* **Table of Contents:**

1. **Overview**
2. **Scope**
3. **Test Environments**
4. **Exclusions**
5. **Test Strategy**
6. **Problem Tracking and Test Tracking Procedures**
7. **Roles/Responsibilities**
8. **Test Schedule**
9. **Pricing**
10. **Entry and Exit Criteria**
11. **Tools**
12. **Approvals**

**Overview**

As part of the project, we will use this document to test few functionalities of

**Trade Aggregation Application**.

This document serves as high level test planning document with details on the scope of the project,

test strategy, test schedule and resource requirements, test deliverables and schedule.

**Scope**

The scope of project includes testing the following features of trade Aggregation.

**Swagger API application.**

**Inclusions:**

* **Enter Trade:**

Valid party entering the trade.

Aggregating the trades based on certain common field(as given in problem statement).

* **Search Trade/Trades:**

Based on party and trade reference number.

Based on party and status.

* **Find all cancelled trades from given aggregated trade):**

To find out all the cancelled trades which have satisfied the specified criteria of Aggregation field values to form an aggregated trade.

Exit Trade after maturity date.

From our understanding, we believe above functional areas need to be Tested.

**Test Environments**

Windows 10-Chrome and Edge

Mac OS-Safari Browser

Linux Ubuntu OS-Firefox

Android Mobile OS-Chrome

iPhone Mobile OS-Safari

**Exclusions**

All the features except that are mentioned under 'Inclusions'

Test Automation

**Test Strategy**

we need functional Testing of all the functionalities mentioned in the above Scope section

**As part of Functional Testing, we will follow the below approach for Testing:**

**Step 1**: Creation of Test Scenarios and Test Cases for the different features in scope.

We will apply several Test Designing techniques while creating Test Cases

* Equivalence Class Partition
* Boundary Value Analysis
* Decision Table Testing
* State Transition Testing
* Use Case Testing

We also use our expertise in creating Test Cases by applying the below:

* Error Guessing
* Exploratory Testing

**Step 2:** Our Testing process, when we get an Application for Testing.

* Firstly, we will perform Smoke Testing to check whether the different and
* functionalities of the application are working.
* We reject the build, if the Smoke Testing fails and will wait for the stable build before
* performing in depth testing of the application functionalities.
* Once we receive a stable build, which passes Smoke Testing, we perform in depth testing
* using the Test Cases created.
* Multiple Test Resources will be testing the same Application on Multiple
* Environments simultaneously.
* We then report the bugs in the ‘Jira’ to the reporter.

As part of the Testing, we will perform the below types of Testing:

* Smoke Testing and Sanity Testing
* Regression Testing and Retesting
* Acceptance Testing,

**Problem Tracking and Test Tracking Procedures**

**Defect Reporting Procedure:**

**During the test execution**

Any deviation from expected behaviour by the application will be noted. If it can't be

reported as a defect, it'd be reported as an observation/issue or posed as a question

• Any usability issues will also be reported.

• After discovery of a defect, it will be retested to verify reproducibility of the defect

Screenshots with steps to reproduce are documented.

• Every day, at the end of the test execution, defects encountered will be sent along with

observations.

• Defects will be documented and analysed.

**Roles/Responsibilities**

|  |  |  |
| --- | --- | --- |
| **Name** | **Roles** | **Responsibilities** |
|  | Test Manager | * Escalations |
|  | Test Lead | * Create the Test Plan and get the client signoffs * Interact with the application, create and execute the test cases * Report defects * Coordinate the test execution. Verify validity of the defects being reported. * Submit daily issue updates and summary defect reports to the client. * Attend any meeting with client. |
|  | Senior Test Engineer | * Interact with the application. * Create and Execute the Test cases. * Report defects |
| Harsh Sharma  Anushka Sabharwal | Test Engineer | * Interact with the application. * Execute the Test cases. * Report defects. |

**Test Schedule**

Following is the test schedule planned for the Trade Aggregation project

|  |  |
| --- | --- |
| Task | Time Duration |
| Creating Test Plan | Aug 30 ,2022 to Sept 1 ,2022 |
| Test Case Creation |  |
| Test Case Execution |  |
| Summary Reports Submission |  |

**Pricing**

Will be shared by Test Manager separately.

**Entry and Exit Criteria**

The below are the entry and exit criteria for every phase of Software Testing Life Cycle:

**Requirement Analysis**

**Entry Criteria:**

Once the testing team receives the Requirements Documents or details about the project.

**Exit Criteria:**

List of Requirements are explored and understood by the Testing team.

Doubts are cleared.

**Test Planning**

**Entry Criteria:**

Testable Requirements derived from the given Requirements Documents or Project.

Doubts are cleared.

**Exit Criteria:**

Test Plan document (includes Test Strategy) is signed-off by the Client.

**Test Designing**

**Entry Criteria:**

Test Plan Document is signed-off by the Client

**Exit Criteria:**

Test Scenarios and Test Cases Documents are signed-off by the Client.

**Test Execution**

**Entry Criteria:**

Test Scenarios and Test Cases Documents are signed-off by the Client.

Application is ready for Testing.

**Exit Criteria:**

Test Case Reports, Defect Reports are ready.

**Test Closure**

**Entry Criteria:**

Test Case Reports, Defect Reports are ready.

**Exit Criteria:**

Test Summary Reports.

**TOOLS:**

* Java
* Spring Tool suits
* Maven
* Swagger
* Junit
* REST Assured

**Approval:**

* Shankar Ganesh
* Mohit Kumar Khatri