AMENDMENT HISTORY

Issue No.	Date	Background Information
V 1.0	11/02/2016	Issued
V 1.1	11/03/2016	Typo change

Contents

1 INTRODUCTION

This test exercise will be used as the basis to test the skills and proficiency of UI Development skills of the candidate. The successful completion of the exercise will be followed by a possible a face-to-face interview where possible or a telephonic discussion. The interview will be based around the solution that the candidate has provided and leading questions.

There are no right or wrong answers to the exercise, although general correctness and neatness of syntax and working of the program are expected.

The approach to the exercise should be original. Reference to the internet and any books or guides is permitted. However it is recommended that the candidates come up with original solutions. This will not only give an impression that the candidate knows UI programming but also allow the candidate to effectively defend their solution.

Cadidates are permitted the use of any opensource UI development frameworks. It is expected that the solution will be based on Java related technologies as the selected candidate will be working in a team of predominantly Java developers, however the use of other UI development frameworks is permitted and encouraged.

2 Deliverables

- A working programme as a solution to the exercise.
- Working unit tests to test the application classes.
- Working TDD and/or BDD tests should be supplied to test the UI flows.
- The solution should be delivered as a fully functional eclipse maven project. All dependent libraries should be configured through maven. The maven pom file should not contain any errors.
- This code should be self explanatory. Overly complicated code could limit the chances of selection for the next stage of the selection process.
- A supporting document to explain the solution and how to run the program should be supplied. Content of the document is more important than its formatting.

3 Time Permitted And Communication

3.1 Time Permitted

It should take 6-10 hours to complete this exercise so the exercise should submitted within 48 hours of receipt of the exercise.

3.2 Communication

Please don’t hesitate to ask questions if you are unsure about any aspects of the exercise.

4 The Exercise

Write a UI application to load, search for meanings of a word, add new words or update meanings to existing words in a dictionary.

The dictionary should be defined in an XML file. When the UI application is started the home screen will request the path to the Dictionary file to be loaded. The user should be able to browse to the file location and select the file to be loaded. The file can be called anything you like.

The format of the dictionary XML will be as follows

```
<dictionary>
  <entry word=”word1” meaning=”meaning of word1” />
  <entry word=”word2” meaning=”meaning of word2” />
  ………..
  …………
</dictionary>
```

If the XML is not in the correct format, the UI should show a suitable error on the screen.

Once the file is successfully loaded, the user should be presented with a menu on the UI with the following options

- Option 1: Search for the meaning of a word
- Option 2: Add a word
- Option 3: Exit

If option 3 is selected the UI will terminate.

If option 2 is be selected, the user will be presented with a screen where he/she will be asked to enter a new word in a text box followed by its meaning in a text area.

After the user presses Add Word button, the following action will be taken:

- If the word has non alphabetic characters in it, an error message will be displayed in red on the top of the dialogue box and the word and it’s meaning will be displayed in the respective text box and text area as they had been entered.

- If the word already exists in the dictionary, a pop up message will appear saying, ‘The word already exists, do you wish to overwrite it?’ There should be two buttons Overwrite and Cancel.

    - If Overwrite is pressed, the word should be overwritten in the dictionary.

    - If Cancel is selected, the user should be returned back to the Menu screen.

- If the word does not exist, it should be added to the dictionary XML file. The user should be returned back to the menu and a confirmation message should be displayed in green above the menu saying that the word was successfully added.

If option 1 is selected, the user will be taken to the next screen where a text box will be presented to enter the word and two buttons should be displayed,

1. Search
2. Return to Menu

- If Search is pressed, the word will be searched in the dictionary:
    - If the word is found the meaning will be displayed below the textbox on the same screen.
    - otherwise the message ‘Word not in dictionary’ will be displayed below the textbox.

- If Return to Menu is selected, the user will be returned back to the Menu.