

Table of Contents

1	INTRODUCTION	2
2	SCOPE	2
3	QUALITY OBJECTIVES	3
3.1	Primary Objectives	3
3.2	Secondary Objectives	3
4	TEST APPROACH	3
4.1	Test Automation	4
5	ROLES AND RESPONSIBILITIES	4
6	ENTRY AND EXIT CRITERIA	5
6.1	Entry Criteria	5
6.2	Exit Criteria	5
7	SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS	5
7.1	Suspension criteria	5
7.2	Resumption criteria	6
8	TEST STRATEGY	6
8.1	QA role in test process	6
8.2	Bug life cycle:	7
8.3	Testing types	8
8.4	Bug Severity and Priority Definition	9
	Severity List	10
	Priority List	10
9	RESOURCE AND ENVIRONMENT NEEDS	11
9.1	Testing Tools	11
9.2	Configuration Management	11
9.3	Test Environment	11
10	TEST SCHEDULE	12
	APPROVALS	13
	TERMS/ACRONYMS	13

Test Plan

Project “etaMath”

Document Revision History

Date	Version	Description	Author	Reviewer	Approver
13.11	0.1	Test plan was created	Jay Kishore Duvvuri		

1 INTRODUCTION

Customer wants a Java Jar file named as ‘etaMath.jar’, which passed the full cycle of testing. Given the specificity of the application it is very important to have the same quality and the approach.

The Test Plan has been created to facilitate communication within the team members. This document describes approaches and methodologies that will apply to the Blackbox testing which is unit and system testing of the application "etaMath.Jar” executable jar file. It includes the objectives, test responsibilities, entry and exit criteria, scope, schedule major milestones, entry and exit criteria and approach. This document has clearly identified what the test deliverables will be, and what is deemed in and out of scope.

2 SCOPE

The document mainly targets the etaMath application testing and validating totally ‘20’ tasks comprising of Addition, Substraction, Multiplication and Division as per Requirements Specifications provided by Client.

2.1 Functions to be tested.

- x etaMath application as an executable Jar file

2.2 Functions not to be tested.

- x GUI
- x Integration Testing
- x Performance

3 QUALITY OBJECTIVES

3.1 Primary Objectives

A primary objective of testing is to: assure that the system meets the full requirements, including quality requirements (functional and non-functional requirements) and fit metrics for each quality requirement and satisfies the use case scenarios and maintain the quality of the product. At the end of the project development cycle, the user should find that the project has met or exceeded all of their expectations as detailed in the requirements.

Any changes, additions, or deletions to the requirements document, Functional Specification, or Design Specification will be documented and tested at the highest level of quality allowed within the remaining time of the project and within the ability of the test team.

3.2 Secondary Objectives

The secondary objectives of testing will be to: identify and expose all issues and associated risks, communicate all known issues to the project team, and ensure that all issues are addressed in an appropriate manner before release. As an objective, this requires careful and methodical testing of the application to first ensure all areas of the system are scrutinized and, consequently, all issues (bugs) found are dealt with appropriately.

4 TEST APPROACH

The approach, that used, is Analytical therefore, in accordance to requirements-based strategy, where an analysis of the requirements specification forms the basis for planning,

estimating and designing tests. Test cases will be created in Selenium WebDriver in Java Language.

All test types are determined properly.

Team also must use experience-based testing and error guessing utilize testers' skills and intuition, along with their experience with similar applications or technologies.

4.1 Test Automation

The test case with class name 'MathsTasks.java' is created in Selenium WebDriver under the package 'com.etaMath.Tasks' for all the 20 Tasks as per the requirements in Java to run and test the application in the Java console or extracting the Jar file and run the Jar file executable in Command prompt.

5 ROLES AND RESPONSIBILITIES

Role	Staff Member	Responsibilities
QA Manager		<ol style="list-style-type: none">1. Acts as a primary contact for development and QA team.2. Responsible for Project schedule and the overall success of the project.
QA Engineer		<ol style="list-style-type: none">1. Understand requirements2. Writing and executing Test cases3. Preparing RTM4. Reviewing Test cases, RTM5. Defect reporting and tracking6. Retesting and regression testing7. Bug Review meeting8. Preparation of Test Data9. Coordinate with QA Manager for any issues or problems encountered during test preparation/execution/defect handling.

6 ENTRY AND EXIT CRITERIA

6.1 Entry Criteria

- x All test hardware platforms must have been successfully installed, configured, and functioning properly.
- x All the necessary documentation, design, and requirements information should be available that will allow testers to operate the system and judge the correct behavior.
- x All the standard software tools including the testing tools must have been successfully installed and functioning properly.
- x Proper test data is available.
- x The test environment such as lab, hardware, software, and system administration support should be ready.
- x QA resources have completely understood the requirements
- x QA resources have sound knowledge of functionality
- x Reviewed test scenarios, test cases and RTM

6.2 Exit Criteria

- x A certain level of requirements coverage has been achieved.
- x No high priority or severe bugs are left outstanding.
- x All high-risk areas have been fully tested, with only minor residual risks left outstanding.
- x Cost – when the budget has been spent.
- x The schedule has been achieved

7 SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS

7.1 Suspension criteria

- The build contains many serious defects which seriously or limit testing progress.

- Significant change in requirements suggested by client
- Software/Hardware problems
- Assigned resources are not available when needed by test team.

7.2 Resumption criteria

Resumption will only occur when the problem(s) that caused the suspension have been resolved

8 TEST STRATEGY

8.1 QA role in test process

- Understanding Requirements:

- Requirement specifications will be sent by client.
- Understanding of requirements will be done by QA

- Preparing Test Cases:

QA will be preparing test cases based on the exploratory testing. This will cover all scenarios for requirements.

- Preparing Test Matrix:

QA will be preparing test matrix which maps test cases to respective requirement. This will ensure the coverage for requirements.

- Reviewing test cases and matrix:

- Peer review will be conducted for test cases and test matrix by QA Manager
- Any comments or suggestions on test cases and test coverage will be provided by reviewer respective Author of Test Case and Test Matrix
- Suggestions or improvements will be re-worked by author and will be send for approval
- Re-worked improvements will be reviewed and approved by reviewer

- Creating Test Data:

Test data will be created by respective QA based on scenarios and Test cases.

- Executing Test Cases:

- Test cases will be executed by respective QA based on designed scenarios, test cases and Test data.

- Test result (Actual Result, Pass/Fail) will updated in test case document Defect Logging and Reporting:

QA will be logging the defect/bugs in Defect management tool, found during execution of test cases. After this, QA will inform respective developer about the defect/bugs.

- Retesting and Regression Testing:

Retesting for fixed bugs will be done by respective QA once it is resolved by respective developer and bug/defect status will be updated accordingly. In certain cases, regression testing will be done if required.

- Deployment/Delivery:

- Once all bugs/defect reported after complete testing is fixed and no other bugs are found, report will be deployed to the client by QA Manager.

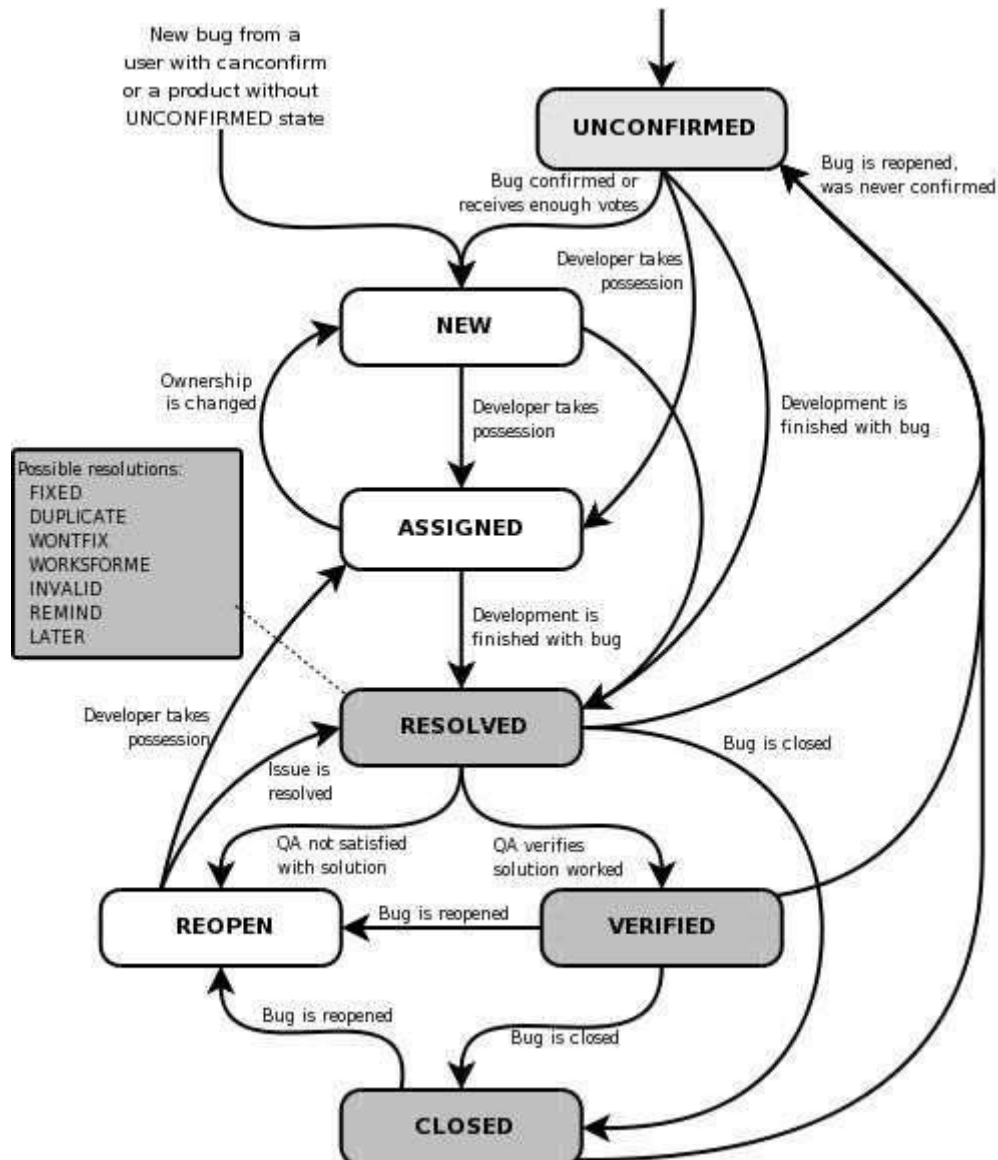
- Once round of testing will be done by QA on client's test site if required Report will be delivered along with sample output by email to respective lead and Report group.

- QA will be submitting the filled hard copy of delivery slip to respective developer.

- Once lead gets the hard copy of delivery slip filled by QA and developer, he will send the report delivery email to client.

8.2 Bug life cycle:

All the issues found while testing will be logged into Defect Management tool.



Black box testing:

It is some time cal

It is some time called behavioural testing or Partition testing. This kind of testing focuses on the functional requirements of the software. It enables one to derive sets of input conditions that will fully exercise all functional requirements for a program.

Integration testing is systematic technique for constructing the program structure while conducting test to uncover errors associated with interacting. In Report, integration testing includes the testing Report from respective location(s).

Functional Testing:

Functional testing is carried out in order to find out unexpected behaviour of the report. The characteristic of functional testing is to provide correctness, reliability, testability and accuracy of the report output/data.

System Testing:

System testing of software is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements.

Performance Testing:

- Check the optimal time the page is loaded
- Check the operation of the system under

load User acceptance testing:

The purpose behind user acceptance testing is to conform that system is developed according to the specified user requirements and is ready for operational use.

Acceptance testing is carried out at two levels - Alpha and Beta Testing. User acceptance testing (UAT) will be done at the Client.

Alpha testing:

The alpha test is conducted at the developer's site by client.

8.4 Bug Severity and Priority Definition

Bug Severity and Priority fields are both very important for categorizing bugs and prioritizing if and when the bugs will be fixed. The bug Severity and Priority levels will be defined as outlined in the following tables below. Testing will assign a severity level to all bugs. The QA Manager will be responsible to see that a correct severity level is assigned to each bug.

The QA Team, Development Lead and Project Manager will participate in bug review meetings to assign the priority of all currently active bugs. This meeting will be known as “Bug Triage Meetings”. The QA Manager is responsible for setting up these meetings on a routine basis to address the current set of new and existing but unresolved bugs.

Severity List

The tester entering a bug into the Defect Management tool is also responsible for entering the bug Severity.

Severity ID	Severity	Severity Description
1	Critical	The module/product crashes or the bug causes non-recoverable conditions. System crashes, GP Faults, or database or file corruption, or potential data loss, program hangs requiring reboot are all examples of a Sev. 1 bug.
2	High	Major system component unusable due to failure or incorrect functionality. Sev. 2 bugs cause serious problems such as a lack of functionality, or insufficient or unclear error messages that can have a major impact to the user, prevents other areas of the app from being tested, etc. Sev. 2 bugs can have a work around, but the work around is inconvenient or difficult.
3	Medium	Incorrect functionality of component or process. There is a simple work around for the bug if it is Sev. 3.
4	Minor	Documentation errors or signed off severity 3 bugs.

Priority List

Priority	Priority Level	Priority Description
1	Must Fix	This bug must be fixed immediately; the product cannot ship with this bug.
2	Should Fix	These are important problems that should be fixed as soon as possible. It would be an embarrassment to the company if this bug shipped.

3	Fix When Have Time	The problem should be fixed within the time available. If the bug does not delay shipping date, then fix it.
4	Low Priority	It is not important (at this time) that these bugs be addressed. Fix these bugs after all other bugs have been fixed. Enhancements/ Good to have features incorporated- just are out of the current scope.

9 RESOURCE AND ENVIRONMENT NEEDS

9.1 Testing Tools

Process	Tool
Test case creation	Selenium WebDriver
Test case tracking	Selenium WebDriver
Test case execution	Manual, Selenium WebDriver, Command Prompt
Test case management	Microsoft Excel
Defect management	Microsoft Word, Excel
Test reporting	Selenium WebDriver, Excel
Check list creating	Microsoft Excel
Project structure	Java Project [JAVA – 8]

9.2 Configuration Management

- x Documents CM: GitHub
- x Code CM: GitHub

9.3 Test Environment

x Support level 1 (browsers):

- x N/A

- x Support level 1 (devices):
 - x N/A
- x Support level 2:
 - x N/A
- x Support level 3:
 - x anything else

10 TEST SCHEDULE

Task Name	Start	Finish	Effort	Comments
Test Planning	12.11	13.11		Support!
Review Requirements documents				
Create test basis	12.11	13.11		
Staff and train new test resources	-	-		
First deploy to QA test environment	13.11			
Functional testing – Iteration 1				
Iteration 2 deploy to QA test environment				
Functional testing – Iteration 2				
System testing	12.11	13.11		
Regression testing				
UAT				
Resolution of final defects and final build testing				
Deploy to Staging environment				
Performance testing				
Release to Production				

APPROVALS:

Role	QA Manager	QA Engineer
Name		Jay Kishore Duvvuri
Signature		

TERMS/ACRONYMS

The below terms are used as examples, please add/remove any terms relevant to the document.

TERM/ACRONYM	DEFINITION
API	Application Program Interface
GUI	Graphical user interface
PM	Project manager
UAT	User acceptance testing
CM	Configuration Management
QA	Quality Assurance
RTM	Requirements Traceability Matrix