**DATE**:17/11/2022

**TEST PLAN**

**ICTAK Trainer Management System**

# 

# Prepared by: Group 1

Akhil K

Divya

Karthika A

Priyala Babu

Treesa Antony

# TABLE OF CONTENTS

**1.** INTRODUCTION

1.1 Purpose

1.2 Project Overview

1.3 Audience

**2.** TESTING STRATEGY

2.1 Test Objective

2.2 Test Completeness

2.3 Scopes and Levels of Testing

2.3.1 Performance Testing

2.3.2 Functional Testing

2.3.3 User Acceptance Testing

2.3.4 Automated Regression Testing

2.4 Test Acceptance Criteria

2.5 Test Deliverables

2.6 Milestone list

2.7 Schedules

**3.**  EXECUTION STRATEGY

3.1 Entry and Exit Criteria

3.1.1 Entry Criteria

3.1.2 Exit Criteria

3.2 Validation and Defect Management

3.3 Test Metrics

3.4 Defect tracking & Reporting

**4.** TEST MANAGEMENT PROCESS

4.1 Test Management Tool

4.2 Test Design Process

4.3 Test Execution Process

4.4 Test Risks and Mitigation Factors

**5.** TEST ENVIRONMENT

**6.** ROLES & RESPONSIBILITIES

**7.** APPROVALS

| **No** | **Date** | **Author** | **Description of Change** |
| --- | --- | --- | --- |
| **1** | **15/11/2022** | **Akhil** | **Decided to use the test plan format sent by akhil** |
| **2** | **26/11/2022** | **Akhil** | **Decided that the test case scenario can be included in the test case document itself** |
| **3** | **02/11/2022** | **Akhil** | **Due to the Website issue decided to put the maven project on hold and concentrate on the performance testing.** |
| **4** | **04/12/2022** | **Akhil** | **As per the advice of the mentor,as far as the maven project was done ,we decided to push the GIT HUB and started the performance test of new website(http://64.227.132. 109/** |
| **5** |  |  |  |
| **6** |  |  |  |

# 1. INTRODUCTION

**1.1 Purpose**

This test plan describes the testing approach and overall framework that will drive the testing of the ICTAK Trainer Management System Web application. The document includes:

* Test Strategy: Rules the test will be based on, including the givens of the project, description of the process to set up a valid test (e.g.: creation of test cases, specific tasks to perform, scheduling, data strategy).
* Execution Strategy: Describes how the test will be performed and process to identify and report defects, and to fix and implement fixes.
* Test Management process: To handle the logistics of the test and all the events that come up during execution (e.g.: communications, escalation procedures, risk and mitigation, team roster)

**1.2 Project Overview**

ICTAK Trainer Management System is a trainer management web application which should enable the trainers to enroll in ICTAK Portal and thereby automatically generate the trainer profile.And the Admin can allocate the trainer to different courses and thereby block their calendars.

**Key features of ICTAK Trainer Management System**:

* Signup/Enrollment of trainers.
* Admin/Trainer login/sign in
* Admin can easily search for specific trainers based on search criteria.
* Admin can allocate the trainers with details such as,

a. Start Date

b. End Date

c. Time

d. Course Name

e. Course ID - DSA, FSD, RPA

f. Batch ID - DSA001, DSA002

g. Meeting Link/Venue

h. Schedule - File Upload

i. Send an Email notification to Traine

* The trainer’s calendar gets blocked for the allocated time.

**1.3 Audience**

* Project team members perform tasks specified in this document, and provide input and recommendations on this document.
* Project Manager Plans for the testing activities in the overall project schedule, reviews the document, tracks the performance of the test according to the task herein specified, approves the document and is accountable for the results.
* The stakeholders’ representatives and participants (individuals as identified by the PMO Leads) may take part in the UAT test to ensure the business is aligned with the results of the test.
* Technical Team ensures that the test plan and deliverables are in line with the design, provides the environment for testing and follows the procedures related to the fixes of defects.
* Business analysts will provide their inputs on functional changes.

# 2. TESTING STRATEGY

# 2.1 Test Objective

The objective of the test is to verify that the functionality of “ICTAK Trainer Management System” works according to the specifications. The test will execute and verify the test scripts, identify, fix and retest all high and medium severity defects per the entrance criteria.

* Ensure the Application Under test conforms to functional and nonfunctional requirement
* Ensure the Application Under Test meets the quality specifications defined by the client
* Bugs/Issues are identified and fixed before going live.

**2.2 Test Completeness**

Criterias to check the test completeness:

* 90% Test coverage
* All Manual and Automated Test Cases executed
* All open bugs are fixed or will be fixed in next release

**2.3 Scopes and Levels of Testing**

**2.3.1 Performance Testing**

**PURPOSE:** Performance Testing is done to provide stakeholders with information about their Website regarding SPEED,STABILITY and SCALABILITY

**SCOPE**: Load Test,Stress Test

**METHOD:** This performance testing will perform using JMeter.it is an open source testing tool used for performance testing

**TESTERS:** Testing Team

**TIMING:** At the end of functional testing

**2.3.2 Functional Testing**

**PURPOSE:** Functional Testing is done to provide a quality assurance process and it seeks to establish whether the Web site feature works as per their requirement.

**SCOPE:**

**LANDING PAGE**

| **Sl.No.** | **TEST MODULES** | **WHAT IT DOES** |
| --- | --- | --- |
| **1** | Home | Home page for the trainer management application. It contains more details regarding Internship, Research and Development, Entrepreneurship and Courses offered. |
| **2** | LOGIN | Enables sign in to the trainer management site using username and password |
| **3** | Signup/enroll | To signup/enroll to the trainer management site by providing user detail |
|  |  |  |

**HOME PAGE**

| **Sl. No.** | **TEST MODULES** | **WHAT IT DOES** |
| --- | --- | --- |
| **1** | Home | Shows the list of trainers the admin had approved. Here the admin can assign the trainer with their type of employment i.e Internal, Empanelled, Industry Expert. |
| **2** | Allocate | Admin can search for specific trainers based on search criteria. Also the admin should be able to allocate trainers. |
| **3** | View Allocation | Shows the allocated trainer list along with ScheduleTime, StartDate, EndDate, Venue/Meeting Link. |
| **4** | Logout | To log out from the trainer management website |

**TESTERS:** Testing Team

**METHOD:** The test will be performed according to functional scripts which are stored in an xl sheet that attached in JIRA

**TIMING:** All levels of testing

# 2.3.3 User Acceptance Testing

# Definition:

The purpose of acceptance test is to confirm that the system is ready for operational use. During acceptance test, end-users (customers) of the system compare the system to its initial requirements.

# Participants:

The User Acceptance Testing is performed by the end users.

# Methodology:

Since the business users are the most indicated to provide input around business needs and how the system adapts to them, it may happen that the users do some validation not contained in the scripts. Test team write the UAT test cases based on the inputs from End user and Business Analyst’s.

**2.3.4 Automated Regression Testing**

**Definition:**

Regression testing is the selective retesting of a system or component to verify that modifications have not caused unintended effects and that the system or component still works as specified in the requirements.

# Participants: The whole team

**Methodology:**Regression testing is a black box testing technique. It is used to authenticate that a code change in the software does not impact the existing functionality of the product. Regression testing is making sure that the product works fine with new functionality, bug fixes, or any change in the existing feature.

**2.4 Test Acceptance Criteria**

* Approved functional specification document, Use case documents must be available prior to start of the Test design phase.
* Test cases approved and signed-off prior to start of Test execution.
* Development completed, Unit tested with pass status and results shared to Testing team to avoid duplicate defects.
* Test environment configured and ready to use state.

**2.5 Test Deliverables**

| **Sl No.** | **Deliverable Name** | **Author** | **Reviewer** |
| --- | --- | --- | --- |
| 1. | Test Plan | Test Lead | Project Manager/  Business Analyst’s |
| 2. | Test Cases | Test Team | Business Analyst’s  Sign off |
| 3. | Daily/weekly status report | Test Team/ Test Lead | Test Lead /Project Manager |
| 4. | Test Summary report | Test Lead | Project Manager |

**2.6 Milestone List**The milestones list is tentative and may change due to below reasons

1. Any issues in the System environment readiness
2. Any change in scope/addiction in scope
3. Any other dependency that impacts effects and timeliness

| Milestone Task | Effort (pd) | Start Date | End Date |
| --- | --- | --- | --- |
| Test Planning | 4 | Nov 14 | Nov 18 |
| Test Design | 4 | Nov 20 | Nov 24 |
| Test Development | 2 | Nov 25 | Nov 27 |
| Test Execution | 8 | Nov 28 | Dec 6 |
| Test Evaluation | 3 | Dec 7 | Dec 10 |

(FINAL SUBMISSION : 11 DEC 2022, 8:00PM)

**2.7 Schedules**

| **Task** | **Members** | **Estimate effort** |
| --- | --- | --- |
| **Create the test specification** | Test Designer | 170 man-hour |
| **Perform Test Execution** | Tester, Test Administrator | 80 man-hour |
| **Test Report** | Tester | 10 man-hour |
| **Test Delivery** |  | 20 man-hour |
| **Total** |  | 280 man-hour |

**3.0 EXECUTION STRATEGY**

**3.1. Entry and Exit Criteria**

**3.1.1 Entry 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.

Entry criteria to start the execution phase of the test: The activities listed in the Test Planning section of the schedule are 100% completed.

**Unit Testing**:

* Planning phase has been completed.
* System design, technical design and other relevant documents are properly reviewed, analyzed and approved.
* Business and functional requirements are defined and approved.
* Testable codes or units are available.
* Availability of test environment.

**System Testing**:

* Successful completion of integration testing process.
* Priority bugs found during previous testing activities have been fixed and closed.
* System testing environment is available.
* Test cases are available to execute.

**Acceptance Testing:**

* Successful completion of system testing phase.
* Priority bugs found during previous testing activities have been fixed and closed.
* Functional and Business requirements have been met.
* Acceptance testing environment is ready.
* Test cases are available.

**3.1.2 Exit criteria**

| Exit Criteria | Test  Team | Technical 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 canceled 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 |  |  |  |

**3.2 Validation and Defect Management**

* It is expected that the testers execute all the scripts in each of the cycles described above. However testers could also do regression testing if they identify a possible gap in the scripts.
* The technical team will gather information on a daily basis and request additional details from the Defect Coordinator. The technical team 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 Defect Manager to review the severity of the defects and facilitate with the technical team the fix and its implementation, communicate the 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; it is the responsibility of the technical team to ask for details if necessary, fix the defect, communicate to the team the fix is done, implement the solution per the testers’s request.

Defects found during the Testing will be categorized according to the bug-reporting tool “Jira” and the categories are:

| **Severity** | **Impact** |
| --- | --- |
| 1. (Critical) | * This bug is critical enough to crash the system, cause file corruption, or cause potential data loss. * It causes an abnormal return to the operating system (crash or a system failure message appears). |
| 1. (High) | * It causes a lack of vital program functionality with workaround. |
| 1. (Medium) | * This bug will degrade the quality of the System. * This bug prevents other areas of the product from being tested. |
| 1. (Low) | * There is an insufficient or unclear error message, which has minimum impact on product use. |

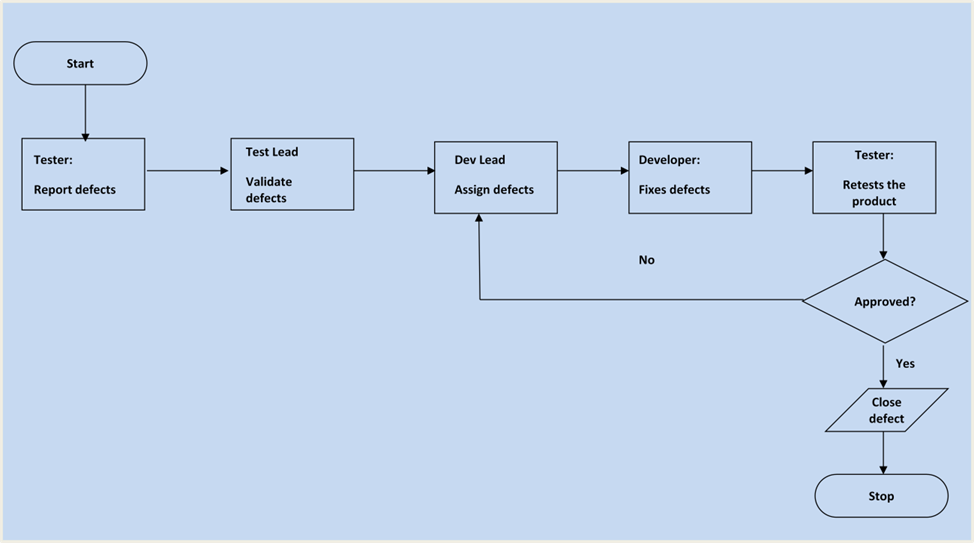
**3.3 Test Metrics**

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

**3.4 Defect tracking & Reporting**

Following flowchart depicts Defect Tracking Process:

****

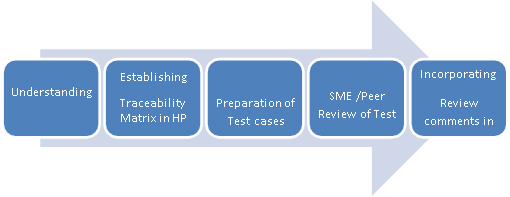
**4.0 TEST MANAGEMENT PROCESS**

**4.1 Test Management Tool**

JIRA is the tool used for Test Management. All testing artifacts such as Test cases, test results are updated in JIRA.

* Each resource in the testing team will be provided with Read/Write access to add/modify Test cases in JIRA.
* During the test design phase, all test cases are written directly into JIRA. Any changes to the test cases will be directly updated in the JIRA.
* Each tester will directly access their respective assigned test cases and update the status of each executed step in JIRA.
* Any defect encountered will be raised as a bug on JIRA, linking to the particular Test case/ test step.
* During defect fix testing, defects are reassigned back to the tester to verify the defect fix. The tester verifies the defect fix and updates the status directly in JIRA.

**4.2 Test Design Process**



|  | |  |  | | --- | --- | |  | * The tester will understand each requirement and prepare a corresponding test case to ensure all requirements are covered. * Each Test case will be mapped to Use cases to Requirements as part of the Traceability matrix. * Each of the Test cases will undergo review by the BUSINESS ANALYST and the review defects are captured and shared to the Test team. The testers will rework on the review defects and finally obtain approval and sign-off. * During the preparation phase, testers will use the prototype, use case and functional specification to write step by step test cases. * Testers will maintain a clarification Tracker sheet and the same will be shared periodically with the Requirements team and accordingly the test case will be updated. The clarifications may sometimes lead to Change Requests or not in scope or detailing implicit requirements. * Sign-off for the test cases would be communicated through mail by Business Analyst’s. * Any subsequent changes to the test case, if any, will be directly updated in Jira. | |
| --- | --- | --- | --- | --- | --- |
|  |  |

**4.3 Test Execution Process**

|  |  |
| --- | --- |
|  |  |

****

* Once all Test cases are approved and the test environment is ready for testing, testers will start an exploratory test of the application to ensure the application is stable for testing.
* Each Tester is assigned Test cases directly in Jira.
* Testers to ensure necessary access to the testing environment, Jira for updating test status and raising defects. If any issues, will be escalated to the Test Lead and in turn to the Project Manager as escalation.
* If any showstopper during exploratory testing will be escalated to the respective development SPOCs for fixes.
* Each tester performs step by step execution and updates the execution status. The tester enters Pass or Fail Status for each of the steps.
* Tester will prepare a Run chart with day-wise execution details.
* If any failures, defects will be raised as per severity guidelines along with screenshots if appropriate.
* Daily Test execution status as well as Defect status will be reported to all stakeholders.
* Testing team will participate in defect triage meetings in order to ensure all test cases are executed with either pass/fail category.
* If there are any defects that are not part of steps but could be outside the test steps, such defects need to be captured and map it against the test case level or at the specific step that issue was encountered after confirming with Test Lead.
* This process is repeated until all test cases are executed fully with Pass/Fail status.
* During the subsequent cycle, any defects fixed applied will be tested and results will be updated in Jira during the cycle.

As per Process, final sign-off or project completion process will be followed.

**4.4 Test Risks and Mitigation Factors**

| **Risk** | **Prob.** | **Impact** | **Contingency Plan** |
| --- | --- | --- | --- |
| **SCHEDULE**  If the start of the testing is delayed due to design tasks. The testing schedule becomes tight. | High | High | The testing team can control the preparation tasks (in advance) and the early communication with involved parties. Increased night shifts to meet delivery date. |
| **RESOURCES**  Not having enough resources at required time. | Medium | High | Planning expected holidays and vacation beforehand and building them into the schedule. |
| **DEFECTS**  Defects discovered late are most likely 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. |
| **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 be resources in the other areas which need to continue (although at a slower pace) the testing activities. |
| **Non-availability of Independent Test environment and accessibility** | Medium | High | Need to be well prepared for a delayed start of Test execution. |
| **Delayed Testing Due To new Issues** | Medium | High | If new defects are discovered, the defect management and issue management procedures are in place to immediately provide a resolution. |

**5. ROLES & RESPONSIBILITIES**

| **Role** | **Responsibility** |
| --- | --- |
| **Software tester** | Execute manual test scenarios for software applications and report defects |
| **Automation tester** | Prepare and execute automated test cases. Provide test execution reports |
| **Performance Tester** | Execute load and stress tests to evaluate application stability and response times.provide performance test reports |
| **Test Analyst** | Monitor and improve the testing process. Contribute to testing, analysis, reporting, and defining quality metrics. |
| **Test Lead** | Plan and coordinate test activities for a team testers |
| **Test manager** | Elaborate test plans and test strategies Manage and coordinate test tea activities. |

**6. TEST ENVIRONMENT**

A testing environment is a setup of software and hardware for the testing teams to execute test cases.

# 

**Operating Systems:** Windows 10, Linux,Mac

**Browsers:** Chrome, Firefox, Safari, Edge

**Mobile Environment:** Android, Iphone

**Automation Tools:** Selenium, Java, TestNG, JMeter

**Bug Tracking Tool:** JIRA

# 

# 

# 7. APPROVALS

Name (In Capital Letters) Signature Date

1.

2.

3.