

# Test Plan Document for Insurance Application

## Change Log

Version	Change Date	By	Description
version number	Date of Change	Name of person who made changes	Description of the changes made
1	05-01-2023	Vishal Vaitheeswaran RK	-

1	INTRODUCTION .....	1
1.1	SCOPE .....	1
1.1.1	In Scope .....	1
1.1.2	Out of Scope .....	1
1.2	QUALITY OBJECTIVE .....	1
1.3	ROLES AND RESPONSIBILITIES .....	2
2	TEST METHODOLOGY .....	2
2.1	OVERVIEW .....	2
2.2	TEST LEVELS .....	2
2.3	BUG TRIAGE .....	3
2.4	SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS .....	3
2.5	TEST COMPLETENESS .....	4
3	TEST DELIVERABLES .....	5
4	RESOURCE & ENVIRONMENT NEEDS .....	5
4.1	TESTING TOOLS .....	5
4.2	TEST ENVIRONMENT .....	5

# 1 Introduction

The goal of this test plan is to ensure that the insurance application is functional, reliable, secure, and easy to use for our target audience of insurance brokers. We will focus on the core features of the application, including quote generation, policy management, and claims processing.

## 1.1 Scope

---

### 1.1.1 In Scope

We will perform a variety of test cases to cover different aspects of the application's functionality and usability. These may include:

- Functional tests to verify that the application performs the intended actions correctly
- Usability tests to evaluate the ease of use and user experience of the application
- Automation Tests to quickly cover the regression scenarios

### 1.1.2 Out of Scope

- Performance tests to ensure that the application is responsive and can handle high volumes of traffic
- Security tests to check for vulnerabilities and confirm that sensitive data is properly encrypted

## 1.2 Quality Objective

---

Here make a mention of the overall objective that you plan to achieve without your testing

Some objectives of your testing project could be

- Ensure the Application Under Test conforms to functional and non-functional requirements
- Ensure the AUT meets the quality specifications defined by the client
- Bugs/issues are identified and fixed before go live

## 1.3 Roles and Responsibilities

---

Detail description of the Roles and responsibilities of different team members like

- QA Analyst
- Test Manager
- Configuration Manager
- Developers
- Installation Team

Among others

# 2 Test Methodology

## 2.1 Overview

---

Scrum is a framework for Agile development that emphasizes collaboration, self-organization, and frequent delivery of working software. Testing is an integral part of the Scrum process, with testing activities integrated into each sprint.

## 2.2 Test Levels

---

**Unit tests:** Unit tests are the lowest level of testing and focus on individual units of code, such as functions or methods. These tests are typically written by developers and are used to verify that the code is correct and meets its specifications.

**Integration tests:** Integration tests verify that different units of code work together correctly. These tests may involve testing the integration of different modules, or the integration of the system with external dependencies such as databases or APIs.

**System tests:** System tests validate that the entire system functions as intended and meets the requirements specified in the design documents. These tests may include performance, load, and stress testing.

**Acceptance tests:** Acceptance tests are used to determine whether the system is ready for release. These tests may be performed by the development team, but are typically done by the customer or end user to ensure that the system meets their needs and expectations.

**Regression tests:** Regression tests are used to verify that changes or updates to the system have not introduced new defects or issues. These tests may include a subset of the tests from previous test levels.

## 2.3 Bug Triage

---

Bug triage is the process of prioritizing and assigning defects, or "bugs," that are identified during testing. It involves analyzing the severity and impact of each bug, and deciding on the appropriate course of action. Here are some steps that may be involved in bug triage:

- Identify the bug: The first step in bug triage is to clearly define and document the bug. This may involve gathering information about the steps to reproduce the bug, the expected behavior, and the actual behavior.
- Assess the severity of the bug: Next, the triage team will determine the severity of the bug based on its impact on the system and the users. Bugs may be classified as critical, high, medium, or low severity.
- Determine the priority of the bug: Based on the severity of the bug and other factors such as the difficulty of the fix and the potential impact on the schedule, the triage team will assign a priority to the bug.
- Assign the bug: Once the bug has been prioritized, it will be assigned to a developer or other team member who is responsible for fixing it.
- Follow up: The triage team will follow up on the status of the bug and ensure that it is being addressed in a timely manner. They may also reevaluate the priority of the bug as more information becomes available.

Bug triage is an important part of the testing process, as it helps to ensure that the most critical issues are addressed first and that the team is working efficiently to resolve defects.

## 2.4 Suspension Criteria and Resumption Requirements

---

Suspension criteria refer to the conditions under which testing should be suspended or paused. These criteria may be based on the status of the system being tested, the availability of resources, or other factors.

**System failure:** If the system being tested experiences a failure that cannot be quickly resolved, testing may need to be suspended until the issue is fixed.

**Lack of resources:** If necessary resources such as test data or hardware are unavailable, testing may need to be suspended until they are restored.

**Environmental issues:** If the test environment is unstable or there are other issues that impact the reliability of the tests, testing may need to be suspended until the issues are resolved.

**Changes to the system:** If significant changes are made to the system being tested, testing may need to be suspended until the changes are fully implemented and the system is stable.

Resumption requirements refer to the conditions that must be met before testing can resume after it has been suspended. These requirements may include:

System stability: The system must be stable and functioning correctly before testing can resume.

Resource availability: Any necessary resources such as test data or hardware must be available and ready for use.

Environmental stability: The test environment must be stable and suitable for testing.

Changes to the system: Any changes to the system must be fully implemented and the system must be stable before testing can resume.

Suspension criteria and resumption requirements help to ensure that testing is conducted in a controlled and reliable manner, and that any issues that arise during testing are addressed in a timely and efficient manner.

## 2.5 Test Completeness

---

General guidelines that can be used to determine when a test suite is complete:

The test scope is clearly defined and covers all relevant functionality: The test scope should be based on the requirements and specifications of the system, and should include all features and functionality that are important to the users of the system.

The test design is thorough and covers a wide range of scenarios: The test cases should be carefully designed to exercise the system in a variety of scenarios, including edge cases and error conditions.

The test data is varied and sufficient: The test data should be sufficient to cover a wide range of scenarios and test the system under different conditions.

The test environment is representative of the production environment: The test environment should be configured to simulate real-world usage of the system and should be representative of the production environment.

The test coverage is sufficient: The test coverage should be measured and monitored to ensure that all relevant areas of the system are being tested.

Ultimately, the goal of a test suite is to identify defects and issues that may impact the quality of the system. When the test suite is able to identify a significant number of defects and issues, and when the testing team is confident that the system is of high quality, the test suite can be considered complete.

## 3 Test Deliverable

- Test plan: A document that outlines the goals, scope, target audience, test environment, test cases, and test schedule for the testing effort.
- Test cases: Detailed descriptions of the tests that will be performed, including the steps to be taken, the expected results, and any relevant data or preconditions.
- Test data: The data that will be used for testing, including input data and expected output data.
- Test reports: Documents that summarize the results of the tests, including any defects or issues that were identified and how they were addressed.
- Test artifacts: Any additional materials that are relevant to the testing effort, such as logs, screenshots, or configuration files.

## 4 Resource & Environment Needs

### 4.1 Testing Tools

---

Make a list of Tools like

- Requirements Tracking Tool (JIRA)
- Bug Tracking Tool (JIRA)
- Automation Tools (Cypress Cucumber)

Required to test the project

### 4.2 Test Environment

---

The test environment will consist of a desktop computer with a modern web browser and a stable internet connection. The application will be hosted on a staging server with a copy of the production database.