Test Plan

For

Broker Insurance WebPage

Version 1.0

**Prepared and Submitted by:**

**Hari Aryal**

**24th Aug 2022**

Document Version History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Description | Author |
|  | 1.9 | Initial Draft |  |

Table of Content

[**Document Version History 2**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.x18p4nsjo87s)

[**Table of Content 3**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.teehmuhsvxti)

[**1. Test Plan Identifier 5**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.pwyrrhild87c)

[**2. References 5**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.lfnon4vumg6n)

[**3. Introduction 5**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.8e65xmy931ib)

[3.1 Background 5](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.9236rqljkbeb)

[3.2 Objectives of the Test Plan 5](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.yhr50c87tshp)

[3.3 Intended Audience 6](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.b1976kcm62o2)

[**4. Test Items 6**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.aebbad4gktae)

[**5. Software Risk Issue 6**](https://docs.google.com/document/d/1W7Fkw38tj0LFJPAibJaDMfUgDbfbnHlriPF9kT94SGI/edit#heading=h.kide38fwtcdk)

[**6. Features to be Tested 7**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.ydixeywdsw98)

[**7. Features Not To Be Tested 8**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.3ubioakdktfg)

[**8. Approach 8**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.ko7foj4b52ed)

[8.1 Testing Levels 8](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.cwon5zt2yzq9)

[8.1.1 Unit Test 8](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.aok1nryygkzm)

[8.1.2 Integration Test 8](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.2vr5qwhdluvw)

[8.1.3 User Acceptance Test 8](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.i6bmg8x3y63l)

[8.2 Test Tools 9](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.cntog66wsqey)

[**9. Item Pass/Fail Criteria 9**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.ulbuy5t21rpz)

[9.1 Installation and Configuration 10](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.pv4ti8qm2885)

[9.2 Document Problem 10](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.46no1m09dk3)

[**10. Suspension criteria and resumption requirements 10**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.bp14vm3zh83v)

[**11. Test Deliverables 11**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.6reangdta3sg)

[**12. Environmental needs 11**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.aspyu0rkmit3)

[12.1 Hardware Needs 11](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.ik8v2o6dz5e2)

[12.2 Software Needs 11](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.13ee9wcnbbfq)

[12.3 Others 11](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.67nw0tohxgd4)

[**13. Staffing and Training Needs 11**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.z61hk7vn1ofy)

[**14. Responsibilities 12**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.d0m2pc82617x)

[14.1 Developers 12](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.li9bdatleiz)

[14.2 QA 12](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.qwtvas9ghj1v)

[14.3 User representative 12](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.cruvt9icfv6)

[**15. Planning Risk and Contingencies 12**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.azgfrtopq9ja)

[**16. Approvals 12**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.52ldb439nn)

[**17. Glossary 13**](https://docs.google.com/document/d/16KDH9S8UGynswRS6JDCTlKI2UFFQ4oUFbQK5HsMy3aA/edit#heading=h.x9eh8mq4dyqi)

1. Test Plan Identifier

GURU99-ID1

2. References

Documents that are referred while making the test plan includes

* Project Plans
* Functional Requirement Specifications

3. Introduction

3.1 Background

This is the level test plan for the implemented Broker Insurance WebPage. This plan will address all those items and elements, (components and subcomponents) that are related to the Aid management process. The primary objective of this test plan is to outline the test strategy necessary to obtain a fully tested system while mainly aiming at removing major defects in each improved version. The project will have three levels of testing, Unit Testing, System/Integration Testing, and Acceptance Testing. The details for each level are addressed in the approach section and will be further defined in the level specific plans.

3.2 Objectives of the Test Plan

The primary objective of this test plan is to communicate to carry out all the possible tests and verify that the requirements are addressed.

Other objectives of these plans are as follows:

* To evaluate the work product; such as requirement, user stories, design and code
* To verify all the specified requirements are fulfilled
* To validate the test is complete and the system meets the users need
* To prevent defects
* To find failures and defects
* To build confidence in the level of the quality of the application
* To comply with legal, contractual and regulatory requirements and standards; and verify the application compliance with such requirements and standards

3.3 Intended Audience

This document is intended for the:

* Project Manager
* QA Team
* Developer Team
* UI/UX Team
* All those who are associated with the project

4. Test Items

NAMS is a critical application, and it includes many subcomponents. The following items will undergo testing.

* Graphical User Interface
* User Home
* Authorized User - Policy Directorate Level

5. Features to be Tested

The list of the features to be tested are:

* Login
* Forgot password
* Add/Edit/View/Delete/Search Entity
* Add/Edit/View/Delete/Search User Data
* Add/Edit/View/Delete/Search Master Data
* Tab Switch
* Responsive GUI
* Search Functionality
* View All Check History
* View Reports
* View Alerts
* Audit Logs
* All Menus functioning
* All buttons functioning
* All links functioning
* Validation and functioning of every form
* Alert messages
* Success message on each successful operation
* Database operations
* Modules of NAMS System
* Business Logic flow
* Approval Mechanism
* User Access Level
* User Access Dashboard and Features

6. Features Not To Be Tested

The list of the features to be tested are:

* Third Party API functioning and correctness
* Digital Signature

7. Approach

The testing of NAMS will consist of Unit Test, Integration Test, and User Acceptance Test. Test cases will be designed accordingly. The static testing shall be designed at the end of each major activity; and as much as required or necessary.

7.1 Testing Levels

The testing levels under consideration of these projects are unit, integration and user acceptance testing.

7.1.1 Unit Test

TBD, decision will be done after consultation with the developer team.

7.1.2 Integration Test

After unit testing, units will be progressively combined and tested. Black box technique will be used to design test cases. The success criteria for integration testing shall be reached when the unit complies and interconnects without any problem.

8.1.3 User Acceptance Test

After integration testing, prior to release, user acceptance testing will be performed. A checklist will be developed and different users from development and the stakeholders will perform the acceptance test. The success criteria for user acceptance test shall be reached when all the business requirement mentioned in the checklist are accepted to be working fine

7.2 Test Tools

**Junit**

It is a Java framework used to perform unit testing. It is used to write repeatable task.

**Mockito**

Mockito is used as a mocking framework, to perform unit testing.

**Gitlab**

It is used for versioning control.

**Trello**

It is used for task management and bug reporting.

8. Item Pass/Fail Criteria

1. **Unit Level**

* All test cases are completed
* A specified percentage of test case is completed with no major defects

1. **Integration Level**

* All test cases are completed
* Integrated units communicate and interact smoothly and properly
* A specified percentage of test case is completed with no major defects

1. **User Acceptance Level**

* All checklists are passed
* No any major issues are detected

The general testing criteria:

* The preconditions are met and all input data is as defined by the input definitions, the test is considered a pass if the result is according to the output definition.
* If a test case is not automated and done properly, but the result isn’t as described in the output definition, the test is considered a fail.
* If the test isn’t done properly, it cannot be said if the test has passed or failed.

8.1 Installation and Configuration

The server side of the application must be properly installed and set up for testing. The system should be stable.

8.2 Document Problem

Documents that can affect testing are the Requirements specification document, because it defines all the necessary requirements which have to be fulfilled, and the Design description document, because it defines software design in detail. However, those documents were reviewed in the previous project iterations, so there is currently no need to do any major changes which would affect testing. Other documents can’t affect testing in any way.

9. Suspension criteria and resumption requirements

During the testing period, people who are performing the testing should be in touch with developers of the applications or server components. Testing can be suppressed and then continued from the same point in the following situations:

* A smaller bug on the client applications that doesn’t affect the server
* A smaller bug in the server functionality that doesn’t affect more than one use-case
  + In this particular case, testers should be very careful because if the bug affects more than one use case it could also affect other tests that were previously run and therefore testing process has to be started from the beginning
* If the server goes down for some reason

Before continuing with the testing the following requirements should be fulfilled:

* The cause of test suspension is no longer valid
  + Bugs are fixed
  + The server is up again and nothing was changed in server-side logic

If the bug is found in test scenarios which involve more than one system component it is very risky to continue with testing from the same point after fixing the bug. Almost every major change of the server-side logic should be the reason for starting the testing from the beginning.

10. Test Deliverables

The following test documentation will be produced:

* Test Plan
* Test Case

11. Environmental needs

11.1 Hardware Needs

* PC with minimum 8 GB RAM
* CPU 2.5 GHz with 4 core at least

11.2 Software Needs

* Java JDK
* Intellij IDE
* Office Packages
* Linux Operating System
* Windows Operating System
* Google Chrome Browser
* Mozilla Firefox Browser
* Internet Explorer Browser
* Safari Browser
* Smart MIS

11.3 Others

* Internet Connection
* Server running
* Test Data

12. Staffing and Training Needs

* To apply systematic testing methods, testing teams require coaching from the testing specialist coach
* Any person involved in the testing shall be given a basic operational training of all sub unit
* Training shall be organized at the end of each release for a better understanding of the project flow.

14. Responsibilities

14.1 Developers

* Unit testing every feature before integrating them into the system
* Testing the whole system
* Fixing bugs

14.2 Quality Analyst

* Testing the whole system
* Finding bugs and passing it to developers
* Verifying the bugs are fixed
* Performing regression testing

14.3 User representative

* Changing and adding requirements on time
* Report if any bugs are found

15. Planning Risk and Contingencies

Some of the potential actions to be taken as the part of resolution of planning risks and contingencies are listed as follows:

* The system will be rejected by the end user, if they find the application difficult to use.
* System Requirement Specification is crucial for testing. Incorrect or insufficient test case will appear in its absence.
* All the requirements should be met before the product can be delivered.

16. Approvals

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Date**  **(mm-dd-yy)** | **Signature** |
| A | CEO |  |  |
| B | Project Manager |  |  |
| C | Business Analyst |  |  |
|  |  |  |  |

17. Glossary

**Unit Test:** Software testing method by which individual units of source code, set of one or more program modules together with associated control data, usage procedures and operating procedures are tested to determine whether they are fit for use.

**Integration Test:**  Checks if group of components (units) interact in the way they are specified by technical system design

**User Acceptance Test:**  User acceptance testing (UAT) is the last phase of the software testing process. During UAT, actual software users test the software to make sure it can handle required tasks in real-world scenarios, according to specifications. UAT is one of the final and critical software project procedures that must occur before newly developed software is rolled out to the market.

**Risk:** A factor that could result in future negative consequences

**Test Plan:**  A document describing the scope, approach and resource of the intended test activities. It also identifies test items, the features to be tested and features not to be tested.

**Defect:** Any part of the software or documentation that is found not to properly express requirements or the design or to cause the software to malfunction.

**IEEE - 829:**  Also known as 829 standard for software and system test documentation. It is an IEEE standard that specifies the form of the set of documents for use in different stage of software testing.