**Validation, Verification and Test plan**

*HashApp*

**Revisions**

|  |  |  |
| --- | --- | --- |
| **Release No.** | **Date** | **Description** |
| Rev. 0 | 02/05/20 | First release |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**TABLE OF CONTENT**

Page #

1.0 Introduction 1-1

1.1 Purpose 1-1

1.2 Scope 1-1

2.0 Tests 2-1

2.1 Matrix

2.2 Test evaluation proceess 2-1

3.0 Test cases 3-1

3.1 Functional tests 3-1

3.1.х [Test Identifier] 3-1

 *Means of control* 3-1

 *Input* 3-1

 *Output* 3-1

 *Test procedure* 3-1

3.2 Unit 3-2

3.2.х [Test Identifier] 3-2

 *Means of control* 3-2

 *Input* 3-2

 *Output* 3-2

 *Test procedure* 3-2

**Introduction**

**Purpose**

This test plan outlines testing that will be done by QE team for 1.0. HashApp release. The plan describes what will and will not be tested, testing approach, descriptions of the tests that will be run.

**Scope**

* Hash generation – three different endpoints for generating hash
* HexUtil helper methods
* Out of scope: Algorithm internal logic

**Tests**

* Unit Testing
  1. These are written and maintained by Development
* Functional Testing
  1. Test for generating hash of string
  2. Test for generating hash of console input
  3. Test for generating hash of file content
* Automation Testing: so far does not provide

**Matrix**

Refer to [‘Test Report.xls’](Test%20Report.xls)

**Test evaluation process**

The definition of successfully passed test is by Junit platform and the class TestHelper.TestResultLoggerExtension is used for collection the data and provide the client. This stands for all the test except HexUtilsTest.convertToUint(). For his specific is need to compare the actual standard output and compare the two results literary

**Test cases**

**Functional tests**

* FkstHashConsoleTest.generateHash(java.lang.String)
* **MC :** TestHelper.TestResultLoggerExtension.class
* **Inp**: @ValueSource({  
  "password",   
  "changeit",   
  "1234",   
  " ", "[Sv.NB]D3X<;f/W[X&VKta:}hUe\*3)r/'jeV6fx6W)W]qUA{yMx=ns<^p@9&%W8G"  
  })
* **Out**: “INFO: -> result : {{calculated hash}}“
* **TP:** run gradle task ‘test’
* FkstHashConsoleTest.generateHash\_null\_input(  
  java.lang.String)
* **MC :** TestHelper.TestResultLoggerExtension.class
* **Inp**: @NullSource
* **Out**: none
* **TP:** run gradle task ‘test’
* FkstHashConsoleTest.generateHash\_empty\_input(java.lang.String)
* **MC :** TestHelper.TestResultLoggerExtension.class
* **Inp**: @EmptySource
* **Out**: “INFO: -> result : {{calculated hash}}“
* **TP**: run gradle task ‘test’
* FkstHashFileTest.generateHash(java.nio.file.Path)
* **MC :** TestHelper.TestResultLoggerExtension.class
* **Inp**: @ValueSource({  
   "src\\test\\resources\\test\_1", "src\\test\\resources\\test\_2", "src\\test\\resources\\test\_3", "src\\test\\resources\\test\_4", "src\\test\\resources\\test\_5", "src\\test\\resources\\test\_6"  
  })
* **Out**: “INFO: -> result : {{calculated hash}}“
* **TP:** run gradle task ‘test’
* FkstHashFileTest.generateHash\_null\_path(  
  java.nio.file.Path)
* **MC :** TestHelper.TestResultLoggerExtension.class
* **Inp**: @NullSource
* **Out**: none
* **TP:** run gradle task ‘test’
* FkstHashStringTest.generateHash(java.lang.String)
* **MC :** TestHelper.TestResultLoggerExtension.class
* **Inp**: @ValueSource({  
  "password",   
  "changeit",   
  "1234",   
  " ", "[Sv.NB]D3X<;f/W[X&VKta:}hUe\*3)r/'jeV6fx6W)W]qUA{yMx=ns<^p@9&%W8G"  
  })
* **Out**: “INFO: -> result : {{calculated hash}}“
* **TP:** run gradle task ‘test’
* FkstHashStringTest.generateHash\_assert\_same(  
  java.lang.String)
* **MC :** TestHelper.TestResultLoggerExtension.class
* **Inp**: @ValueSource({"password"})
* **Out**: “INFO: -> result : {{calculated hash}}“
* **TP:** run gradle task ‘test’
* FkstHashStringTest.generateHash\_assert\_different(  
  java.lang.String)
* **MC :** TestHelper.TestResultLoggerExtension.class
* **Inp**: @ValueSource({"password"})
* **Out**: “INFO: -> result : {{calculated hash}}“
* **TP:** run gradle task ‘test’
* FkstHashStringTest.generateHash\_empty\_string(  
  java.lang.String)
* **MC :** TestHelper.TestResultLoggerExtension.class
* **Inp**: @EmptySource
* **Out**: “INFO: -> result : {{calculated hash}}“
* **TP:** run gradle task ‘test’
* FkstHashStringTest.generateHash\_null\_string(  
  java.lang.String)
* **MC :** TestHelper.TestResultLoggerExtension.class
* **Inp**: none
* **Out**: “INFO: -> result : {{calculated hash}}“
* **TP:** run gradle task ‘test’

**Unit tests**

* HexUtilsTest.convertFromLittleEndianTo64\_AndBack()
* **MC :** TestHelper.TestResultLoggerExtension.class
* **Inp**: String.class password
* **Out**: none
* **TP:** run gradle task ‘test’
* HexUtilsTest.convertFromLittleEndianTo64\_AndBack()
* **MC :** manual
* **Inp**: String.class password
* **Out**: System.out stream
* **TP:** run gradle task ‘test’