System Testing - Donut View
System should act as a cafe interface and be able to take in orders, present their totals, list the orders, and remove orders.

Test Case #	Requirement	Test Description and Input Data	Expected Result/Output
1	The system should add any number and type of donut to the order when "✓" is pressed	● Select type as "Yeast Donut" ● Select flavor as "Matcha" ● Select quantity as "2" ● Press "✓"	Current Order should display "2 Yeast Donut (Matcha)" and subtotal should be $3.18
2	The system should do nothing when no donut flavor selected and "✓" is pressed	● Select any type (prefilled) ● Select any quantity (prefilled) ● DO NO select any flavor ● Press "✓"	Nothing
3	The system should remove a donut from the orders list only if the exact same type/flavor/quantity is selected and "X" is pressed	● Select 2 matcha yeast donuts ● Press "✓" ● Add any other donuts as desired ● Select 2 matcha yeast donuts ● Press "X"	First instance of "2 Yeast Donut (Matcha)" should disappear from Current Order
4	The system should do nothing when attempting to remove a donut not on the orders list (pressing "X")	● Add any order of donuts to the Current Order ● Select any type, flavor and quantity whose combination is NOT in Current Order ● Press "X"	Nothing
5	The system should change the donut image when a new donut type is selected	● Select type as "Cake Donut" from "Yeast Donut"	Image should change from a tan donut to a donut with pink frosting
6	The system should add the donut order to the basket when "Add to Basket" is pressed	● Add any order of donuts to the Current Order ● Press "Add to Basket"	Donut window should close, basket window should display the donut list

7	The system should do nothing when "Add to Basket" is pressed but the order is empty	● Press "Add to Basket"	Nothing

System Testing - Coffee View
System should act as a cafe interface and be able to take in orders, present their totals, list the orders, and remove orders.

Test Case #	Requirement	Test Description and Input Data	Expected Result/Output
1	The system should update the order when any new number, add-in or type of coffee is selected	● Open "Coffee" tab ● Toggle between different size options ● Add or remove add-ins with check boxes ● Select various quantities for coffee order	Can observe that "Subtotal" field in button left adjusts price with every change
2	The system should add the coffee order to the basket when "Add to Basket" is pressed	● Add any order of coffee and click "Add to Basket"	Coffee window closes Added item will be visible under "Current Basket" tab

System Testing - Basket View
System should act as a cafe interface and be able to take in orders, present their totals, list the orders, and remove orders.

Test Case #	Requirement	Test Description and Input Data	Expected Result/Output
1	The system should remove the selected item from the basket when "Remove Item" is pressed and an item is selected	● Add any order of donuts or coffee to the current order and "Add to Basket" ● Open basket and select item with mouse cursor ● Click "Remove Item" button	Selected item should be removed from basket with rest of order intact
2	The system should do nothing when "Remove Item" is pressed	● Add any order of donuts or coffee to the current	No change

	but no item is selected	order and "Add to Basket" ● Open basket ● Click "Remove Item" button	
3	The system should clear all items from basket when "Clear" is pressed	● Add any order of donuts or coffee to the current order and "Add to Basket" ● Open basket ● Click "Clear" button	Basket should be emptied with all items removed
4	The system should do nothing when "Clear" is pressed but no items in basket	● Open basket ● Click "Clear button"	No change
5	The system should add the current order to the list of Store Orders when "Finalize Order" is pressed	● Add any order of donuts or coffee to the current order and "Add to Basket" ● Open basket ● Click "Finalize Order" button	Basket window closes and order is added Order may be found under "Store Orders"
6	The system should do nothing when "Finalize Order" is pressed but the order is empty	● Open basket ● Click "Finalize Order" button	Basket window closes and no order is added Order will not be found under "Store Orders"

System Testing - Store Orders View System should act as a cafe interface and be able to take in orders, present their totals, list the orders, and remove orders.			
Test Case #	Requirement	Test Description and Input Data	Expected Result/Output
1	The system should display the corresponding order's contents when its number is selected in "Order Number"	● Using Donut and Coffee pages, create an order ● Using Basket page, add that order to the list of Store Orders ● Repeat as necessary ● Using Store Orders page, select any order using the Order Number drop down	List of items in the corresponding store order in the list view

2	The system should delete the selected order from the list of Store Orders when "Cancel Order" is pressed	● Using Donut and Coffee pages, create an order ● Using Basket page, add that order to the list of Store Orders ● Repeat as necessary ● Using Store Orders page, select any order using the Order Number drop down ● Press "Cancel Order"	Canceled order number should no longer be available in the Order Number drop down
3	The system should do nothing when "Cancel Order" is pressed but no order is selected	● Using Store Orders page, select any order using the Order Number drop down ● Press "Cancel Order"	Nothing
4	The system should export all orders in the Store Orders list to a file when "Export" is pressed	● Using Donut and Coffee pages, create an order ● Using Basket page, add that order to the list of Store Orders ● Repeat as necessary ● Using Store Orders page, press "Export" OR ● Using Store Orders page, press "Export"	A new file should be created (or old file overwritten) in the repository with contents of the current Store Orders list