### **Project: E-commerce Website Test Automation with Selenium Using POM and Extent Reports**

**Website Under Test:** [Health Hub E-commerce Website](https://health-hub-rust.vercel.app/)

### **Objective:**

Develop an automated testing framework using **Selenium** and the **Page Object Model (POM)** for the Health Hub e-commerce platform. Integrate **Extent Reports** to provide detailed reporting of test execution results.

### **What You Will Do**

#### **1. Set Up Selenium with Extent Reports:**

* Install and configure **Selenium WebDriver** in your project.
* Add dependencies for Extent Reports to your project (e.g., Maven dependency for Extent Reports).
* Configure the extent-config.xml file for customizing report layout.

#### **2. Implement POM:**

* Create dedicated page classes for different pages on the website:  
  + **HomePage**: Automate search bar and navigation to categories.
  + **LoginPage**: Handle login/signup functionalities.
  + **ProductListingPage**: Automate filtering, sorting, and selecting products.
  + **CartPage**: Test adding/removing products and updating quantities.
  + **CheckoutPage**: Automate the checkout flow.
  + **ContactUsPage**: Test the feedback submission form.

#### **3. Write and Execute Test Cases:**

* Write reusable test cases leveraging POM.
* Examples of test cases:  
  + Verify user login with valid and invalid credentials.
  + Validate product search and filtering functionality.
  + Test adding/removing products in the cart and applying discounts.
  + Automate checkout with and without creating an account.

#### **4. Generate Extent Reports:**

* Integrate Extent Reports into your framework:  
  + Capture test case execution results (Pass/Fail/Skip).
  + Include screenshots for failed test steps.
  + Provide timestamps and detailed logs.

### **Key Features to Test**

1. **Homepage:**
   * Validate navigation, banners, and product categories.
2. **User Authentication:**
   * Test login/signup workflows.
3. **Search and Filtering:**
   * Validate search functionality and product filtering/sorting options.
4. **Product Details:**
   * Verify product specifications and "Add to Cart" functionality.
5. **Cart and Checkout:**
   * Test cart interactions and checkout processes.
6. **Contact Us:**
   * Validate feedback form submission.
7. **Footer Links:**
   * Verify redirection to pages such as "Terms of Use" and "Privacy Policy."

### **Documentation Tasks**

#### **1. Test Plan:**

Define the scope, objectives, test strategy, and test environment.

#### **2. Test Scenarios and Cases:**

Write test scenarios and detailed test cases for the above features.

#### **3. Bug Report:**

Log defects found during testing with:

* Steps to reproduce.
* Severity level.
* Screenshots of failed test cases.

#### **4. Summary Report:**

Provide a high-level overview of test results:

* Total test cases executed.
* Passed/failed/skipped tests.
* Bugs identified.

#### **5. Mind Map:**

Create a visual representation of the testing process.