GET Group Holdings Ltd. task

Project Overview

- This test automation project follows the Page Object Model (POM) design pattern, implementing Object-Oriented Programming (OOP) concepts. The project is structured into two main parts:

  1-a package for pages to implement needed methods

  2-a folder for tests to implement desired test actions)

- Pages package contains two classes :
  1- registerPage class for implementing methods needed to interact with web elements in registration page
  2- successfullRegisterPage class for Handling interactions with the page that confirms successful registration.
- Test folder contains two packages :
  1- base test package that contains base test class with essential methods that contains three methods :
     - setup(): Initializes necessary resources before executing each test.
     - teardown(): Cleans up resources after each test execution.
     - takeScreenshot(): Captures a screenshot upon test failure.
  2- registerTest package that contains two classes Includes various scenarios:
     - SuccessfulSignUpTest class: Tests successful registration with valid data for all required fields.
     - BadScenariosTest class: Contains tests for scenarios with invalid inputs:
       1- First name and last name starting with a lowercase letter.
       2- First name being equal to last name.
       3- Invalid phone number input.

Assertion Handling

- Each test within the BadScenariosTest class includes an assertion to verify the presence of an error message below the register button, indicating "Please correct errors before submitting this form." The test fails if the expected error message does not appear, indicating a bug in the application.
- I put the assertion of the first bad scenario inside a try catch block to handle the assertion failure and quit the browser as it is required, the test passes and in this case no screenshot will be taken, so how to solve it?

Reporting and Error Handling

- Upon executing test cases, an Allure report is generated to provide detailed insights into test results and execution steps. Additionally, screenshots are captured for failed tests and stored in the screenshots folder within the project directory.