Testing plan for Orinoco Website

Introduction

Testing plans are crucial for developing websites as it grounds core features and functions accommodating the company/brand's unique needs and service. In this testing plan, it presents the Orinoco website's certain core backend functions and features including Orinoco's Camera. It will begin with the website's structure and features to test containing specific details and functions included in the page.

Website's structure

Orinoco's Website starts with a home page (index.html) presenting all featured products, shown in individual elements. Other details include:

- Specific details of each camera including name, image, description and price (More details below).

- Two buttons, More info and Add to cart, which directs customers to either more information of the chosen product or add it to cart (more details below).

- A footer that consist of the Orinoco's social media platform presented as icons

The next is the product page that contains further details of each featured camera. In this page, it only focus individual camera. Its features include:

- Further providing details such as camera description and available type of lenses that customers could pick

- Also two buttons, one being see other items (directing customers back to home page) and add to cart function (more details below).

The third page is the cart in which contains all the user's chosen products for purchase. It includes:

- A list of the name, image, price of each chosen products and a button that removes a specific item

- Subtotal of all chosen products and "buy it now"button

- A contact form that appears when users click "buy it now" containing inputs for first name, last name, address, city and email.

Lastly, confirmation page that confirms that the order have been processed containing a unique ID that reference the order.

Features to Test

1. Users can add chosen products to the cart and adds the chosen item's price towards the total cost

- Frontend

- Button (Add to cart) that allows user to add an item to the cart. In homepage, pressing the button stores the default camera lense.

- Chosen products appears in the cart page including adding the price to the total cost.

- Backend

- Stores camera details (e.g. name, price, description, etc.) into local storage using appropriate scripts (JSON.stringify)

2. Generating DOM elements for the website including the information within

- Frontend

- Creating appropriate DOM elements for showing products (camera), namely semantic tags; divs, section, etc., and text information

- Backend

- Requesting data namely camera details (names, description, image, etc.) from a API to be included in DOM

Information about the featured products

3. Remove items from the cart through clicking a button and after purchasing the products

- Frontend

- Takes out unwanted items from the cart

- Backend

- If local storage (cart) is empty, pressing ‘home’ button would delete cart storage.

- After purchasing products, local storage will be cleared

4. Hides specific section (contact forms and cart list) by clicking certain buttons (‘buy it now’ and ‘remove’ buttons)

- Frontend

- Certain webpage section hides/appears following certain parameters

- Backend

- In the cart page, contact form DOM will only appear after pressing ‘Buy it now’. Likewise, when users click remove button in the cart list which presents users information that the cart is empty.

5. Validation of user input in the details form in buying chosen products

- Frontend

- Confirmation button is disabled if inputs have invalid characters

- Input border turns into red while a small text appears when users include invalid characters. For instance, typing numbers in the first name input will change the border into red and a text will appear suggesting a valid example.

- Backend

- Each input contains certain rules for users input such as only letters for names and email input should include a text, @ sign and .com.

6. User can send contact details to API (after purchasing products) and then receives a unique code (product ID) for the order

- Frontend

- An alert appear indicating that the order is being processed

- Backend

- User’s input contact details are sent to an assigned API through “POST” function and sends back certain aspect of that sent data. In this case, a unique code (product Id, the user input name and the total cost of the whole purchase).

Features not to test

- Footer icons have no direct links
- Products featured cannot be changed
- Less emphasis on improving the website’s design