**SIT323 Cloud Application Development**

**Trimester 2, 2018**

**Assessment Task 1 – Programming Task 1**

Student Name

Student ID Number

**Contents**

1. Introduction 999
2. Test Scenario 1 999
3. Test Scenario 2 999
4. Test Scenario 3 999
5. Test Scenario 4 999
6. Test Scenario 5 999
7. Test Scenario 6 999
8. Test Scenario 7 999
9. Test Scenario 8 999
10. Test Scenario 9 999
11. Test Scenario 10 999
12. Test Scenario 11 999
13. Test Scenario 12 999
14. **Introduction**

This is the test design document for Assignment 1.

1. **Test Scenario 1**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Description** | | | | |
| 1 | Check if a value is Boolean data type | | | | |
| Test Method | Method Tested | | | | |
|  | Validator.IsBooliean(String field, out Boolean aBoolean) | | | | |
| Test Case ID | Parameters | Expected Data | Actual Data | **Test Result** | **Test Comments** |
| 1 | "True” | Return value = true, out value = true | Return value = true, out value = true | passed |  |
| 2 | “False” | Return value = true, out value = false | Return value = true, out value = false | passed |  |
| 3 | “Error” | Return value = false | Return value = false | passed |  |

1. **Test Scenario 2**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Description** | | | | |
| 1 | Check if a value is integer data type | | | | |
| Test Method | Method Tested | | | | |
|  | Validator.IsInt32(String field, out Boolean aBoolean) | | | | |
| Test Case ID | Parameters | Expected Data | Actual Data | **Test Result** | **Test Comments** |
| 1 | "100” | Return value = true, out value = true | Return value = true, out value = 100 | passed |  |
| 2 | “-99999” | Return value = true, out value = false | Return value = true, out value = -99999 | passed |  |
| 3 | “Error” | Return value = false | Return value = false | passed |  |

1. **Test Scenario 3**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Description** | | | | |
| 1 | Check if a value is HEX colour code data type | | | | |
| Test Method | Method Tested | | | | |
|  | Validator.IsInt32(String field) | | | | |
| Test Case ID | Parameters | Expected Data | Actual Data | Test Result | **Test Comments** |
| 1 | "#00A3F9” | Return value = true, out value = true | Return value = true | passed |  |
| 2 | “#FFFFFF” | Return value = true, out value = false | Return value = true | passed |  |
| 3 | “ABCDEFG” | Return value = false | Return value = false | passed |  |

1. **Test Scenario 4**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Description** | | | | |
| 1 | Try to parse a key-value pair data string | | | | |
| Test Method | Method Tested | | | | |
|  | KeyValue.TryParse(String originalKeyValueData, String keyPattern, out KeyValue aKeyValue) | | | | |
| Test Case ID | Parameters | Expected Data | Actual Data | **Test Result** | **Test Comments** |
| 1 | Data = “A=2”, pattern = “A” | Return value = true, out value = true | Return value = true, out kv.key = “A”. kv.value = “2” | passed |  |
| 2 | Data = “Secret=Password”, pattern = “Secret” | Return value = true, out value = false | Return value = true, out kv.key = “Secret”. kv.value = “Password” | passed |  |
| 3 | Data = “Key:Value”, pattern = “Key” | Return value = false | Return value = false | passed |  |

1. **Test Scenario 5**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Description** | | | | |
| 1 | Test crozzle score calculation function | | | | |
| Test Method | Method Tested | | | | |
|  | Crozzle.Score() | | | | |
| Test Case ID | Parameters | Expected Data | Actual Data | **Test Result** | **Test Comments** |
| 1 | Use Test 1 data | “Score = 192” |  |  |  |
| 2 | Use Test 2 data | “Score = INVALID CROZZLE” |  |  |  |
| 3 | Use Test 3 data | Can’t validate invalid crozzle |  |  |  |

1. **Test Scenario 6**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Description** | | | | |
| 1 | Check duplicated word uses are within the correct limit | | | | |
| Test Method | Method Tested | | | | |
|  | CrozzleSequences.CheckDuplicateWords(int lowerLimit, int upperLimit) | | | | |
| Test Case ID | Parameters | Expected Data | Actual Data | **Test Result** | **Test Comments** |
| 1 | Use Test 1 data | No output, no DuplicateWordCountError added | No output, no DuplicateWordCountError added | passed |  |
| 2 | Use Test 2 data | No output, no DuplicateWordCountError added | No output, no DuplicateWordCountError added | passed |  |
| 3 | Use Test 3 data | No output, no DuplicateWordCountError added | No output, no DuplicateWordCountError added | passed |  |

1. **Test Scenario 7**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Description** | | | | |
| 1 | Test crozzle validation function | | | | |
| Test Method | Method Tested | | | | |
|  | Crozzle.Validate() | | | | |
| Test Case ID | Parameters | Expected Data | Actual Data | **Test Result** | **Test Comments** |
| 1 | Use Test 1 data | “Score = 192” |  |  |  |
| 2 | Use Test 2 data | “Score = INVALID CROZZLE” |  |  |  |
| 3 | Use Test 3 data | Can’t validate invalid crozzle |  |  |  |

1. **Test Scenario 8**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Description** | | | | |
| 1 | Check Crozzle HTML output | | | | |
| Test Method | Method Tested | | | | |
|  | Crozzle.ToStringHTML() | | | | |
| Test Case ID | Parameters | Expected Data | Actual Data | **Test Result** | **Test Comments** |
| 1 | Use Test 1 data | Return HTML formatted crozzle 1. Actual data too long (4381 characters) to display. | Same as expected data | passed |  |
| 2 | Use Test 2 data | Return HTML formatted crozzle 1. Actual data too long (4396 characters) to display. | Same as expected data | passed |  |
| 3 | Use Test 3 data | Return HTML formatted crozzle 1. Actual data too long (4245 characters) to display. | Same as expected data | passed |  |

1. **Test Scenario 9**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Description** | | | | |
| 1 | Check crozzle group count function | | | | |
| Test Method | Method Tested | | | | |
|  | CrozzleMap.GroupCount() | | | | |
| Test Case ID | Parameters | Expected Data | Actual Data | **Test Result** | **Test Comments** |
| 1 | Use Test 1 data | Group count = 1 | Group count = 1 | passed |  |
| 2 | Use Test 2 data | Group count = 4 | Group count = 4 | passed |  |
| 3 | Use Test 3 data (errors ignored) | Group count = 6 | Group count = 6 | passed |  |

1. **Test Scenario 10**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Description** | | | | |
| 1 | Check configuration parsing function | | | | |
| Test Method | Method Tested | | | | |
|  | Configuration.TryParse() | | | | |
| Test Case ID | Parameters | Expected Data | Actual Data | **Test Result** | **Test Comments** |
| 1 | Use Test 1 data | Return value: true |  |  |  |
| 2 | Use Test 2 data | Return value: true |  |  |  |
| 3 | Use Test 3 data | Return value: false |  |  |  |

1. **Test Scenario 11**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Description** | | | | |
| 1 | Check Word List parsing function | | | | |
| Test Method | Method Tested | | | | |
|  | WorList.TryParse() | | | | |
| Test Case ID | Parameters | Expected Data | Actual Data | **Test Result** | **Test Comments** |
| 1 | Use Test 1 data | Return value: true |  |  |  |
| 2 | Use Test 2 data | Return value: true |  |  |  |
| 3 | Use Test 3 data | Return value: false |  |  |  |

1. **Test Scenario 12**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Description** | | | | |
| 1 | Check Crozzle File parsing function | | | | |
| Test Method | Method Tested | | | | |
|  | Crozzle.TryParse() | | | | |
| Test Case ID | Parameters | Expected Data | Actual Data | **Test Result** | **Test Comments** |
| 1 | Use Test 1 data | Return value: true |  |  |  |
| 2 | Use Test 2 data | Return value: true |  |  |  |
| 3 | Use Test 3 data | Return value: false |  |  |  |