PROJECT EXERCISE FOR SELENIUM WITH JAVA COURSE ATTENDEES

Congratulations for completing the "BDD with Cucumber Java" and the "Selenium with Java" courses.

For maximizing the learning outcome and to help you with assessing your knowledge, we invite you to complete a "Project Exercise".

In the project exercise you need to make a working test automation solution for a few functionalities of an existing Demo App. Please upload the completed solution to GitHub and send the link to gaspar@specsolutions.eu for review.

The completed solution should fulfill the following criteria:

- It has to be based on the prepared project starting point at: https://github.com/specsolutions/20220728-SpecOverflowTests that is
    - It a Maven Java project, using Java 8 or newer
    - Uses Cucumber Java
    - Uses Selenium WebDriver with Chrome browser
- It has to contain at least 3 BDD scenarios
- Optional: It can also use page object classes (nice to have!)
- The project should not require any special setup to be able to run the tests
- All tests should pass
- The tests should support multiple test executions (they should pass for the second run as well)

EXERCISE DESCRIPTION

- Use the "Spec Overflow" demo application for the exercise. You can use the hosted version (see link below), but you can also host it for yourself (faster, allows destructive tests) by executing "start-demoapp.cmd" from the "DemoApp" folder of the project.
- The demo app implements a simple Q&A site, similar to Stack Overflow. You will be probably familiar with the features it provides.
- Choose a feature from the application and identify 1-2 business rules that the application fulfills.
- Write altogether 3-5 scenarios for the rule(s) that illustrate the behavior of the rules. Make sure you follow the BRIEF principles (business readable, real data, intention revealing, essential, focused, brief).
- Automate the scenarios with Cucumber Java and Selenium WebDriver

HOW TO DELIVER YOUR RESULT

- It is recommended to work on your own GitHub fork (copy) of the project starting point. Check the description https://github.com/specsolutions/20220728-SpecOverflowTests#how-to-create-your-own-test-project for details how this can be done. To get feedback, just send the link of your fork to the instructor.

- Alternatively, you can also send your results per email, but the project zip will be too big as an email attachment, so you need to upload it somewhere and send the link.
- **It is recommended to get feedback about your scenarios before you start automating them!**
- **Deadline: Deliver your complete solution by 4th August 2022, Thursday evening latest. Prepare to make fixes on 5th August, Friday.**

Sample features & rules

Here are a few features and rules from the application for inspiration. Feel free to use either these or find other ones.

- Feature: Asking questions
  - Rule: Only authenticated users can ask questions
  - Rule: The asked question should be valid (mandatory, etc.)
  - Rule: Similar questions to the one being asked are suggested
- Feature: Voting answers
  - Rule: The answers should be listed by votes descending
  - Rule: Only authenticated users can vote
  - Rule: Answer should move up/down right after voting if the vote changes the order
- Feature: Registration
  - Rule: Registration data should be valid
  - Rule: Password must be at least 4 characters long

Links

Access the Demo application online:

https://specflowmasterclassspecoverflowweb20220706204956.azurewebsites.net/