CSC 316: Project UserActivityLogManager - System Test Plan

Document Author: Michael Warstler
Date: 10/30/2024

Introduction

For the system test cases below, I will be using the following test file named ***records.txt***. The main goals of the tests are to check for proper sorting via frequency, date, and hour of activities, correct input of a file of log entries, and error checking for invalid user inputs.

To run the test:

1. Right click on the UserActivityLogManagerUI.java in the Package Explorer.
2. Select Run As -> Java Application

records.txt: (located in project's input folder)

USERNAME, TIMESTAMP, ACTION, RESOURCE

labyrum, 04/06/2013 07:30:42PM, call, office OV04392

labyrum, 02/27/2020 05:30:50PM, call, office OV04392

mwwarstl, 12/13/2019 09:40:48PM, register, HL3 Code 691

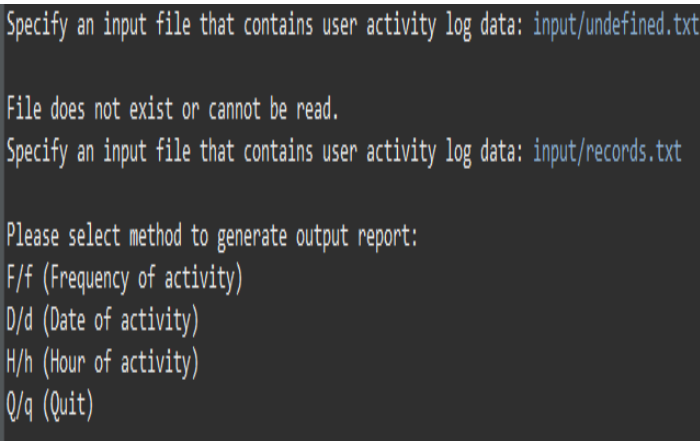mwwarstl, 02/22/2016 11:09:46PM, register, HL3 Code 691

mwwarstl, 02/27/2020 07:18:42AM, notify, HL6 Code 783

mwwarstl, 02/27/2020 05:30:50PM, register, HL3 Code 691

labyrum, 02/19/2017 06:16:58PM, register, HL3 Code 691

mwwarstl, 01/04/2016 12:44:52PM, register, HL3 Code 691

labyrum, 11/08/2016 10:43:29AM, register, HL3 Code 691

mwwarstl, 07/06/2015 04:17:06PM, register, HL3 Code 691

labyrum, 12/18/2017 03:02:54AM, register, HL3 Code 691

labyrum, 09/11/2016 09:14:44PM, register, HL3 Code 691

labyrum, 04/15/2017 09:14:59PM, notify, HL6 Code 783

labyrum, 10/06/2016 04:58:44AM, register, HL3 Code 691

labyrum, 01/23/2017 12:05:22AM, register, HL3 Code 691

labyrum, 09/12/2023 01:00:15AM, unmerge, notification NX1115

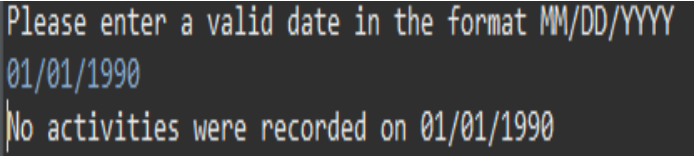labyrum, 01/24/2024 02:16:27AM, view, HL7 Code 422

System Tests

Actual results sometimes displayed as a screenshot from the console.
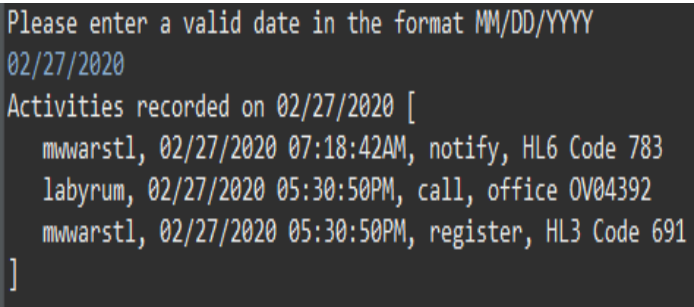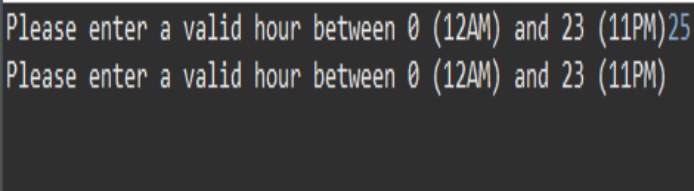
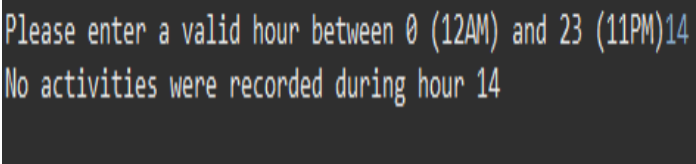| Test ID	Description	Expected Results	Actual Results
Test #1 **testID: testLoadFile** **Strategy:** **(Exception/unexpected input - trying to load file that does not exist and confirming correct program flow.)**	**Preconditions:** - UserActivityLogManagerUI has been loaded successfully. - File records.txt exists. **Steps:** 1. User types in input/undefined.txt 2. User types in input/records.txt 3. *Check results.*	Step 1: Program begins and below is first prompt to user on the command window: "Specify an input file that contains user activity log data: " Program reprompts user to specify a new file with following text: "File does not exist or cannot be read. Specify an input file that contains user activity log data: " Step 2-3: File has been successfully loaded. New prompt to user states below: "Please select method to generate output report: F/f (Frequency of activity) D/d (date of activity) H/h (hour of activity)	Specify an input file that contains user activity log data: input/undefined.txt File does not exist or cannot be read. Specify an input file that contains user activity log data: input/records.txt Please select method to generate output report: F/f (Frequency of activity) D/d (Date of activity) H/h (Hour of activity) Q/q (Quit)

		Q/q (Quit)"	
Test #2 **testID: testInvalidFrequency** **Strategy:** **(Exception/unexpected input.)** **(Boundary value - choosing 0 for number of activities.)**	**Preconditions:** - UserActivityLogManagerUI has been loaded successfully. - File records.txt exists. - User types in input/records.txt as the input file. **Steps:** 1. User selects F/f for frequency of activity. 2. User enters 0. 3. *Check results.*	Step 1: Prompt appears for user stating the following: "Please specify how many activities should appear in the report: " Step 2: Prompt states the following: "Please enter a number > 0"	F Please specify how many activities should appear in the report:0 Please enter a number > 0
Test #3 **testID: testFrequencyReport** **Strategy: (Equivalence class - generating correct output)**	**Preconditions:** - UserActivityLogManagerUI has been loaded successfully. - File records.txt exists. - User types in input/records.txt as the input file. **Steps:**	Program displays the following report to console: Top User Activities Report [　11: register HL3 Code 691 　2: call office OV04392 　2: notify HL6 Code 783]	F Please specify how many activities should appear in the report:3 Top User Activities Report [　11: register HL3 Code 691 　2: call office OV04392 　2: notify HL6 Code 783]

	1. User selects F/f for frequency of activity. 2. User enters 3. 3. *Check results.*		

Test #4
 	
testID: testFrequencyReportOneActivity
 	
Strategy: (Equivalence class - generating correct output)
 	
(Boundary value - choosing only 1 activity.
	Preconditions:
	
	- UserActivityLogManagerUI has been loaded successfully.
	- File records.txt exists.
	- User types in input/records.txt as the input file.
	
	Steps:
	
	1. User selects F/f for frequency of activity.
	2. User enters 1.
	3. *Check results.*
		Program displays the following report to console:
		
		Top User Activities Report [
		    11: register HL3 Code 691
		]
			Please specify how many activities should appear in the report:1
			Top User Activities Report [
			    11: register HL3 Code 691
			]

Test #5
 	
testID:
testFrequencyReportAllActivities
 	
Strategy: (Equivalence class - generating correct output)
	Preconditions:
	
	- UserActivityLogManagerUI has been loaded successfully.
	- File records.txt exists.
	- User types in input/records.txt as the input file.
	
	Steps:
	
	1. User selects F/f for frequency of activity.
	2. User enters 50.
	3. *Check results.*
		Program displays the following report to console:
		
		Top User Activities Report [
		  11: register HL3 Code 691
		  2: call office OV04392
		  2: notify HL6 Code 783
		  1: unmerge notification NX1115
		  1: view HL7 Code 422
		]
			Please specify how many activities should appear in the report:50
			Top User Activities Report [
			  11: register HL3 Code 691
			  2: call office OV04392
			  2: notify HL6 Code 783
			  1: unmerge notification NX1115
			  1: view HL7 Code 422
			]

Test #6 **testID: testInvalidDateReport** **Strategy: (Exception/unexpected input - user formats date incorrectly)**	**Preconditions:** - UserActivityLogManagerUI has been loaded successfully. - File records.txt exists. - User types in input/records.txt as the input file. **Steps:**	Step 1: Prompt appears for user stating the following: "Please enter a valid date in the format MM/DD/YYYY" Step 2: Program reprompts user with following message: "Please enter a valid date in the format MM/DD/YYYY"	d Please enter a valid date in the format MM/DD/YYYY 12345 Please enter a valid date in the format MM/DD/YYYY

	1. User selects D/d for date of activity. 2. User enters 12345 3. *Check results.*		
Test #7 **testID:** **testDateReportNoDateFound** **Strategy: (Equivalence class - generating correct output)**	**Preconditions:** - UserActivityLogManagerUI has been loaded successfully. - File records.txt exists. - User types in input/records.txt as the input file. **Steps:** 1. User selects D/d for date of activity. 2. User enters 01/01/1990 4. *Check results.*	Program displays following message: "No activities were recorded on 01/01/1990"	Please enter a valid date in the format MM/DD/YYYY 01/01/1990 No activities were recorded on 01/01/1990

Test #8
 	
testID: testDateReport
 	
Strategy: (Equivalence class - generating correct output)
	Preconditions:
	
	- UserActivityLogManagerUI has been loaded successfully.
	- File records.txt exists.
	- User types in input/records.txt as the input file.
	
	Steps:
	
	1. User selects D/d for date of activity.
	2. User enters 02/27/2020
	3. *Check results.*
		Program displays following report to console:
		
		Activities recorded on 02/27/2020 [
		   mwwarstl, 02/27/2020 07:18:42AM, notify, HL6 Code 783
		   labrum, 02/27/2020 05:30:50PM, call, office OV04392
		   mwwarstl, 02/27/2020 05:30:50PM, register, HL3 Code 691
		]	

Test #9 **testID: testInvalidHourReport** **Strategy: (Exception/unexpected input - user chooses invalid hour)**	**Preconditions:** - UserActivityLogManagerUI has been loaded successfully. - File records.txt exists. - User types in input/records.txt as the input file. **Steps:**	Step 1: Prompt appears for user stating the following: "Please enter a valid hour between 0 (12AM) and 23 (11PM)"	

	1. User selects H/h for hour of activity. 2. User enters 25. 3. *Check results*.		
Test #10 **testID: testEmptyHourReport** **Strategy: (Equivalence class - generating correct output.)** **(Exception/unexpected input - user picks hour where no activities occurred.)**	**Preconditions:** - UserActivityLogManagerUI has been loaded successfully. - File records.txt exists. - User types in input/records.txt as the input file. **Steps:** 1. User selects H/h for hour of activity. 2. User enters 14. 3. *Check results*.	Program displays following message: "No activities were recorded during hour 14"	Please enter a valid hour between 0 (12AM) and 23 (11PM)14 No activities were recorded during hour 14

| **Test #11**<br><br>**testID: testHourReport**<br><br>**Strategy: (Equivalence class - generating correct output.)** | **Preconditions:**<br><br>- UserActivityLogManagerUI has been loaded successfully.<br>- File records.txt exists.<br>- User types in input/records.txt as the input file.<br><br>**Steps:**<br><br>1. User selects H/h for hour of activity.<br>2. User enters 21<br>3. *Check results.* | Program displays following report to console:<br><br>Activities recorded during hour 21 [<br>  labrum, 09/11/2016 09:14:44PM, register, HL3 Code 691<br>  labyrum, 04/15/2017 09:14:59PM, notify, HL6 Code 783<br>  mwwarstl, 12/13/2019 09:40:48PM, register, HL3 Code 691<br>] | ```
Please enter a valid hour between 0 (12AM) and 23 (11PM)21
Activities recorded during hour 21 [
    labyrum, 09/11/2016 09:14:44PM, register, HL3 Code 691
    labyrum, 04/15/2017 09:14:59PM, notify, HL6 Code 783
    mwwarstl, 12/13/2019 09:40:48PM, register, HL3 Code 691
]
``` |
| **Test #12**<br><br>**testID:<br>testHourReportSameTime**<br><br>**Strategy: (Equivalence class - generating correct output.)** | **Preconditions:**<br><br>- UserActivityLogManagerUI has been loaded successfully.<br>- File records.txt exists.<br><br>- User types in input/records.txt as the input file.<br><br>**Steps:** | Program displays following report to console:<br><br>Activities recorded during hour 17 [<br>  labrum, 02/27/2020 05:30:50PM, call, office OV04392<br>  mwwarstl, 02/27/2020 05:30:50PM, register, HL3 Code 691<br>] | ```
Please enter a valid hour between 0 (12AM) and 23 (11PM)17
Activities recorded during hour 17 [
    labyrum, 02/27/2020 05:30:50PM, call, office OV04392
    mwwarstl, 02/27/2020 05:30:50PM, register, HL3 Code 691
]
``` |

	1. User selects H/h for hour of activity. 2. User enters 17 3. *Check results*.		
Test #13 **testID: testZeroHourReport** **Strategy: (Equivalence class - generating report file.)** **(Boundary value - choosing an hour of activity that is 0)**	**Preconditions:** - UserActivityLogManagerUI has been loaded successfully. - File records.txt exists. - User types in input/records.txt as the input file.. **Steps:** 1. User selects H/h for hour of activity. 2. User enters 0 3. *Check results*.	Program displays following report to console: Activities recorded during hour 0 [labyrum, 01/23/2017 12:05:22AM, register, HL3 Code 691]	

		Program successfully quits.	Program terminates successfully.
Test #14 **testID: testQuit** **Strategy: (Equivalence class - proper program functionality)**	**Preconditions:** - UserActivityLogManagerUI has been loaded successfully. - File records.txt exists. - User types in input/records.txt as the input file. **Steps:** 1. User selects Q/q for quit. 2. *Check results*.		