Preparation Guide for the Coding Exercises

Automation with Cucumber Java Tutorial

MACHINE PREREQUISITES

- Approximately 500MB free space is necessary in total (including size of dependencies).
- ***Java 8 JDK*** (or later) with ***Maven*** installed
- Optional: Java IDE (e.g. ***IntelliJ,*** **Eclipse** ***or Visual Studio Code***)

PREPARATION STEPS

- Clone the coding exercises from GitHub (https://github.com/specsolutions/20230530-ExpoQA-AutomationWithCucumberJava) or download it as zip and unzip it. **Avoid using shared drives!** We recommend using a folder with a shorter path to avoid reaching the maximum folder path limit of Windows.
- ***Now we are going to test your setup.*** For that we will use the B1 exercise as an example. Please perform the following steps carefully. If you find any problems, please inform us immediately at courses@specsolutions.eu, so that we can fix the problems in time.
    - We will refer the `mvn` command a few times below. If that does not work, you can also use the `mvnw` command instead (`mvnw` is an mvn wrapper that chooses the right version of `mvn`).
- *Download packages:* Download packages for the B1 exercise (this will ensure that all packages that are used by the exercises are downloaded to the machine)
    - On a command line shell, go to the `B1` folder in the downloaded repository.
    - Invoke: `mvn compile`
- *Test compilation and test execution:* Run tests of B1 exercise: there should be one passing test.
    - On a command line shell, go to the `B1` folder again.
    - Invoke: `mvn test`
    - The command should complete without errors and report “Tests run: 1, Failures: 0, Errors: 0, Skipped: 0”
    - Invoke: `mvn spring-boot:run`
    - The command should complete without errors and you should able to navigate to http://localhost:8019 and see a pizza website, with some pizzas on the menu.
- *Test IDE setup:* Open `B1` exercise in your IDE and run the demo app and the tests.
    - Open/import `pom.xml` from the `B1` exercise in your IDE.
    - Build solution. Make sure it builds successfully.

- Run tests (e.g. `CucumberApiTest` class) from IDE. You should get the same result as in Maven.
- Open `home.feature` file. You should see a feature file with syntax. If not: you need a Cucumber plugin.
- Run the sample application. You should able to navigate to http://localhost:8019.

CONGRATULATIONS, YOU ARE READY TO GO!