Test Cases

Test Case #1: A manager wants to access the full list of products.

Step 1: Open the application

Step 2: Run the application

Step 3: The index page (a view) will load, taking you to the html home page with text defined in the Home controller.

Step 4: Go to the end of the URL for the website and type “/api/Product.”

Step 5: The system will define and gather the format of data from the Product class, returning a list of the defined variables from the Product control class.

Step 6: Wait for the products to be shown on the website

Step 7: All of the products are displayed

Test Case #2: A manager wants to look at an individual item with an id of 1001.

Step 1: Open the application

Step 2: Run the application

Step 3: The index page (a view) will load, taking you to the html home page with text defined in the Home controller.

Step 4: Go to the end of the URL for the website and type “api/Product?id=1001.”

Step 5: The system will define and gather the format of data from the Product class, returning the individual variable that was defined in the URL, in accordance with the data of products from the Product control class.

Step 6: Wait for the product to be shown on the website

Step 7: The individual product is displayed

Test Case #3: A customer wants to look at the shipping cost for an item

Step 1: Open the application

Step 2: Run the application

Step 3: The index page (a view) will load, taking you to the html home page with text defined in the Home controller.

Step 3: Fill the zip code of the customer into the textbox

Step 4: Press the Enter button

Step 5: The system will look in the Home Controller to see what text should be replaced, and the view for zip code will display the message

Step 6: The code in JavaScript will determine whether a $5 or $10 shipping fee will be added; this result will be printed with the message from the Home controller.

Step 6: The shipping cost for the item is displayed under the textbox and Enter button in a div.

Test Case #4: A customer wants to know how many hours have been spent in making a product

Step 1: Open the application

Step 2: Run the application

Step 3: The index page (a view) will load, taking you to the html home page with text defined in the Home controller.

Step 4: Fill in the radio buttons for the name and size of the item

Step 5: Press the button that says, “Calculate Labor Cost.”

Step 6: The system will look in the Home controller to see what text should be replaced, and the view for labor cost will display the message.

Step 7: The code in JavaScript will print out a message (according to the name and size that was selected); this code will be printed with the message from the Home Controller.

Step 8: The number of hours will be displayed underneath the button in a div

Test Case #5: A customer wants to find the price of an item

Step 1: Open the application

Step 2: Run the application

Step 3: The index page (a view) will load, taking you to the html home page with text defined in the Home controller.

Step 4: Fill in the radio buttons for the name and size of the item

Step 5: Press the button that says “Calculate Price”

Step 6: The system will look in the Home controller to see what text should be replaced, and the view for price will display the message.

Step 7: The code in JavaScript will print out a message (according to the name and size that was selected); this code will be printed with the message from the Home controller

Step 8: The price will be displayed under the button

Test Case # 6: A customer wants to buy an item

Step 1: Open the application

Step 2: Run the application

Step 3: The index page (a view) will load, taking you to the html home page with text defined in the Home controller.

Step 3: Fill in the radio buttons for the name, size, and color of the product

Step 4: Press the button that says “Print Properties”

Step 5: Other functions are called, including the labor cost and price of the product; these functions run

Step 6: The properties for ID, name, description, size, color, labor cost, and price are displayed in their respective locations (divs), according to the radio buttons that are selected with the code to output created in JavaScript

Step 7: the product’s id is incremented by 1 and printed in its respective div.