**Test plan for**

**Liferay QA**

**Revision and Signoff Sheet**

*Document History - To maintain a list of changes being made*

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Change Date** | **Author** | **Description** |
| 00.01 | 2021/5/12 | Jenny Lyn Ngo Gui | First Draft |
|  |  |  |  |
|  |  |  |  |

*Approvers List - To track who has reviewed and signoff on the Test plan*

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Approver/Reviewer** | **Approval/Review Date** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

*Reference Documents - Clearly mark the document used as an input to create the test plan*

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Document Name** |
|  |  |  |
|  |  |  |
|  |  |  |

1 Introduction 3

1.1 Scope 3

1.1.1 In Scope 3

1.1.2 Out of Scope 4

1.2 Quality Objective 4

1.3 Roles and Responsibilities 4

2 Test Methodology 5

2.1 Overview 5

2.2 Test Levels 5

2.3 Test Cycles 5

2.4 Validation and Defect Management 5

2.5 Suspension Criteria and Resumption Requirements 6

2.6 Test Completeness 6

3 Test Deliverables 6

4 Resource & Environment Needs 7

4.1 Testing Tools 7

4.2 Test Environment 7

5 Terms/Acronyms 7

# Introduction

This test plan describes the testing approach and overall framework that will drive the testing of the Liferay Test Team Recruitment – Onboarding page.

Liferay Test Team Recruitment – Onboarding page will be the landing page for new hires joining Liferay’s Test Team, to get to know them better, specifically why they decided to join the team, and make them feel welcome.

The functionality of this spage spans from the “Welcome aboard!” landing page, where new hires enter their information for input of the Test Team and HR.

## Scope

### In Scope

All the feature of Liferay Test Team Recruitment page which were defined in software requirement specs

need to tested.

|  |  |  |
| --- | --- | --- |
| **Test Item** | **Description** | **Responsibility** |
| Forms page/landing page (display) | Forms page showing description field, input fields | iOS – Safari: QA1  Android – Chrome: QA2 |
| Validation | Validation for each field:   * Normal cases – as per specs * Abnormal cases, Edge test cases | iOS – Safari: QA1  Android – Chrome: QA2 |
| Submit | Submit should not be processed if there are validation errors. Submit should only proceed if validation is successful. | iOS – Safari: QA1  Android – Chrome: QA2 |
| Multi-language support – English, Portuguese | Check multi-language support (except translation itself), mainly technical points and consistency | iOS – Safari: QA1  Android – Chrome: QA2 |
| Cross-browser / platform support | Test on: Windows, Mac/iOS, mobile iOS, Android  Browsers: IE, Chrome, Firefox, Safari, Opera | QA2 – automated |
| Usability | Check usability, user-friendliness | QA1 |
| Other page-related errors / display | Check for any other page-related errors, ex. Dummy information / test code | QA1 |

### Out of Scope

These features are not be tested because they are not included in the software requirement specs

1. Linguistic translation (Portuguese) not covered by this testing (translation is expected to be correct)
2. Hardware Interfaces
3. Database logical
4. Website security and performance

## Quality Objective

* Ensure the Application Under Test (AUT) 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
* The final product of the test is twofold:
  + A production-ready software;
  + A set of stable test scripts that can be reused for Functional and UAT test execution.

## Roles and Responsibilities

The following details the Roles and responsibilities of the team members in this project

|  |  |  |
| --- | --- | --- |
| **No.** | **Member** | **Tasks** |
| 1 | Project/Test Manager | Manage the whole project  Define project directions  Acquire appropriate resources |
| 2 | Product Owner | Manage the requirements of the project  Define project requirements and specifications  Check to confirm whether the test specifications cover the requirements |
| 3 | SQA/Test Lead | In-charge of the quality aspect of the project / software to be delivered to client  Define overall test strategy and test approach  Manage QA/testing team resources  Check to confirm whether the testing process is meeting specified requirements |
| 5 | Test Administrator | Builds up and ensures Test Environment and assets are managed and maintained.  Support Tester to use the test environment for test execution |
| 6 | QA/Test Team (Test Designer, Tester) | Identifying and describing appropriate test techniques/tools/automation architecture  Verify and assess the Test Approach  Execute the tests, Log results, Report the defects |
| 7 | QA Engineer/Dev | Implement the test cases, test program, test suite  Write the automated test scripts, |

# Test Methodology/Strategy

## Overview

Given the pseudo-Agile (Dev team runs work in Agile Sprints, while client is not as involved as a Product Owner / stakeholder as expected in Agile) nature of the project, **Iterative** testing approach will be adopted.

Testing will be done alongside the Dev-team Sprints, test report submitted to the client alongside the Iteration release.

The objective of the test is to verify that the functionality of Liferay Test Team Recruitment – Onboarding page works according to the specifications.

The final product of the test is twofold:

• A production-ready software;

• A set of stable test scripts that can be reused for Functional and UAT test execution.

## Test Principles

* Testing will be focused on meeting the business objectives, cost efficiency, and quality.
* There will be common, consistent procedures for all teams supporting testing activities.
* Testing processes will be well defined, yet flexible, with the ability to change as needed.
* Testing activities will build upon previous stages to avoid redundancy or duplication of effort.
* Testing environment and data will emulate a production environment as much as possible.
* Testing will be a repeatable, quantifiable, and measurable activity.
* Testing will be divided into distinct phases, each with clearly defined objectives and goals.
* There will be entrance and exit criteria.

## Test Levels

For this project, the following testing will be conducted:

* Exploratory Testing – make sure critical defects are removed before the next levels of testing can start
* Functional Testing – check the functionality of the application
* API Testing - Test all the APIs create for the application under testing
* Usability Testing – check user friendliness, logical flow that is easy for users to follow
* Compatibility (Cross-browser/device) testing – automated

## Suspension Criteria and Resumption Requirements

If the team members report that there are 40% of test cases failed, suspend testing until the development team fixes all the failed cases.

## Test Completeness

* Specifies the criteria that denote a **successful** completion of a test phase
* **Run** rate is mandatory to be **100%** unless a clear reason is given.
* **Pass** rate is **80%**, achieving the pass rate is **mandatory**

## Project task and estimation and schedule

|  |  |  |
| --- | --- | --- |
| **Task** | **Members** | **Estimate effort** |
| **Design/Create test specification** | Test Designer | 143 man-hour |
| **Perform Test Execution** | Tester, Test Administrator | 77 man-hour |
| **Test Report** | Tester | 10 man-hour |
| **Test Delivery** |  | 20 man-hour |
| **Contingency factor 25%** |  | 55 man-hour |
| **Total** |  | **305 man-hour** |

Details of the estimate in the attached excel file:



Note that these are ballpark estimates and subject to change after gaining clarity about the requirements and the actual iteration release plan for the project.

## Risks and Mitigation

| Risk | Prob. | Impact | Mitigation Plan |
| --- | --- | --- | --- |
| **SCHEDULE**  The project schedule is too tight; it's hard to complete this project on time | High | High | * Set Test Priority for each of the test activity. * The testing team can control the preparation tasks (in advance) and the early communication with involved parties. * Some buffer has been added to the schedule for contingencies. |
| **RESOURCES**  Not enough resources, resources on boarding too late (process takes around 15 days. | Medium | High | Holidays and vacation have been estimated and built into the schedule; deviations from the estimation could derive in delays in the testing. |
| Team member lack the required skills for website testing. | Medium | High | Plan training course to skill up members |
| Test Manager has poor management skill | Medium | High | Plan leadership training for manager |
| A lack of cooperation negatively affects your employees' productivity | Medium | High | Encourage each team member in his task, and inspire them to greater efforts. |
| **DEFECTS**  Defects are found at a late stage of the cycle or at a late cycle; defects discovered late are most likely be due to unclear specifications and are time consuming to resolve. | Medium | High | Defect management plan is in place to ensure prompt communication and fixing of issues. |
| **SCOPE**  Scope completely defined | Medium | Medium | Scope is well defined but the changes are in the functionality are not yet finalized or keep on changing. |
| Natural disasters | Low | Medium | Teams and responsibilities have been spread to two different geographic areas. In a catastrophic event in one of the areas, there will resources in the other areas needed to continue (although at a slower pace) the testing activities. |
| Non-availability of Independent Test environment and accessibility | Medium | High | Due to non availability of the environment, the schedule gets impacted and will lead to delayed start of Test execution. |
| Delayed Testing Due To new Issues | Medium | High | During testing, there is a good chance that some “new” defects may be identified and may become an issue that will take time to resolve.  There are defects that can be raised during testing because of unclear document specification. These defects can yield to an issue that will need time to be resolved.  If these issues become showstoppers, it will greatly impact on the overall project schedule.  If new defects are discovered, the defect management and issue management procedures are in place to immediately provide a resolution. |

# Execution Strategy

## Entry and Exit Criteria

* The entry criteria refer to the desirable conditions in order to start test execution; only the migration of the code and fixes need to be assessed at the end of each cycle.
* The exit criteria are the desirable conditions that need to be met in order proceed with the implementation.
* Entry and exit criteria are flexible benchmarks. If they are not met, the test team will assess the risk, identify mitigation actions and provide a recommendation. All this is input to the project manager for a final “go-no go” decision.
* Entry criteria to start the execution phase of the test: the activities listed in the Test Planning section of the schedule are 100% completed.
* Entry criteria to start each cycle: the activities listed in the Test Execution section of the schedule are 100% completed at each cycle.

|  |  |  |  |
| --- | --- | --- | --- |
| **Exit Criteria** | **Test Team** | **Dev Team** | **Notes** |
| 100% Test Scripts executed |  |  |  |
| 95% pass rate of Test Scripts |  |  |  |
| No open Critical and High severity defects |  |  |  |
| 95% of Medium severity defects have been closed |  |  |  |
| All remaining defects are either cancelled or documented as Change Requests for a future release |  |  |  |
| All expected and actual results are captured and documented with the test script |  |  |  |
| All test metrics collected based on reports from JIRA |  |  |  |
| All defects logged in JIRA |  |  |  |
| Test Closure Memo completed and signed off |  |  |  |
| Test environment cleanup completed and a new back up of the environment |  |  |  |

## Test Cycles

* There will be two cycles for functional testing. Each cycle will execute all the scripts.
* The objective of the first cycle is to identify any blocking, critical defects, and most of the high defects. It is expected to use some work-around in order to get to all the scripts.
* The objective of the second cycle is to identify remaining high and medium defects, remove the work-around from the first cycle, correct gaps in the scripts and obtain performance results.

## Validation and Defect Management

* It is expected that the testers execute all the scripts in each of the cycles described above. However it is recognized that the testers could also do additional testing if they identify a possible gap in the scripts.
* The defects will be tracked through JIRA only. The technical team will gather information on a daily basis from JIRA (also discussed briefly during daily stand-up), and request additional details from the QA/defect reporter. The technical team/Dev will work on fixes.
* It is the responsibility of the tester to open the defects, link them to the corresponding script, assign an initial severity and status, retest and close the defect; it is the responsibility of the SQA/Test Lead to review the severity of the defects and facilitate with the technical team the fix and its implementation, communicate with testers when the test can continue or should be halt, request the tester to retest, and modify status as the defect progresses through the cycle.
* Defects found during the Testing will be categorized according to JIRA. Severity is defined as a custom field for Bugs.

|  |  |
| --- | --- |
| **Severity** | **Impact** |
| Critical (S1) | * A defect that completely hampers or blocks testing of the product/ feature is a critical defect. * This bug is critical enough to crash the system, cause file corruption, or cause potential data loss * It causes the application to hang and requires re-booting the system. |
| Major (S2) | * This bug causes a lack of vital program functionality with workaround. * Any Major feature implemented that is not meeting its requirements/use case(s) and behaves differently than expected may be classified under this. |
| Minor/Medium (S3) | * This Bug will degrade the quality of the System. However there is an intelligent workaround for achieving the desired functionality – for example through another screen. * This bug prevents other areas of the product from being tested. However other areas can be independently tested. * Any Major feature implemented that is not meeting its requirements/use case(s) and behaves differently than expected may be classified under this. |
| Low (S4) | * A defect where there is an insufficient or unclear error message, which has minimum impact on product use. |
| Cosmetic (S5) | * A defect where there is an insufficient or unclear error message that has no impact on product use |

## Test Metrics

Test metrics to measure the progress and level of success of the test will be developed and shared

with the project manager for approval. The below are some of the metrics

|  |  |  |
| --- | --- | --- |
| **Report** | **Description** | **Frequency** |
| Test preparation & Execution Status | To report on % complete, %WIP, % Pass, % Fail  Defects severity wise Status – Open, closed, any other Status | Weekly / Daily (optional) |
| Daily execution  status | To report on Pass, Fail, Total defects, highlight Showstopper/ Critical defects | Daily |
| Project Weekly Status report | Project driven reporting (As requested by PM) | Weekly – If project team needs weekly update apart from daily and there is template available with project team to use. |

# Test Deliverables

Test deliverables are provided as below

**Before testing phase**

* Test plans document
* Test cases documents
* Test Design specifications
* Test scripts

**During the testing**

* Test Data
* Test Traceability Matrix

**After the testing cycles are over**

* Test Results/reports
* Defect Report
* Installation/ Test procedures guidelines
* Release notes

# Resource & Environment Needs

## Testing Tools

* Requirements Tracking Tool – JIRA, Requirements Traceability Matrix plugin. Xray
* Bug Tracking Tool – JIRA
* Automation Tools – Selenium Webdriver, Appium

## Test Environment

Test Environment to be setup as per below.

Windows environment testing:

1. OS: Windows 8 and above
2. Browsers: IE9, 10, and 11, IE Edge, Chrome, Firefox, Opera

Mac environment testing:

1. OS: iOS
2. Browsers: Safari, Chrome, Firefox

iPhone, iPad

1. Browsers: Safari, Chrome, Firefox

Android phone, tablet

1. Browsers: Chrome, Firefox, Opera

# Terms/Acronyms

Make a mention of any terms or acronyms used in the project

| TERM/ACRONYM | DEFINITION |
| --- | --- |
| API | Application Program Interface |
| AUT | Application Under Test |