Test Design Specification for etaMath Application

13.11.2018

Approvals Signature Block

Organization Responsibility	Signature	Date
Customer /customer representative		
Project Manager		
Software Quality Assurance Engineer	Jay Kishore Duvvuri	13.11.2018
Software Configuration Management Leader		
User Documentation Leader		
User Training Leader		
QA Manager		

Table of Contents

A test design specification generally has the following structure, in the following order:

a) *Test design specification identifier;*
b) *Features to be tested;*
c) *Approach refinements;*
d) *Test identification;*
e) *Feature pass/fail criteria.*

Additional sections may be included at the end. If some or all of the content of a section is in another document, then a reference to that material may be listed in place of the corresponding content. The referenced material must be attached to the test design specification or available to users of the design specification.

1. TEST DESIGN SPECIFICATION IDENTIFIER

Specify the unique identifier assigned to this test design specification. Supply a reference to the associated test plan, if it exists.

Test Plan with the name 'etaMath Test Plan' is provided to describe the objective, scope and approach to testing.
Requirements Traceability matrix document with name 'Requirements Traceability Matrix' is provided.
An Inline document describing the code with name 'etaMath Inline Document' is provided.

2. FEATURES TO BE TESTED

Identify the test items and describe the features and combinations of features that are the object of this design specification. Other features may be exercised, but need not be identified. For each feature or feature combination, a reference to its associated requirements in the item requirement specification or design description should be included.

etaMath application designed in Selenium Webdriver. The runnable **etaMath. Jar** executable file is tested in the command prompt 'CMD'. A total of 20 tasks with Addition, Substraction, Multiplication and Division of numbers is tested for etaMath application on the console.
All the requirements below are to be tested.

Requirement Number	Requirement Description
RQ_01:	Each task shall be one of the following groups:
RQ_02:	Addition with Overflow of calculations resulting in a range of 21 to 99
RQ_03:	Substraction as 'reversion task' of a) --> e.g. 50 - 14 = ?? as reversion task of 14 + 36 = ??
RQ_04:	Multiplication of the numbers
RQ_05:	Division as 'reversion task' ob c) --> e.g. 35 / 5 = ?? as reversion task of 7 * 5 = ??
RQ_06:	Within a single run, no task shall be repeated
RQ_07:	Is the context of RQ_06: A * B == B * A
RQ_08:	In the context of RQ_06: A + B == B + A
RQ_09:	The user shall have 10 sec. of time for answering each task
RQ_10:	The application shall maintain a highscore
RQ_11:	The highscore shall be ordered by correct answers
RQ_12:	The highscore shall be ordered by 'used time' in case correct answers where equal.

3. APPROACH REFINEMENTS

Specify refinements to the approach described in the test plan. Include specific test techniques to be used. The method of analyzing test results should be identified (e.g., comparator programs or visual inspection).

Specify the results of any analysis that provides a rationale for test case selection. For example, one might specify conditions that permit a determination of error tolerance (e.g., those conditions that distinguish valid inputs from invalid inputs).

Summarize the common attributes of any test cases. This may include input constraints that must be true for every input in the set of associated test cases, any shared environmental needs, any shared special procedural requirements, and any shared case dependencies.

The etaMath application asks the user to select option 1 or 2. If the user selects option 1 a list of 20 tasks with add, substract, Multiply and divide of numbers is performed ad asks user to input or enter the answer on the console and performs the calculations and returns the score and time to the user.
If user select the option 2 it displays a list of high score players previously available.

All the 20 tasks share a common code for the user answer input timing of 10 seconds is validated and checked.
The requirements mentioned in the requirements traceability matrix are to meet successfully for complete test coverage.

4. TEST IDENTIFICATION

List the identifier and a brief description of each test case associated with this design. A particular test case may be identified in more than one test design specification. List the identifier and a brief description of each procedure associated with this test design specification.

The test case is written in Java-Selenium Webdriver and is shared in the GitHub link - https://github.com/JayKishoreDuvvuri/etaMAth.git. The test case is executed in a single run asking the user to enter answer 20 times for 20 tasks and finally displays the score and time for the user and terminates finally.

The Test cases are below from the console the etaMath application when user inputs an option of 1 it will ask the user to input the answer. User inputs 20 times. Whenever the user exceeds the time limit of 10 seconds in answering the question, The application returns a warning message.

Note: `User input is in this font colour below`

```
Welcome to etaMath!
Enter '1' to START the Game
Enter '2' to SHOW the Highscores List
Enter Option '1' OR '2' :1
Input/Enter your answer within 10 seconds:
14 + 7 = ?
21 → User input
Answer is Correct
Input/Enter your answer within 10 seconds:
50 - 14 = ?
36
Answer is Correct
Input/Enter your answer within 10 seconds:
14 + 36 = ?
50
Answer is Correct
Input/Enter your answer within 10 seconds:
8 * 7 = ?
56
Answer is Correct
Input/Enter your answer within 10 seconds:
```

```
55 / 11 = ?
5
Answer is Correct
Input/Enter your answer within 10 seconds:
9 * 4 = ?
36
Answer is Correct
Input/Enter your answer within 10 seconds:
35 / 5 = ?
7
Answer is Correct
Input/Enter your answer within 10 seconds:
7 * 5 = ?
35
Answer is Correct
Input/Enter your answer within 10 seconds:
40 - 10 = ?
30
Answer is Correct
Input/Enter your answer within 10 seconds:
10 + 30 = ?
40
Answer is Correct
Input/Enter your answer within 10 seconds:
85 - 25 = ?
60
Answer is Correct
Input/Enter your answer within 10 seconds:
25 + 60 = ?
85
Answer is Correct
Input/Enter your answer within 10 seconds:
66 / 6 = ?
11
Answer is Correct
Input/Enter your answer within 10 seconds:
11 * 6 = ?
66
Answer is Correct
Input/Enter your answer within 10 seconds:
90 - 16 = ?
74
Answer is Correct
Input/Enter your answer within 10 seconds:
16 + 74 = ?
90
Answer is Correct
Input/Enter your answer within 10 seconds:
36 / 12 = ?
3
Answer is Correct
Input/Enter your answer within 10 seconds:
3 * 12 = ?
...Sorry, You didn't responded fast enough
```

```
36
Answer is Correct
Input/Enter your answer within 10 seconds:
29 - 7 = ?
22
Answer is Correct
Input/Enter your answer within 10 seconds:
7 + 22 = ?
...Sorry, You didn't responded fast enough
29
Answer is Correct
Great! Within 64 seconds,you solved 20 Tasks correctly!
You made it to the Highest score! (Place 20)
What is your Name? :Jay
Thank you for playing the Game Jay.The Game is now terminated.
```

The Test case is written as a single test case to handle the single run performance of 20 tasks and selecting **options 1 and option 2** for user.

User selecting Option 2:

```
Welcome to etaMath!
Enter '1' to START the Game
Enter '2' to SHOW the Highscores List
Enter Option '1' OR '2' :2
******************************   HighScores List
****************************************
Michael 19
Peter 18
Andrew 17
Jay 20
Scott 16
Nikos 19
Paula 17
Smith 18
David 19
Simon 20
```

5. FEATURE PASS/FAIL CRITERIA

Specify the criteria to be used to determine whether the feature or feature combination has passed or failed.

Tests executed against the system use the test requirements given to the QA engineer to determine pass or fail.

If a test exhibits a product failure to meet the objectives of any of the test requirements, it will fail and a defect/issue will be reported in the defect tracking system for review by the triage team.

For example: Refer to the Point 4: ‘Test Identification ‘of the document for the running the tests with user input and Test Results.

DOCUMENT CONTROL

CHANGE HISTORY

Revision	Release Date	Description [list of changed pages and reason for change]

DOCUMENT STORAGE

This document was created using <> The file is stored **in below location in GitHub** https://github.com/JayKishoreDuvvuri/etaMAth.git

DOCUMENT OWNER

<> is responsible for developing and maintaining this document.