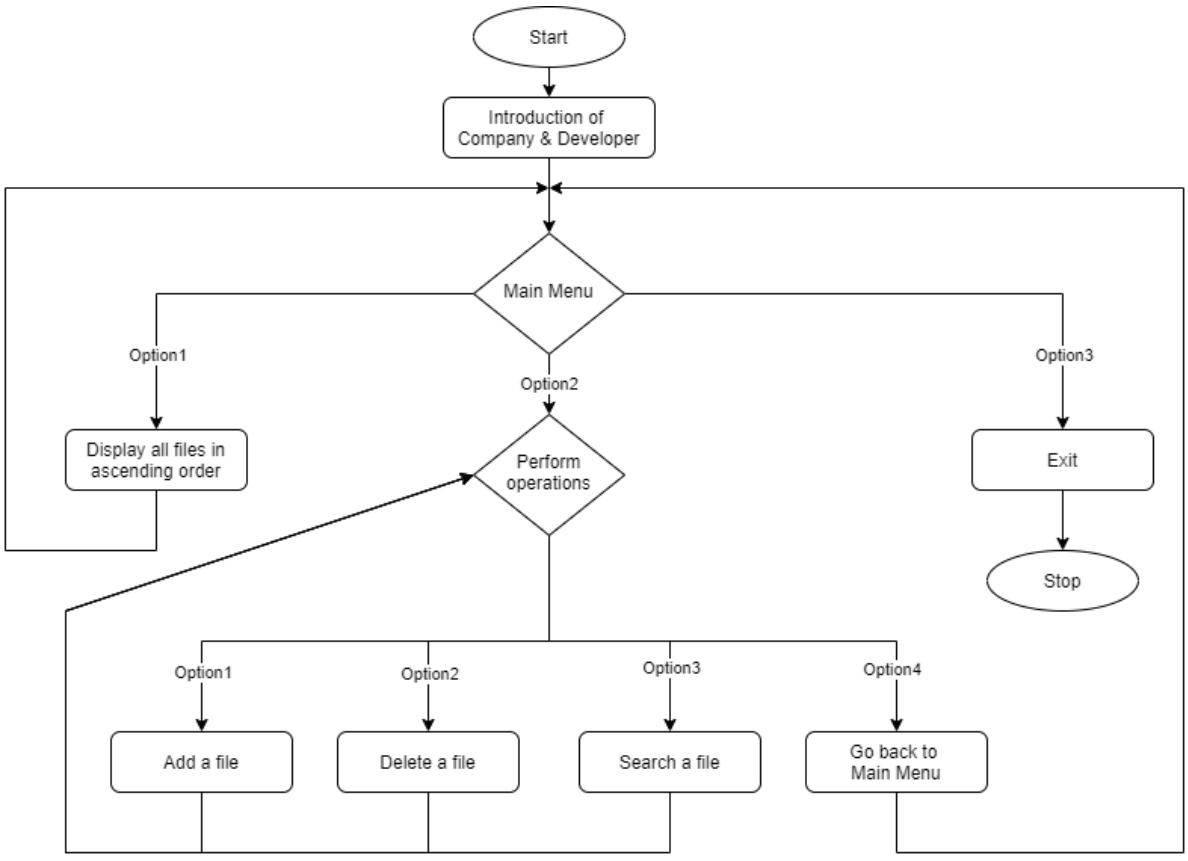
I am Ankita Dagar, a Full Stack Developer hired by the company.

SPRINT PLANNING

The duration of project is 3 weeks (15 working days). All the tasks are planned for a timely delivery of the project.

Deliverable	**Due Date**
Statement of work and Project planning Project Scope and specifications	Week 1
Prototype implementation of the project	Week 2
Final Project for the review	Week 3

ALGORITHMS AND FLOWCHARTS



The code starts with an introduction of the application and its developer.

```
**********************************************************************
Hello User!
Lockers Pvt.Ltd welcomes you to LockedMe.com, your digitalized locker.
This application is developed by Ankita Dagar in 2021.
Hope You enjoy using it!
**********************************************************************
```

Then the MAIN MENU gets displayed and the user can enter the desired choice.

```
------------MAIN MENU------------
1. Display existing files
2. Perform operations on files
3. Exit
Enter your choice:
```

MainMenu Option 1: Displaying existing files

This displays the name of all the files and folders in ascending order present in the root directory. The root directory is already mentioned in the code (and is not asked from the user).

Firstly, we check if the given directory path exists or not. If it exists then, all the files present in the directory are taken into a list. If the size of the list is equal to zero, then message is printed that “there are no files present”. Else the list of all the files is printed in ascending order. The file names are sorted using Collections.sort().

```
------------MAIN MENU------------
1. Display existing files
2. Perform operations on files
3. Exit
Enter your choice:
1
--> This option displays all the existing files in ascending order <--
Display
<---------------------------------------------------------------------->
1.jpg
2.jpg
CEIR Request Details.pdf
Full_Stack_Java_Developer_Masters_Program_Brochure.pdf
GitHub Desktop.lnk
Microsoft Teams.lnk
New Microsoft Word Document.docx
Online Recommendation Form_udit.pdf
Online and Classroom Training for Professional Certification Courses.pdf
Thank You_Udit.pdf
addME.txt
akansha dagar
alien.gif
ankita_digital_Art.jpg
bananaOne.gif
bananaTwo.gif
boatWave.gif
camera
circle.gif
coursera Algorithms-1
desktop.ini
earth.gif
files
glowBall.gif
headTurn.gif
headTurn1.gif
hp offer
my documents
peaParty.gif
pony.gif
random.gif
simplilearn Phase1
smileyRing.gif
sotm
thinkingMan.gif
toonboom harmony
uTorrent Web.lnk
wacom MOM.pdf
waveCover.gif
<---------------------------------------------------------------------->
```

MainMenu Option 2: Performing operations on files

This option takes us to a SUB MENU and user can enter the desired choice.

```
Enter your choice:
2
------------SUB MENU------------
1. Add a file
2. Delete a file
3. Search a file
4. Go Back to MAIN MENU
Enter your choice:
```

SUB MENU has 4 choices among which the user can select the desired operation.

SubMenu Option 1: Add a file

This option asks the user to enter the path of the file which needs to be added. Then, it asks the filename along with its type. The path (destination folder) where the file needs to be added is already set.

Firstly, the code checks if the input file(file to be added) exists or not. If it doesn't exist then, the error message is shown "File to be added doesn't exists". If it exists then, the code checks if there already exists an output file (in the destination folder) with the given filename or not. If it exists then an error message is shown "File with this filename already exists". If it doesn't exists then, a new file is added with the contents of the input file. And a message is displayed saying "File is added".

```
------------SUB MENU------------
1. Add a file
2. Delete a file
3. Search a file
4. Go Back to MAIN MENU
Enter your choice:
1
<--------------------------------------------------------------------->
Enter the path of the file to be added to the existing directory:
C:\\Users\\Ankita\\
Enter the filename along with the type(.txt) to be added to the existing directory (Case-Insensitive):
addME.txt
##Exception: File to be added doesn't exist!!
<--------------------------------------------------------------------->
------------SUB MENU------------
1. Add a file
2. Delete a file
3. Search a file
4. Go Back to MAIN MENU
Enter your choice:
1
<--------------------------------------------------------------------->
Enter the path of the file to be added to the existing directory:
C:\\Users\\Ankita\\Desktop\\
Enter the filename along with the type(.txt) to be added to the existing directory (Case-Insensitive):
addME.txt
File is added!
<--------------------------------------------------------------------->

------------SUB MENU------------
1. Add a file
2. Delete a file
3. Search a file
4. Go Back to MAIN MENU
Enter your choice:
1
<--------------------------------------------------------------------->
Enter the path of the file to be added to the existing directory:
C:\\Users\\Ankita\\Desktop\\
Enter the filename along with the type(.txt) to be added to the existing directory (Case-Insensitive):
addME.txt
File with this filename already exists!!
<--------------------------------------------------------------------->
```

SubMenu Option 2: Delete a file

This option asks the user to enter the filename along with its type which needs to be deleted from the current directory.

The code checks if the filename entered exists then it deletes the file and displays a message “File is Deleted”. If the filename entered doesn’t exists then it displays a message “File to be deleted doesn’t exists”.

```
------------SUB MENU------------
1. Add a file
2. Delete a file
3. Search a file
4. Go Back to MAIN MENU
Enter your choice:
2
<--------------------------------------------------------------------->
Enter the filename along with the type(.txt) to be deleted (Case-sensitive):
addme.txt
File is Deleted!
<--------------------------------------------------------------------->
------------SUB MENU------------
1. Add a file
2. Delete a file
3. Search a file
4. Go Back to MAIN MENU
Enter your choice:
2
<--------------------------------------------------------------------->
Enter the filename along with the type(.txt) to be deleted (Case-sensitive):
addme.txt
File to be deleted doesn't exist!!
<--------------------------------------------------------------------->
```

SubMenu Option 3: Search a file

This option asks the user to enter the filename along with its type which needs to be searched in the current directory.

For searching, we will use .contains() method of list which uses linear search algorithm at the back.

The code checks will fetch the list of all the files in the current directory and use .contains() method to see if the filename entered exists then it displays a message “File found”. If the filename entered doesn’t exists then it displays a message “File not found”.

```
------------SUB MENU------------
1. Add a file
2. Delete a file
3. Search a file
4. Go Back to MAIN MENU
Enter your choice:
3
<--------------------------------------------------------------------->
Enter the filename to be searched along with the type(.txt) (Case-Insensitive):
addme.txt
File Found!
<--------------------------------------------------------------------->
------------SUB MENU------------
1. Add a file
2. Delete a file
3. Search a file
4. Go Back to MAIN MENU
Enter your choice:
3
<--------------------------------------------------------------------->
Enter the filename to be searched along with the type(.txt) (Case-Insensitive):
balh.txt
File not found!!
<--------------------------------------------------------------------->
```

SubMenu Option 4: Go Back to MAIN MENU

This option takes the user back to the MAIN MENU.

```
------------SUB MENU------------
1. Add a file
2. Delete a file
3. Search a file
4. Go Back to MAIN MENU
Enter your choice:
4
------------MAIN MENU------------
1. Display existing files
2. Perform operations on files
3. Exit
Enter your choice:
```

MainMenu Option 3: Exit

This option exits the code.

CORE CONCEPTS USED

This project uses the concepts of

- File Handling (Create a new file, Reading a file, Writing in a file, Deleting a file, checking if a file Exists),
- Exceptions,
- Scanner (for user inputs),
- Arrays,
- List,
- Collections.

GITHUB ACCOUNT

https://github.com/adagar95/simplilearn/tree/main/SLPhase1Assessment

This link to the GitHub repository of the project.

CONCLUSION

The purpose of this project is to digitalise the lockers to store all your important files at your click. This will make the management process of the files easy and efficient. This locker has various features like displaying all the files in ascending order which makes it easy to view already stored files. User can also add, delete and search any file with a quick response time. The algorithms used in the project not only saves time but also keeps the user engaged without losing their interest with an active interactive UI them making it the USP of the project.