

## TEST CASES - Parts(A)

**TEST CASE 1:** View list of parts used on selected vehicle under an existing “diagnosis and repair” booking.

**Description:** This tests that for a selected vehicle with an existing booking a list of parts installed on that vehicle could be accessed. These records are represented in a table. In case the table is empty, this means that the selected vehicle required zero parts to repair or the vehicle has not been diagnosed yet i.e. it belongs to a future booking.

**Associated module requirements:** 1 and 2.

**Steps and User Inputs (if any):** The user selects a record from the the table which corresponds to a vehicle with an existing booking. Then, the user clicks on the **View Installed Parts** button.

The screenshot shows the 'PARTS' module interface. At the top, there are tabs for CUSTOMERS, VEHICLES, BOOKINGS, PARTS (selected), SPC, ADMIN, and LOGOUT. Below the tabs is a search bar with 'Search Records...' and a 'Search' button, and a 'Search by' dropdown menu. A 'Refresh' button is also present. The main content area is titled 'Vehicles with existing bookings:' and contains a table with the following data:

Vehicle ID	Booking ID	Booking Type	Booking Date	Customer		
				ID	First Name	Second Name
AIO8977	1	Repair	2016-11-16 13:00	6	Warren	Buffet
AKI9087	2	Maintenance	2016-11-17 09:00	6	Warren	Buffet
GHJ6753	3	Repair	2016-12-01 10:00	7	Ash	Ketchum
TAB2435	4	Maintenance	2017-01-04 16:00	8	Alexis	Sanchez
KJH7865	5	Maintenance	2017-01-12 09:00	5	Frank	Marshsall
TAB2435	6	Repair	2017-01-25 11:00	8	Alexis	Sanchez
KHH9087	7	Maintenance	2017-01-28 09:00	4	Dwayne	Johnson
AKI9087	8	Maintenance	2017-02-01 10:00	6	Warren	Buffet
TAB2435	9	Repair	2017-02-04 15:00	8	Alexis	Sanchez
UIO8976	10	Maintenance	2017-02-06 09:00	9	Adi	Dassler

On the right side of the table, there is a sidebar with the following elements:

- A button labeled 'Past/Future Bookings'.
- A button labeled 'Bookings'.
- A button labeled 'Parts Installed' with a blue arrow pointing to a button labeled 'Installed'.

**Expected Outcome:** A window pops up with a table representing the parts used on the selected vehicle. Also, details, such as description and cost, about each part shall be represented in that table.

**Actual Outcome:** List of parts installed on the selected vehicle along with details about the booking and the customer.

**Result:** Passed.

**TEST CASE 2:** Select a part used and list details.

**Description:** Details such as ID (unique number identifying each stock item in the garage), description and cost are represented in a table. Other details, such as the installation date and the warranty's expiration date of an installed part are also available for the user to view.

**Associated module requirements:** 6, 7 and 8

**Steps and User Inputs (if any):** The user should select a vehicle. Then, by clicking on **View Installed Parts** button, a window will pop up. To view the installation date and warranty expiration date, the user selects a part, from the table, and click on the **More Details** button. Dates will appear in the relevant position on the screen.

**Installed Parts**

Parts installed :

ID	Name	Description	Price
7	Hoods	Automatic hig...	75.0

**Record Details:**

Customer ID: 5  
 Customer Name: Frank Marshall  
 Vehicle ID: KJH7865  
 Booking ID: 5  
 Booking Type: Maintenance  
 Booking Date: 2017-01-12 09:00  
 Installation Date: 12/01/2017  
 Warranty Date: 12/01/2018

More Details

Add Edit Delete Close

**Expected Outcome:** Installation date and warranty expiration date appears upon selecting a part and clicking on the **More Details** button.

**Actual Outcome:** Details of selected part listed.

**Result:** Passed

**TEST CASE 3:** Search for a part to add to a vehicle by name.

**Description:** A user can add parts to the selected vehicle either by selecting a part from a table or by searching for a part.

**Associated module requirements:** 8

**Steps and User Inputs (if any):** The user should select a vehicle and then click on **View Installed Parts** button. Then, by clicking on the **Add** button, a window will pop up which has a table, a search bar and buttons. The user can either select a part directly from the table or searches for a part by name. Finally, the user shall click on the **Add Part** button to confirm his choice and successfully add the part.

**Expected Outcome:** After successfully adding a part, the user should be able to see that the part added appears in the parts installed table which will show up starting away after confirming his choice (i.e. Clicking on **Add Part** button).

**Actual Outcome:** Newly added part appears in the table labelled Parts Installed.

**Result:** Passed

**TEST CASE 4:** Add a part to a repair of the selected vehicle. Show that stock level reduced.

**Description:** After successfully adding a part to a vehicle. A record of this withdraw is kept in the database and it could be accessed using the software. Also, the stock level of that part could be checked before and after addition.

**Associated module requirements:** 4 and 5

**Steps and User Inputs (if any):** The user should select a vehicle and then click on **View Installed Parts** button. Then, by clicking on the **Add** button, a window will pop up which has a table, a search bar and buttons. The user can either select a part directly from the table or searches for a part by name. Finally, the user shall click on **the Add Part** button to confirm his choice and successfully add the part.

To check that the stock level reduced, the user should go to tab labelled **PARTS** in the main menu. Then, by clicking on the **Stock** button, a window will pop up showing the item withdrawals represented in a table on the bottom left side of the window. Also, the user shall notice that the stock level decreased by one in the table main table labelled Stock Items.

ID	Name	Description	Quantity	Cost
1	Handbrakes	High quality hydraulic handbrakes	20	50.0
2	Engines	Super powerful engine with varied displac...	20	200.0
3	Turbo	Super powerful turbo	20	100.0
4	Nitrous	Super powerful nitrous	20	100.0
5	Brakes	High quality brakes	20	50.0
6	Rear Mirrors	Clear vision mirrors	20	25.0
7	Hoods	Automatic high quality hoods	20	75.0
8	Bumpers	Protective high quality bumpers	20	125.0
9	Doors	High quality powered doors	20	150.0
10	Tail Lights	High quality LED lights	20	25.0

Deliveries and Withdrawals:		
Receive Delivery		
Deliveries:		
ID	Date	Quantity
1	2017-03-20	2
2	2017-03-17	6
3	2017-03-20	2
4	2017-03-20	3
5	2017-03-20	1
6	2017-03-17	3
7	2017-03-20	1

Withdrawals:		
ID	Date	Quantity
6	18/03/2017	1
3	23/03/2017	1
7	02/04/2017	1
10	30/03/2017	1
6	26/03/2017	1
1	21/03/2017	1

**Expected Outcome:** Stock level reduced by 1 to become 19 (initially was 20).

**Actual Outcome:** Stock level reduced by 1 to become 19.

**Result:** Passed.

**TEST CASE 5:** View current stock list.

**Description:** View the list of the 10 distinct items available in the garage.

**Associated module requirements:** 3

**Steps and User Inputs (if any):** From the **PARTS** tab, the user should click on the Stock button. A window will pop up will the stock list represented in a table.

**Expected Outcome:** List of stock items in the garage.

**Actual Outcome:** List of stock items in the garage along with all the details relating to each stock item included in the table.

**Result:** Passed.

**TEST CASE 6:** Add new stock item. Show stock of part increased.

**Description:** Adding new stock item (i.e. stock item delivery). And, shows that the stock level increased. Also, a record of this delivery will be kept along with the delivery date and the quantity delivered.

**Associated module requirements:** 5

**Steps and User Inputs (if any):** From the **PARTS** tab, the user should click on the **Stock** button. A window will pop up and then the user should select the desired stock item from the table and then clicks on the **Receive Delivery** button. A new window will pop up that has a field labelled Received Quantity. The user should input the delivered quantity and clicks on the **Confirm** button to confirm the delivery and update the database.

**Expected Outcome:** Stock level of the selected item increase by the amount specified by the user.

**Actual Outcome:** Stock level successfully increased and database is updated. Also, a delivery record could be seen in the Deliveries table.

**Result:** Passed.

**TEST CASE 7:** Edit the record for a stock item.

**Description:** Any detail, except of the item ID, could be edited. And changes will be updated in the database as well as in the software.

**Steps and User Inputs (if any):** From the **PARTS** tab, the user should click on the **Stock** button. A new window will pop up with a list of the stock items in the garage. The user, then, selects an item and clicks on the **Edit** button. A new window will pop up that has an editable form where the user can edit details about a stock item.

**Expected Output:** Any changes made to the stock item record will be successfully updated in the database and in the software.

**Actual Outcome:** Amendments updated in the software and in the database.

**Result:** Passed.

**TEST CASE 8:** Delete a stock item.

**Description:** Deleting a stock item i.e. when a stock item is no longer needed in the garage.

**Steps and User Inputs (if any):** From the **PARTS** tab, click on the **Stock** button. A new window will pop up. Select a stock item from the table labelled Stock Items then click on the button **Delete**. A confirmation window will pop up where the user should click **Ok** to confirm deletion.

**Expected Outcome:** Deleted item is no longer in the Stock Items table and database should be deleted as well.

**Actual Outcome:** Deleted stock item successfully removed from the system.

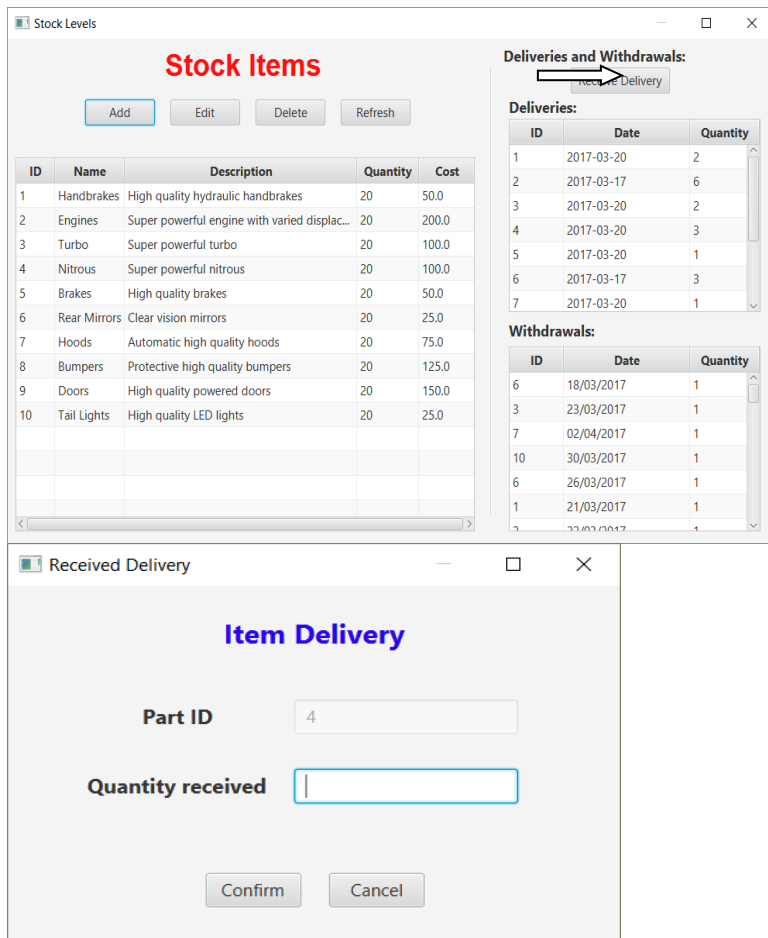
**Result:** Passed.

**TEST CASE 9:** Create a new part in the stock list and add items.

**Description:** Adding a new stock item and being able to add this item to a vehicle.

**Steps and User Inputs (if any):** To add a new stock item, the use should click on the **Stock** button while in the **PARTS** tab. And then from the popped-up window, the user should click on the **Add** button. A new window will pop up with a form to be filled by the user. The user should provide a name, a description, a cost and a quantity to the new stock item. All the fields are required for a successful addition of a new item.

Then to add this item to a vehicle, the user should go back to the **PARTS** tab and select a vehicle/booking for the table. Then, by clicking on the **View Parts Installed** button, a new window will pop up and then the user should click on the **Add** button. A new window will pop up where should be able to find the newly added stock item in the table. After selecting the item, the user should click on the **Add** button to confirm the addition of this part to the selected vehicle/booking.



**Expected Outcome:** New stock item seen on the Stock Items table. Newly added item could be successfully added to a vehicle.

**Actual Outcome:** Newly added item successfully added to the database and could be accessed anywhere in the software.

**Result:** Passed.

**TEST CASE 10:** Logout, close application and log back in. Show created record is persistently stored.

**Description:** After adding a stock item to the garage/adding a part to a vehicle/booking. Records should be stored in the database and the user should be able to see these changes after logging out and then logging back in again.

**Steps and User Inputs (if any):** From the main screen, the user should click on the **Logout** button at the top right corner of the screen. Then, in the login screen, the user should input, for an admin system user, *10003* in the username field and "*password*" in the password field, or, for a normal system user, *10005* in the username field and "*password*" in the password field. Then, from the **PARTS** screen, the user should select a vehicle/booking and then clicks on the **View Installed Parts** button.

**Expected Outcome:** New added stock item should be seen in the Parts Installed of the selected vehicle/booking.

**Actual Outcome:** Record persistently stored in the database and successfully retrieved to the system.

**Result:** Passed.

