**Auden – QA Test**

**(Pre-interview Exercise for Test Engineer Candidates)**

**Part:1**

The following exercise will assess your ability to create a well-structured test automation framework. The exercise should only take 1-2 hours; don’t spend much longer than this. Please complete the following exercise C# or similar languages. If this exercise goes well, you may be asked to make further amendments to your code during a pair programming exercise with a member of the team. Please pick the language you think you will be best able to demonstrate your skill and experience.

Please use git to version control your exercise and submit your solution by sending us a link to your Github (or similar) account via your agent.

Exercise​: Prepare a test framework from scratch and use this to implement a number of tests against the following test website:

<http://automationpractice.com/index.php>

Please provide a set of automated UI tests for the following:

1. Create a new account

2. Login

3. Find the most expensive dress and add it to the cart

4. Log out and back in again, ensuring the dress is still in the cart Expected​:

A written explanation for the automation including reasoning for the tools you’re using

● The solution you provide has to run all tests with one command.

● The solution should to be based on C#/.Net and use Selenium Webdriver to interact with the web pages

● Ideally, a BDD framework (such as specflow) would be used to describe the test scenarios

● Page Object or an alternative pattern used to organise code

**Part 2: API testing**

Write a simple tests for below WSDL/ REST using Groovy test step

WSDL Address:

<http://currencyconverter.kowabunga.net/converter.asmx?WSDL>

Rest Address:

<http://currencyconverter.kowabunga.net/converter.asmx>

Automate below testcase using SOAPUI as tool (please provide us with readme document in details and make sure you use groovy scripting for assertions and also property transfers etc…)

Test Case 1

* Request 1 – USD to GBP - 1 request to convert USD to GBP
  + Assert conversion rate result exists
  + Assert value (0.5967)
  + Any other assertions that you think are appropriate
* Request2 – Another request to convert GBP to USD, however get USD from request1 by performing a ‘Get Data’ property request.
  + Assert conversion rate result exists
  + Assert value (1.6757)
  + Any other assertions that you think are appropriate

Note: Conversion rate varies from day to day so use only current day conversion rate to assert it

If you need any details or if you face any issues please do not hesitate to contact us