Daniel Avisse

Cleaning Manager System Test Plan

Test Data:

Input filename	Contents of the file
Input-rooms-1.txt	ROOM_ID,LENGTH,WIDTH Office,20,20 Lunch Room,30,25 Game Room,22,24 Living Room,15,20 Gym,50,50 Closet,5,8 Bedroom,14,20 Hallway,10,10 Attic,30,17 Bathroom,10,10
Input-cleaned-rooms-1.txt	TIMESTAMP,ROOM_ID,PERCENT_CLEANED 04/20/2022 05:30:59,Office,58 12/25/2022 02:22:22,Lunch Room,42 10/31/2022 05:01:21,Living Room,66 10/26/2022 04:17:22,Gym,69 03/14/2022 03:14:15,Closet,85 07/04/2022 07:04:22,Bedroom,99 01/01/2022 03:17:22,Hallway,7 02/14/2022 04:01:22,Lunch Room,1 07/14/2022 04:51:57,Lunch Room,2 06/19/2022 06:01:01,Lunch Room,3 05/08/2022 07:32:10,Closet,64 02/21/2022 00:00:00,Closet,70 10/10/2022 08:15:34,Bedroom,98 11/08/2022 12:00:00,Bedroom,100 10/26/2022 23:59:59,Gym,69 12/31/2022 00:00:00,Hallway,22

Invalid-rooms.txt	**ROOM_ID,LENGTH,WIDTH** **Room,Twenty,Thirty** **Bedroom,Ten,Four** **Office,Ninety,Twelve**
Invalid-cleaned-rooms-1.txt	**TIMESTAMP,ROOM_ID,PERCENT_CLEANED** **13/1/2021 19:20:06, Office,50** **12/22/2021 15:20:22, Game Room,25** **1/21/2022 09:55:02, Attic,99**
Input-boundary-rooms-1.txt	**ROOM_ID,LENGTH,WIDTH** **Gym,60,88**
Input-boundary-cleaned-rooms-1.txt	**TIMESTAMP,ROOM_ID,PERCENT_CLEANED** **01/01/2022 00:00:00,Gym,50** **12/31/2022 23:59:59,Gym,50**
sample-logs.csv	**TIMESTAMP,ROOM_ID,PERCENT_CLEANED** **6/1/2021 13:39,Office,78** **5/31/2021 9:27,Dining Room,89** **5/30/2021 10:14,Living Room ,68** **5/28/2021 17:22,Living Room,70** **5/21/2021 9:16,Dining Room,86** **5/23/2021 18:22,Dining Room,89** **5/23/2021 11:51,Guest Bedroom,77** **5/17/2021 4:37,Guest Bathroom,91** **5/9/2021 18:44,Living Room,89** **5/12/2021 18:59,Living Room ,94** **5/13/2021 22:20,Guest Bedroom,74** **5/8/2021 7:01,Guest Bathroom,91** **5/1/2021 10:03,Foyer,93** **5/3/2021 17:22,Living Room,92** **5/11/2021 19:00,Living Room ,89**
sample-rooms.csv	**ROOM_ID,LENGTH,WIDTH** **Office,12,14** **Dining Room,15,15** **Living Room,25,35** **Guest Bedroom,17,16** **Guest Bathroom,10,8**

	Foyer ,8,8 **Kitchen,18,20**
empty-rooms.txt	**ROOM_ID,LENGTH,WIDTH**
empty-logs.txt	**TIMESTAMP,ROOM_ID,PERCENT_CLEANED**

Setup: To start the program the user should select the CleanMangagerUI.java file and run it.

Test ID	Description	Expected Results	Actual Results
testID(Test1): testLoadValidFiles **Strategy: (Equivalence class - Loading a valid room and log files)**	**Preconditions:** • **CleaningManagerUI has been loaded successfully and prompts the user for input.** • **The files `Input-rooms-1.txt` and `Input-cleaned-rooms-1.txt` exist.** **Steps:** 1. **Type the name input/Input-rooms-1.txt into the command line for rooms file.** 2. **Type the name input/Input-cleaned-rooms-1.txt into the command line for cleaning logs.**	**A message indicating that the files have loaded should appear and the following menu is then displayed asking for input from the user.** ***Cleaning Manager - Please Choose One of the Options Below*** ***F - View the Most Frequently Cleaned Rooms*** ***R - View a Report of Cleanings by Room*** ***V - View Estimated Remaining Vacuum Bag Life*** ***Q - Quit the Program*** ***Option:***	• The CleaningUI successfully loads, and the file names are entered into the program. • A message appears showing the files have been loaded and the menu displaying the different options and the choice for allowing to pick an option shows up. **Results Match Expected (Success)**

testID(Test 2): testLoadInvalidFiles **Strategy: (Unexpected input/Error Handling – Invalid files. Files are invalid because they either don't follow the correct room format or have incorrect information)**	**Preconditions:** ● **The input files `Invalid-rooms.txt and Invalid-cleaned-rooms.txt` exist.** ● **The CleaningManager loads up successfully.** **Steps:** 1. **Type the name input/Invalid-rooms.txt into the command line for rooms file.** 2. **Type the name input/Invalid-cleaned-rooms.txt into command line for cleaning logs.**	**The files are loaded however the program displays the following message.** ***ERROR: The files you have entered either do not exist or are formatted incorrectly.*** ***Please try again.*** **The user is then prompted again to input the names of files again.** ***File Location of Rooms:*** ***File Location of Cleaning Log Entries:***	• The CleaningUI successfully loads, and the file names are entered into the program. • The program tries to load the files but instead an error message appears saying that the files don't exist or formatted incorrectly. • The program then prompts for new files. **Results Match Expected (Success)**
testID(Test 3):testViewMostCleanedRooms **Strategy: (Equivalence class - checking to see if the correct rooms are displayed with how many times they are cleaned)**	**Preconditions:** ● **CleaningManagerUI has been loaded successfully.** ● **The file `Input-rooms-1.txt and Input-cleaned-rooms-1.txt` exist.** ● **testLoadValidFiles (Test 1) has passed.** **Steps:**	**The valid files have been loaded and the menu is displayed.** **The letter "F" is inputted and the user inputs that they want to see 10 rooms. The following should be outputted by the program.** ***Frequency of Cleanings [*** • ***Lunch Room has been cleaned 4 times*** • ***Bedroom has been cleaned 3 times*** • ***Closet has been cleaned 3 times*** • ***Gym has been cleaned 2 times*** • ***Hallway has been cleaned 2 times*** • ***Living Room has been cleaned 1 times***	• The CleaningUI successfully loads, and the file names are entered into the program. • A message appears showing the files have been loaded and the menu displaying the different options and the choice for allowing to pick an option shows up.

	1. **Type the name input/Input-rooms-1.txt into the command line for rooms file.** 2. **Type the name input/Input-cleaned-rooms-1.txt int command line for cleaning logs.** 3. **Type in the letter "F" for frequently cleaned rooms** 4. **Type in the number 10 to see all rooms**	• ***Office has been cleaned 1 times*** • ***Attic has been cleaned 0 times*** • ***Bathroom has been cleaned 0 times*** • ***Game Room has been cleaned 0 times*** ***]*** **The program then redisplays the menu prompting the user for another option.**	• The letter "F" is entered to pick the view most frequently cleaned rooms option. • The program then asks how many rooms should be displayed and the number 10 is entered. • The program outputs the list of rooms and how many times they have been cleaned matching the same order as the expected. • The menu is then redisplayed and another option can be made. **Results Match Expected (Success)**
testID(Test 4): testViewMostClearnedRoomsEmpty **Strategy: (Equivalence class - checking to see if the correct output will be made when the user has rooms**	**Preconditions:** • **CleaningManagerUI has been loaded successfully.** • **The file `empty-rooms.txt and empty-logs.txt` exist.** • **testLoadValidFiles (Test 1) has passed.** **Steps:**	**The valid files have been loaded and the menu is displayed.** **The letter "F" is inputted and the user inputs that they want to see 4 rooms. The following should be outputted by the program.** ***No rooms have been cleaned.*** **The program then redisplays the menu prompting the user for another option.**	• The CleaningUI successfully loads, and the file names are entered into the program. • A message appears showing the files have been loaded and the menu displaying the different options and the choice for allowing to

and logs that are empty)	**1. Type the name input/empty-rooms.txt into the command line for rooms file.** **2. Type the name input/empty-logs.txt into command line for cleaning logs.** **3. Type in the letter "F" for frequently cleaned rooms** **4. Type in the number 4**		pick an option shows up. • The letter "F" is entered to pick the view most frequently cleaned rooms option. • The program then asks how many rooms should be displayed and the number 4 is entered. • The program outputs that no rooms have been cleaned since the files contain empty logs and rooms. • The menu is then redisplayed and another option can be made. **Results Match Expected (Success)**
testID(Test 5):testReportOfCleanings **Strategy: (Equivalence class- checking to see if each room that has been**	**Preconditions:** • **CleaningManagerUI has been loaded successfully.** • **The file `Input-rooms-1.txt and Input-cleaned-rooms-1.txt` exist.** • **testLoadValidFiles (Test 1) has passed.**	**The valid files have been loaded and the menu is displayed.** **The letter "R" is inputted. The following should be outputted by the program.** ***Room Report [*** • ***Attic was cleaned on [*** ○ ***(never cleaned)***	• The CleaningUI successfully loads, and the file names are entered into the program. • A message appears showing the files have been loaded and the menu displaying the different options and the

clean be reported by date.	**Steps:**
	
	1. Type the name input/Input-rooms-1.txt into the command line for rooms file.
	
	2. Type the name input/Input-cleaned-rooms-1.txt into the command line for cleaning logs.
	
	3. Type in the letter "R" for report of cleanings by room.
		• ***Bathroom was cleaned on [***
		  ○ ***(never cleaned)***
		• ***Bedroom was cleaned on [***
		  ○ ***11/08/2022 12:00:00***
		  ○ ***10/10/2022 08:15:34***
		  ○ ***07/04/2022 07:04:22***
		• ***Closet was cleaned on [***
		  ○ ***05/08/2022 07:32:10***
		  ○ ***03/14/2022 03:14:15***
		  ○ ***02/21/2022 00:00:00***
		• ***Game Room was cleaned on [***
		  ○ ***(never cleaned)***
		• ***Gym was cleaned on [***
		  ○ ***10/26/2022 23:59:59***
		  ○ ***10/26/2022 04:17:22***
		• ***Hallway was cleaned on [***
		  ○ ***12/31/2022 00:00:00***
		  ○ ***01/01/2022 03:17:22***
		• ***Living room was cleaned on [***
		  ○ ***10/31/2022 05:01:21***
		• ***Lunch Room was cleaned on [***
		  ○ ***12/25/2022 02:22:22***
		  ○ ***07/14/2022 04:51:57***
		  ○ ***06/19/2022 06:01:01***
		  ○ ***02/14/2022 04:01:22***
		• ***Office was clean on [***
		  ○ ***04/20/2022 05:30:59***
		• *]*
		The program then redisplays the menu prompting the user for another option.
			choice for allowing to pick an option shows up.
			• The letter "R" is entered to pick the view room report.
			• The program outputs the rooms and each date they were cleaned in the order as the expected.
			• The menu is then redisplayed and another option can be made.
			
			Results Match Expected (Success)

testID(Test 6):testReportOfCleaningsEmpty	**Preconditions:** • **CleaningManagerUI has been loaded successfully.**	**The valid files have been loaded and the menu is displayed.** **The letter "R" is inputted. The following should be outputted by the program.**	• The CleaningUI successfully loads, and the file names are entered into the program.

Strategy: (Equivalence class - checking to see if the correct output will be made when the user has rooms and logs that are empty)	• **The file `empty-rooms.txt and empty-logs.txt` exist.** • **testLoadValidFiles (Test 1) has passed.** **Steps:** 1. **Type the name input/empty-rooms.txt into the command line for rooms file.** 2. **Type the name input/empty-logs.txt into the command line for cleaning logs.** 3. **Type in the letter "R" for report of cleanings by room.**	***No rooms have been cleaned.*** **The program then redisplays the menu prompting the user for another option.**	• A message appears showing the files have been loaded and the menu displaying the different options and the choice for allowing to pick an option shows up. • The letter "R" is entered to pick the view room report. • The program outputs that no rooms have been cleaned since the files contain empty logs and rooms. • The menu is then redisplayed and another option can be made. **Results Match Expected (Success)**
testID (Test 7): testVacuumBagLifeExactlyZero **Strategy: (Boundary value - Testing for when the vacuum bag**	**Preconditions:** • **CleaningManagerUI has been loaded successfully.** • **The file `Input-boundary-rooms-1.txt and Input-boundary-cleaned-rooms-1.txt` exists.**	**The valid files have been loaded and the menu is displayed.** **The letter "V" is inputted. Then the date "12/31/2021 23:59:59" is entered. The following should be outputted.** ***Vacuum Bag Report (last replaced 12/31/2021 23:59:59) Bag is due for replacement in 0 SQ FT***	• The CleaningUI successfully loads, and the file names are entered into the program. • A message appears showing the files have been loaded and the menu displaying the

life has reached 0 SQ FT.)	• **testLoadValidFiles (Test 1) has passed.** **Steps:** 1. **Type the name input/Input-boundary-rooms-1.txt into the command line for rooms file.** 2. **Type the name input/Input-boundary-cleaned-rooms.txt into the command line for cleaning logs.** 3. **Type in the letter "V" for estimated bag life.** 4. **Enter "12/31/2021 23:59:59" when prompted for the last vacuum bag replacement.** 5. **Check vacuum bag life.**	**The program then redisplays the menu prompting the user for another option.**	different options and the choice for allowing to pick an option shows up. • The letter "V" is entered to pick the view expected vacuum bag life. • The date is then entered and then the message is displayed showing the vacuum bag life has reached 0 SQ FT. • The menu is then redisplayed and another option can be made. **Results Match Expected (Success)**
testID (Test 8): testVacuumBagLife **Strategy: (Equivalence class - Testing for when the vacuum bag life has 2640 SQ FT remaining.)**	**Preconditions:** • **CleaningManagerUI has been loaded successfully.** • **The file `Input-boundary-rooms-1.txt` and `Input-boundary-cleaned-rooms-1.txt` exists.** • **testLoadValidFiles (Test 1) has passed.** **Steps:**	**The valid files have been loaded and the menu is displayed.** **The letter "V" is inputted. Then the date "6/29/2022 23:59:59" is entered. The following should be outputted.** ***Vacuum Bag Report (last replaced 6/29/2022 23:59:59) Bag is due for replacement in 2640 SQ FT*** **The program then redisplays the menu prompting the user for another option.**	• The CleaningUI successfully loads, and the file names are entered into the program. • A message appears showing the files have been loaded and the menu displaying the different options and the choice for allowing to

	1. **Type the name input/Input-boundary-rooms-1.txt into the command line for rooms file.** 2. **Type the name input/Input-boundary-cleaned-rooms.txt into command line for cleaning logs.** 3. **Type in the letter "V" for estimated bag life.** 4. **Enter "6/29/2022 23:59:59" when prompted for the last vacuum bag replacement.** 5. **Check vacuum bag life.**		pick an option shows up. • The letter "V" is entered to pick the view expected vacuum bag life. • The date is then entered and then the message is displayed showing the vacuum bag life has reached 2640 SQ FT. • The menu is then redisplayed and another option can be made. **Results Match Expected (Success)**
testID (Test 9): testVacuumBagLifePastZero **Strategy: (Equivalence class – Testing if the UI will output the correct message when the coverage is greater than the vacuum bag life)**	**Preconditions:** • **CleaningManagerUI has been loaded successfully.** • **The file `sample-rooms.csv and sample-logs.csv` exist.** • **testLoadValidFiles (Test 1) has passed.** **Steps:** 1. **Type the name input/sample-rooms.csv into the command line for rooms file.**	**The valid files have been loaded and the menu is displayed.** **The letter "V" is inputted. Then the date "6/21/2019 14:50:02" is entered. The following should be outputted.** ***Vacuum Bag Report (last replaced 06/21/2019 14:50:02) Bag is overdue for replacement!*** **The program then redisplays the menu prompting the user for another option.**	• The CleaningUI successfully loads, and the file names are entered into the program. • A message appears showing the files have been loaded and the menu displaying the different options and the choice for allowing to pick an option shows up.

	2. **Type the name input/sample-logs.csv for cleaning logs.** 3. **Type in the letter "V" for estimated bag life.** 4. **Enter "6/21/2019 14:50:02" when prompted for the last vacuum bag replacement.** 5. **Check vacuum bag life.**		• The letter "V" is entered to pick the view expected vacuum bag life. • The date is then entered and then the message is displayed showing the vacuum bag is over due for a replacement. • The menu is then redisplayed and another option can be made. **Results Match Expected (Success)**
testID(Test 10): testVacuumBagLifeInvalidDate **Strategy:** **(Unexpected input/Error Handling – Check to see if the program will output to the user that the date they entered is invalid and that they need**	**Preconditions:** • **CleaningManagerUI has been loaded successfully.** • **The file `sample-rooms.csv and sample-logs.csv` exist.** • **testLoadValidFiles (Test 1) has passed.** **Steps:** 1. **Type the name input/sample-rooms.csv into the command line for rooms file.** 2. **Type the name input/sample-logs.csv for cleaning logs.** 3. **Type in the letter "V" for estimated bag life.**	**The valid files have been loaded and the menu is displayed.** **The letter "V" is inputted. Then the date "July/Fourth/2021 19:29:33" is entered. The following should be outputted.** ***Date & time must be in the format: MM/DD/YYYY HH:MM:SS*** **The program then redisplays the menu prompting the user for another option.**	• The CleaningUI successfully loads, and the file names are entered into the program. • A message appears showing the files have been loaded and the menu displaying the different options and the choice for allowing to pick an option shows up. • The letter "V" is entered to pick the view expected vacuum bag life.

to format it correctly)	**4. Enter "July/Fourth/2021 19:29:33" when prompted for the last vacuum bag replacement.** **5. Check vacuum bag life.**		• The date is then entered and the program outputs that the date is formatted incorrectly and that the date should be formatted as MM/DD/YYYY HH:MM:SS. • The menu is then redisplayed and another option can be made. **Results Match Expected (Success)**
testID (Test 11): testCloseCleaningManager **Strategy: (Equivalence class - The CleaningManager successfully closes out.)**	**Preconditions:** • **The CleaningManager is currently running.** **Steps:** **1. Type the letter "Q" when the program prompts for files.**	**The CleaningManagerUI should close when the letter "Q" is entered, and the following exit message should appear.** ***Thank you for using the Cleaning Manager.***	• The CleaningUI successfully loads. The letter "Q" is entered and then the CleaningUI closes with the exit message. **Results Match Expected (Success)**