Lilli Lewis and Maritza Mozo
CSC-161-02 for April 28th, 2023

Test Plan for Project 4: Tech Support Ticket Tracker

The program ticket.c prompts the user to pick one of the seven available options in a menu. Each option will allow the user to perform a different task. Option 1 will allow the user to create a new tracking ticket, option 2 will allow the user to assign a new ticket to a technician for handling, option 3 will allow the user to mark a ticket complete, option 4 will allow the user to print a list of new tickets, option 5 will allow the user to print a list of assigned tickets, option 6 will allow the user to print a list of completed tickets, and option 7 will allow the user to exit the program. At the start of the main program, the function printMenu will be called to allow the user to see the menu options before being prompted to select one. After the user is finished selecting the options they want to perform, the program will then return to the menu where the user will choose to exit the program, and it will exit the program successfully.

===

Test Plan for Project 4: Tech Support Ticket Tracker

===

The test plan will test valid and expected inputs (assuming that the user selects an option 1 to 7 from the menu) to demonstrate that the program runs correctly. It will also include testing the edge cases (printing out a list when it is empty and ticket options that do not exist at the time they request it) The test plan will also test when the user selects an option number that is not on the menu.

Test cases:

1. Test case 1: The user selects option 1 then option 4 before exiting the program with option 7.
   First, the program will print out its menu for the user to select an option. After choosing option 1, the program will create a new tracking ticket and confirm to the user that it was created. After its appropriate confirmation is printed, the program will then print out the menu again, where the user selects option 4. Since option 4 will print out a list of new tickets, we will be able to see if the new ticket appears on the list. If it does, we will know that the function worked. The program will then return to the menu where the user will choose to exit the program, and it will exit the program successfully with a friendly message.

2. Test case 2: The user selects option 1, 2, and then 4 and 5 on the menu before exiting the program to show that the tickets are located in the correct lists.
   First, the program will print out its menu for the user to select an option. After choosing option 1, the program will create a new tracking ticket and confirm to the user that it was created. After its appropriate confirmation is printed, the program will print out the menu again, where the user selects option 2 to assign this ticket to a technician for handling. After this is completed, the program will confirm to the user that it was assigned and go back to the menu again, where the user will select option 5. Since option 5 will print out a list of assigned tickets, we should see the ticket that was created in this test case appear on the list. The program will then return to the menu where the user will choose to exit the program, and it will exit the program successfully with a friendly message.

3. Test case 3: The user selects option 1, 2, 3 and then then 4, 5, 6 on the menu before exiting the program to show that the tickets are located in the correct lists.
   First, the program will print out its menu for the user to select an option. After choosing option 1, the program will create a new tracking ticket and confirm to the user that it was created. After its appropriate confirmation is printed, the program will print out the menu again, where the user selects option 2 to assign a ticket to a technician. Then, the user will select option 3 to mark this ticket as complete. After the program marks it as complete, the program will confirm to the user that it was marked as completed and go back to the menu again, where the user will select option 6. Since option 6 will print out a list of assigned tickets, we should see the ticket that was created in this test case appear on the list. The program will then return to the menu where the user will choose to exit the program, and it will exit the program successfully with a friendly message.

4. Test case 4: The user selects option 4, 5, and 6 before exiting the program. The user will also choose option 2 and option 3 to show what happens when there are no new tickets or assigned tickets.
   First, the program will print out its menu for the user to select an option. Since the user does not create any tracking tickets in this test case, there should be none that print out to any of the three lists. That said, when the user selects option 4, the program will print out an empty list and return to the menu. The program will do the same when options 5 and 6 are selected. Then the user will choose option 2 in which it will print out a message stating there are no new tickets to be assigned. Then the user will choose option

3 in which the program will print out a message stating there are no assigned tickets that can be marked as completed. The program will then return to the menu where the user will choose to exit the program, and it will exit the program successfully with a friendly message.

5. Test case 5: The user selects option 1 three times and then option 4 before exiting the program.
   First, the program will print out its menu for the user to select an option. After choosing option 1, the user will create a new tracking ticket before the program returns back to the menu. The user will repeat this two more times, creating a total of three new tracking tickets. With each creation of a new tracking ticket, the program will confirm that the new tracking ticket was created and return to the menu. The user will then select option 4, where the program will print a list of new tickets. That said, when this list of new tickets is printed out, we will see that the list consists of the three new tracking tickets that were created by the user. The program will then return to the menu where the user will choose to exit the program, and it will exit the program successfully with a friendly message.

6. Test case 6: The user selects option 1 three times, then option 2 three times, and then option 4 and option 5 to show the tickets are in the correct lists before exiting the program.
   First, the program will print out its menu for the user to select an option. After choosing option 1, the user will create a new tracking ticket before the program returns back to the menu. The user will repeat this two more times, creating a total of three new tracking tickets. With each creation of a new tracking ticket, the program will confirm that the new tracking ticket was created and return to the menu. The user will then select option 2, three different times to assign all three new tickets to a technician for handling. With each new assignment of a new tracking ticket, the program will confirm that the new tracking ticket was assigned to a technician and return to the menu. Afterwards, the user will select option 5 from the menu to print out a list of assigned tickets. When the program prints out this list, we will see that the three tickets created in this test case are in the list. The program will then return to the menu where the user will choose to exit the program, and it will exit the program successfully with a friendly message.

7. Test case 7: The user selects option 1 three times, then option 2 three times, then option 3 three times, and then option 4, option 5, and option 6 to show the tickets are in the correct lists before exiting the program.

First, the program will print out its menu for the user to select an option. After choosing option 1, the user will create a new tracking ticket before the program returns back to the menu. The user will repeat this two more times, creating a total of three new tracking tickets. With each creation of a new tracking ticket, the program will confirm that the new tracking ticket was created and return to the menu. The user will then select option 2, three different times to assign all three new tickets to a technician for handling. With each new assignment of a new tracking ticket, the program will confirm that the new tracking ticket was assigned to a technician and return to the menu. The user will then select option 3 three different times to mark all three new tickets as complete. They will be able to mark tickets as completed in any order. In this case we moved the tickets in the following order 2, 3, 1. With each completion of a new tracking ticket, the program will confirm that the new tracking ticket was marked as complete and return to the menu. Afterwards, the user will select option 6 from the menu to print out a list of completed tickets. When the program prints out this list, we will see that the three tickets created in this test case are in the list. The program will then return to the menu where the user will choose to exit the program, and it will exit the program successfully with a friendly message.

8. Test case 8: The user selects option 1 three times, then option 2 two times, then option 3 one time, and then option 4, 5, 6 to print out all the different lists before exiting the program.
   First, the program will print out its menu for the user to select an option. After choosing option 1, the user will create a new tracking ticket before the program returns back to the menu. The user will repeat this two more times, creating a total of three new tracking tickets. With each creation of a new tracking ticket, the program will confirm that the new tracking ticket was created and return to the menu. The user will then select option 2 to assign two of the new tickets to a technician for handling and the program will confirm this to the user and return to the menu. The user will then select number 3 to mark another ticket as complete and the program will confirm this as well then return to the menu. The user will then select option 4, 5, and 6 separately to print out a list of new tickets, assigned tickets, and completed tickets. Since we left one of the three new tickets to match each category, each list should have one ticket in it corresponding to which one the user chose to put in which. The program will then return to the menu where the user will choose to exit the program, and it will exit the program successfully with a friendly message.

9. Test case 9 (Edge case): The user selects option 1, 2, 3, 1, 2 and then 4, 5, 6 before exiting the program.
   This test case will ensure that when the user creates and changes the status of a ticket in different orders, it will still end up in its appropriate list. First, the program will print out its menu for the user to select an option. After choosing option 1, the user will create a new tracking ticket before the program prints a confirmation message and returns back to the menu. The user will then select option 2 to assign one of the new tickets to a technician for handling and the program will confirm this to the user and return to the menu. The user will then select number 3 to mark the same ticket as complete and the program will confirm this as well then return to the menu. The user will then create a new ticket using option 1. Then, the user will then select option 2 to change the status of the now completed ticket to assigned. The program will confirm this change and return to the menu. Then, the user will choose options 4, 5, and 6 separately to print out a list of new tickets, assigned tickets, and completed tickets. Since we only had one ticket and changed it to be assigned by the end, only the list that prints out assigned tickets should have one ticket in it while the others have none. The program will then return to the menu where the user will choose to exit the program, and it will exit the program successfully with a friendly message.

10. Test case 10 (Edge case): The user selects option 2 or 3 when the list is empty.
    This test case will ensure that the program handles cases where the user attempts to assign a ticket or mark one as complete when there are no tickets available. First, the program will print out its menu for the user to select an option. The user will select option 2 or 3, in which the program should see that there are no existing tickets available to perform these tasks on. Thus, the program will simply print out a message to the screen notifying the user about this and return back to the menu where the user will choose to exit the program, and it will exit the program successfully with a friendly message.

11. Test case 11 (Edge case): The user selects option 1 three times, option 2 one time, then option 3, where the user tries to mark a ticket as complete that has not been assigned but exists in the new list and a ticket number that does not exist.
    This test case will ensure that the program handles cases where the user tries to mark a ticket as complete that has not been assigned but exists in the new list. First, the program will print out its menu for the user to select an option. The user will make three new tickets and only assign one of them to a technician. Then, the user will try to mark a ticket that is not on the assigned list as

complete in which the program will print out a message stating that the ticket is not located in the assigned list. This will work for ticket numbers higher and lower than the ones that exist and for tickets located only in the new list. The program will then return to the menu where the user will choose to exit the program, and it will exit the program successfully with a friendly message.

12. Test case 12 (Edge case with invalid number input): The user will select a number that does not exist in the menu.
    This test case will ensure that the program handles cases where the user attempts to choose an invalid option. First, the program will print out its menu for the user to select an option. The user will select an number that does not exist in the list, in which the program will simply print out a message to the screen notifying the user about this error and exit the program itself.

13. Test case 12 (Edge case with invalid character input): The user will select a number that does not exist in the menu.
    This test case will ensure that the program handles cases where the user attempts to choose an invalid option. First, the program will print out its menu for the user to select an option. The user will write a character, in which the program will simply print out a message to the screen notifying the user about this error and exit the program itself.

==

Conclusion

==

The program is expected to operate correctly provided valid input and give accurate output. It handles edge cases as well as invalid input that is an integer out of range 1-7 and it also handles the edge cases within the options and lists.