**Technical Exercise – TextAnalyzer – WordFrequencyCounter Descritpion**

There are four projects in the solutions:

1. TextAnalyzer Interface
2. TextAnalyzer Implementation
3. WPF Application
4. BDD Test

These all applications are implemented with NetFramework 4.5, C#, WPF and SpecFlow.

**1.Text Analyzer Interface**

**Property**

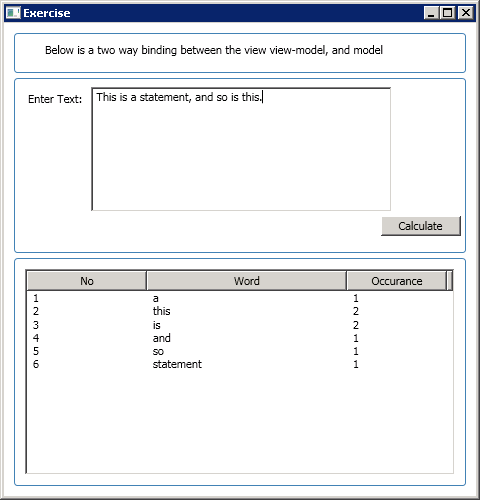
|  |  |  |
| --- | --- | --- |
| **Property Name** | **Type** | **Description** |
| Sentences | ConcurrentQueue<string> | In memory FIFO queue multi thread application |
| WordFrequecy | ConcurrentDictionary<string, int> WordFrequency | To store calculated word count. |

**Mehtod**

|  |  |  |
| --- | --- | --- |
| **Method Name** | **Parameters Type** | **Description** |
| AddSentence | String | To add a sentence to queue |
| Calculate | - | To calculate frequency word count |
| Reset | - | To clean previous data |

**2. TextAnalyzer WPF Application**

I have used MVVM pattern in WPF application, in order to separate out to decouple between UI and application logic. The main “TestAnalyzer” interface is injected to ViewModel using with Ninjet.



**SpacFlow Test**

In order to test the application logic, I have implemented “TestAnalyzer.Tests” project using with SpacFlow and Nunit. I divided the test case into the main group.

1. Product Acceptance Test

|  |  |
| --- | --- |
| Name | Description |
| Calculate Word Count | To test Calculate function |
| Adding a new sentence | To test concurrent queue |
| Reset Word Frequency Count List | To test rest function |
| Case sensitive | To check case sensitive count |

1. Negative Acceptance Test

|  |  |
| --- | --- |
| Name | Description |
| Blank character | Should not include in word count |
| Extra spaces between words | Should be trim and not include in counting |
| Extra spaces before and after words | Should be trim and not include in counting |
| Extra spaces after a word | Should be trim and not include in counting |

