**Task description: Automate user registration process of e-commerce website with TestNG and Selenium**

Steps to Automate:

1. Open <https://demo.nopcommerce.com>
2. Create 2 Test case to test Registration Functionality (1 positive and 1 negative)
3. Create 2 Test case to test Login Functionality (1 positive and 1 negative)

**General guidelines:**

1-Use soft assertion instead of hard assertion and put an assertion error message upon failure.

2-Follow the following naming conventions:

* Classes names should follow the upper camel case ex: GsmServiceAccountsSearch.
* Webelements and methods should follow the lower camel case ex:createAccountButton.
* Packages names should be in lower case ex: adsl.
* Follow the page object model, i.e., each web page should be modeled in a separate POM class.
* WebElements should be descriptive, e.g., loginButton,userNameTextField.
* Methods should be descriptive, e.g., clickOnLoginButton,enterTextInUserNameTextField.

3- Use explicit wait before every action.

4- Use maven to handle external dependencies.

5-Use relative xpath to find elements instead of id (for practice purposes).

6- Use Page Object Model Design Pattern.

**Useful links:**

[Soft Asserts and Hard Asserts in TestNG - Total-QA](http://total-qa.com/soft-asserts-hard-asserts-testng/)

[Selenium Wait – Implicit, Explicit and Fluent Waits (guru99.com)](https://www.guru99.com/implicit-explicit-waits-selenium.html)

[XPath in Selenium: How to Find & Write Text, Contains, OR, AND (guru99.com)](https://www.guru99.com/xpath-selenium.html#11)