LOCKEDME(**Virtual Key for Your Repositories)**

**Source code:**

**1)welcome screen.**

**package org.example.lockedme.screens;**

**import org.example.lockedme.services.DirectoryService;**

**import org.example.lockedme.services.ScreenService;**

**import java.util.ArrayList;**

**import java.util.InputMismatchException;**

**import java.util.Scanner;**

**public class WelcomeScreen implements Screen {**

**private String welcomeText = "Welcome to lockedme!";**

**private String developerText = "Developer: VERRINAIDU CHITIKELA";**

**private ArrayList<String> options = new ArrayList<>();**

**public WelcomeScreen() {**

**options.add("1. Show Files");**

**options.add("2. Show File Options Menu");**

**options.add("3. Quit");**

**}**

**public void introWS() {**

**System.out.println(welcomeText);**

**System.out.println(developerText);**

**System.out.println("\n");**

**Show();**

**}**

**@Override**

**public void Show() {**

**System.out.println("Main Menu");**

**for (String s : options) {**

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

**}**

**}**

**public void GetUserInput() {**

**int selectedOption = 0;**

**while ((selectedOption = this.getOption()) != 3) {**

**this.NavigateOption(selectedOption);**

**}**

**}**

**@Override**

**public void NavigateOption(int option) {**

**switch(option) {**

**case 1:**

**this.ShowFiles();**

**this.Show();**

**break;**

**case 2:**

**ScreenService.setCurrentScreen(ScreenService.FileOptionsScreen);**

**ScreenService.getCurrentScreen().Show();**

**ScreenService.getCurrentScreen().GetUserInput();**

**this.Show();**

**break;**

**default:**

**System.out.println("Invalid Option");**

**break;**

**}**

**}**

**public void ShowFiles() {**

**System.out.println("List of Files: ");**

**DirectoryService.PrintFiles();**

**}**

**private int getOption() {**

**Scanner in = new Scanner(System.in);**

**int returnOption = 0;**

**try {**

**returnOption = in.nextInt();**

**}**

**catch (InputMismatchException ex) {**

**}**

**return returnOption;**

**}**

**}**

**2)FileoptionScreen**

**package org.example.lockedme.screens;**

**import java.io.File;**

**import java.io.IOException;**

**import java.nio.file.FileSystems;**

**import java.nio.file.Path;**

**import java.util.ArrayList;**

**import java.util.InputMismatchException;**

**import java.util.Scanner;**

**import org.example.lockedme.entities.Directory;**

**import org.example.lockedme.services.ScreenService;**

**public class FileOptionsScreen implements Screen {**

**private Directory dir = new Directory();**

**private ArrayList<String> options = new ArrayList<>();**

**public FileOptionsScreen() {**

**options.add("1. Add a File");**

**options.add("2. Delete A File");**

**options.add("3. Search A File");**

**options.add("4. Return to Menu");**

**}**

**@Override**

**public void Show() {**

**System.out.println("File Options Menu");**

**for (String s : options) {**

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

**}**

**}**

**public void GetUserInput() {**

**int selectedOption;**

**while ((selectedOption = this.getOption()) != 4) {**

**this.NavigateOption(selectedOption);**

**}**

**}**

**@Override**

**public void NavigateOption(int option) {**

**switch(option) {**

**case 1: // Add File**

**this.AddFile();**

**this.Show();**

**break;**

**case 2: // Delete File**

**this.DeleteFile();**

**this.Show();**

**break;**

**case 3: // Search File**

**this.SearchFile();**

**this.Show();**

**break;**

**/\***

**case 4: // Return to Menu**

**ScreenService.setCurrentScreen(ScreenService.WelcomeScreen);**

**ScreenService.getCurrentScreen().Show();**

**ScreenService.getCurrentScreen().GetUserInput();**

**break;**

**\*/**

**default:**

**System.out.println("Invalid Option");**

**break;**

**}**

**}**

**//TODO: Add functionality to all**

**// Finished TODO**

**public void AddFile() {**

**System.out.println("Please Enter the Filename:");**

**String fileName = this.getInputString();**

**System.out.println("You are adding a file named: " + fileName);**

**try {**

**Path path = FileSystems.getDefault().getPath(Directory.name + fileName).toAbsolutePath();**

**File file = new File(dir.getName() + fileName);**

**if (file.createNewFile()) {**

**System.out.println("File created: " + file.getName());**

**dir.getFiles().add(file);**

**} else {**

**System.out.println("This File Already Exits, no need to add another");**

**}**

**}catch (IOException e){**

**System.out.println(e);**

**}**

**}**

**public void DeleteFile() {**

**System.out.println("Please Enter the Filename:");**

**String fileName = this.getInputString();**

**System.out.println("You are deleting a file named: " + fileName);**

**//TODO: Delete file**

**// Finished TODO**

**Path path = FileSystems.getDefault().getPath(Directory.name + fileName).toAbsolutePath();**

**File file = path.toFile();**

**if (file.delete()) {**

**System.out.println("Deleted File: " + file.getName());**

**dir.getFiles().remove(file);**

**} else {**

**System.out.println("Failed to delete file:" + fileName + ", file was not found.");**

**}**

**}**

**public void SearchFile() {**

**Boolean found = false;**

**System.out.println("Please Enter the Filename:");**

**String fileName = this.getInputString();**

**System.out.println("You are searching for a file named: " + fileName);**

**//TODO Fix it so ArrayList obtains files**

**//Finished TODO**

**ArrayList<File> files = dir.getFiles();**

**for(int i = 0; i < files.size(); i++) {**

**if(files.get(i).getName().equals(fileName)) {**

**System.out.println("Found " + fileName);**

**found = true;**

**}**

**}**

**if (found == false) {**

**System.out.println("File not found");**

**}**

**}**

**private String getInputString() {**

**Scanner in = new Scanner(System.in);**

**return(in.nextLine());**

**}**

**private int getOption() {**

**Scanner in = new Scanner(System.in);**

**int returnOption = 0;**

**try {**

**returnOption = in.nextInt();**

**}**

**catch (InputMismatchException ex) {**

**System.out.println("Invalid input");**

**}**

**return returnOption;**

**}**

**}**

**3)screen**

**package** org.example.lockedme.screens;

**public** **interface** Screen {

**public** **void** Show();

**public** **void** NavigateOption(**int** option);

**public** **void** GetUserInput();

}

**4)Directory .java**

**package org.example.lockedme.entities;**

**import java.util.ArrayList;**

**import java.util.Collections;**

**import java.io.File;**

**import java.nio.file.FileSystems;**

**import java.nio.file.Path;**

**public class Directory {**

**public static final String name = "src/main/directory/";**

**//TODO: Possibly use a HashMap? Too complex for now.**

**private ArrayList<File> files = new ArrayList<File>();**

**Path path = FileSystems.getDefault().getPath(name).toAbsolutePath();**

**File Dfiles = path.toFile();**

**public String getName() {**

**return name;**

**}**

**public void print() {**

**System.out.println("Existing Files: ");**

**files.forEach(f -> System.out.println(f));**

**}**

**public ArrayList<File> fillFiles() {**

**File[] directoryFiles = Dfiles.listFiles();**

**files.clear();**

**for (int i = 0; i < directoryFiles.length; i++) {**

**if (directoryFiles[i].isFile()) {**

**files.add(directoryFiles[i]);**

**}**

**}**

**Collections.sort(files);**

**return files;**

**}**

**public ArrayList<File> getFiles() {**

**fillFiles();**

**return files;**

**}**

**public void mergeSort() {**

**}**

**}**

5)Lockedme

**package** org.example.lockedme;

**import** org.example.lockedme.screens.WelcomeScreen;

**public** **class** lockedme {

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

WelcomeScreen welcome = **new** WelcomeScreen();

welcome.introWS();

welcome.GetUserInput();

}

}

6)ScreenService

package org.example.lockedme.services;

import org.example.lockedme.screens.FileOptionsScreen;

import org.example.lockedme.screens.Screen;

import org.example.lockedme.screens.WelcomeScreen;

import org.example.lockedme.entities.Directory;

public class ScreenService {

//static Directory dir = new Directory();

public static WelcomeScreen WelcomeScreen = new WelcomeScreen();

public static FileOptionsScreen FileOptionsScreen = new FileOptionsScreen();

public static Screen CurrentScreen = WelcomeScreen;

public static Screen getCurrentScreen() {

return CurrentScreen;

}

public static void setCurrentScreen(Screen currentScreen) {

CurrentScreen = currentScreen;

}

}

7)Directory Service

package org.example.lockedme.services;

import java.io.File;

import org.example.lockedme.entities.Directory;

public class DirectoryService {

private static Directory fileDirectory = new Directory();

public static void PrintFiles() {

fileDirectory.fillFiles();

for (File file : DirectoryService.getFileDirectory().getFiles())

{

System.out.println(file.getName());

}

}

public static Directory getFileDirectory() {

return fileDirectory;

}

public static void setFileDirectory(Directory fileDirectory) {

DirectoryService.fileDirectory = fileDirectory;

}

}