# Test Plan Document for

# Gurukula

## Description

### 1. Introduction

The detailed design of the application is available for reference at: <<hyperlink>>

Gurukula is an application to manage the information of available branches and staff in any given academic institution. This application provides functionalities like Create, View, Edit and Delete the branches and staff in the given institution.

### 2. Functional Items Under Test

 There will be separate test case document describing test cases in detail and their execution status.

All the above tests marked as ‘Automation-yes’ will be 100% Automated unless they are not Automat-able.

 Each tasks in JIRA will be having corresponding test tasks and detailed test cases for that task are mentioned.

In-Scope:

The following items are in scope of this test execution plan. The term ‘test item’ or ‘item’ refers to one of the following items in scope.

>Login to the web application

>Manage settings for logged in user

>Logout from the application

>Registration of new user

>Manage (Create, View, Edit and Delete) branches

>Manage (Create, View, Edit and Delete) staff

Out-Scope:

>Unit tests of the modules is excluded

>Interactions of Gurukula with any external systems

### 2.1 High level Scenarios to be tested

Below are the functional tests that will be covered as part of end-to-end testing.

| **Item to Test** | **Test Description** | **Comment** |
| --- | --- | --- |
| E2E – Login with correct and valid user name and password. | Verify the login is successful | Automated |
| E2E – Login with invalid user name and correct password of an existing user. | Verify the login is unsuccessful |  |
| E2E – Login with valid user name and incorrect password | Verify the login is unsuccessful |  |
| E2E – Forget password for existing user’s email address | Verify the reset password link is sent to valid user’s forget password request |  |
| E2E – Forget password for non-existing email address | Verify the error message on clicking ‘Reset password’ |  |
| E2E - Registration | Verify the new user registration is successful |  |
| E2E – Fields validation in Registration page | Verify that all the fields have the policies (size, format of text etc.,.) enforced. |  |
| E2E – Registration of an existing user | Verify the user registration is unsuccessful as the user already exists in system. |  |
| E2E – New Branch creation | Verify the creation of new branch is successful |  |
| E2E – Existing Branch creation | Verify the creation of existing branch with same name and code is unsuccessful. |  |
| E2E – Branch search | Verify that all the matching branches should be shown as search results. |  |
| E2E – Branch search with partial text | Verify that all the matching branches should be shown as search results. |  |
| E2E – Fields validation in Branch creation page | Verify that all the fields have the policies (size, format of text etc.,.) enforced. |  |
| E2E – View of an existing branch | Verify the details of the selected branch are shown on clicking ‘view’ |  |
| E2E – Edit of an existing branch | Verify the details of the selected branch are allowed to be edited and saved on clicking ‘edit’ |  |
| E2E – Delete of an existing branch | Verify the selected branch is deleted on clicking ‘delete’ and confirming the action. |  |
| E2E – Pagination while viewing branches | Verify the pagination is available while viewing the available branches |  |
| E2E – New Staff creation | Verify the creation of new staff is successful |  |
| E2E – Existing Staff creation | Verify the creation of existing staff with same name and branch is unsuccessful. |  |
| E2E – Staff search | Verify that all the matching staff should be shown as search results. |  |
| E2E – Staff search with partial text | Verify that all the matching staff should be shown as search results. |  |
| E2E – Fields validation in Staff creation page | Verify that all the fields have the policies (length, format of text etc.,.) enforced. |  |
| E2E – View of an existing Staff | Verify the details of the selected branch are shown on clicking ‘view’ |  |
| E2E – Edit of an existing Staff | Verify the details of the selected branch are allowed to be edited and saved on clicking ‘edit’ |  |
| E2E – Delete of an existing Staff | Verify the selected branch is deleted on clicking ‘delete’ and confirming the action. |  |
| E2E – Pagination while viewing staff | Verify the pagination is available while viewing the available Staff |  |
| E2E – Account Settings | Verify the Account settings changed are saved successfully |  |
| E2E – Account Settings fields validation | Verify that all the fields have the policies (length, format of text etc.,.) enforced. |  |
| E2E – Account Password | Verify that the password updated successfully. |  |
| E2E – Account Password field validation | Verify that the password field has the policies (length) enforced |  |
| E2E – Account Sessions | Verify that the Account session details are shown |  |
| E2E – Account Session invalidation | Verify that the Account session details are cleared once the invalidate is clicked |  |
| E2E - Logout | Verify that the account is logged out of the page on clicking ‘Account -> Logout’ |  |

### 2.1.1 Negative Test scenarios(High Level):

Below are the test cases that will be covered as part of negative testing for this feature.

| **Item to Test** | **Test Description** | **Comment** |
| --- | --- | --- |
| Negative - |  |  |

### 2.1.2 E2E Tests (Existing Regression Test Suite)

As there is no regression suite yet, this section becomes irrelevant

### 2.2 Features NOT to be tested

Features not in scope as part of the design will not be tested.

| **Item to Test** |
| --- |
| Gurukula’s integration with external systems, if any. |
|  |

### 3. Test techniques

All the above identified End-to-End test scenarios will be automated and added to the Regression test suites.

The automation frameworks to be used are but not limited to Java, TestNG, Selenium, Apache POI HSSF, ..

Perf testing for each module will be done separately and the generic scripts are identified. We will use JMeter to achieve the performance and Stress testing.

Test failures' will be monitored in Continuous Integration tool Jenkins..

### 3.1 Testing tasks

* We will be covering 100% automation for all the major module functional tasks and this will be part of scheduled Qualification.
* Performance test for all the modules will be executed.

#### 3.1.1 Functional Test task ( End-to-End)

        End-to-end functional tests needs to be developed from the above test scenarios identified. If there are any extra scenarios that are thought of during test case generation, the test plan needs to be identified accordingly.

Manual execution of all the identified test cases

100% Automation of all medium to high priority tests needs to be completed before the release of the application.

#### 3.1.3 FMEA Test Tasks

       N/A

#### 3.1.3 Performance Test task

       The normal performance test for modules needs to be run and no degradation of performance is accepted. Use Jmeter or equivalent performance test tool to execute performance test on the application.

#### 3.1.4 Quality Attributes Testing efforts’ Checklist

##### Storage, Sizing

 N/A

##### Performance

##### See Performance Test task section above

##### High Availability

##### See Performance Test task section above

##### Scalability

##### See Performance Test task section above

##### Multi-tenancy

     N/A

##### Configuration

     N/A

##### Enablement

    N/A.

##### Security

     We have to run security tools for WAPT(Web Application Penetration testing) for Gurukula. Applications like ZAP, Paros proxy etc., will help analysing the traffic and uncover any security issues. Also, all the communication between Client and Gurukula application must be secured using HTTPS. No HTTP should communications are allowed.

##### Reliability

     N/A

##### Management / Monitoring / Alerting

     N/A

##### Self-Service / Tools

       N/A

##### Backup / DR

       N/A

##### Support-ability (Logging etc)

        N/A

### 3.2 Suspension criteria

* Broken Builds
* Environments issues

### 3.3 Features pass or fail criteria

* 100% coverage for all unit tests, integration and End-to-End tests
* Code coverage is at the current level or higher
* No degradation in performance trend

### 3.4 Entry criteria, Exit criteria

**Entry Criteria :**

* A proper dedicated test environment is in place to support the entire system test process, QA environment must be available.
* All documentation and design of the architecture must be made available.
* Test plan should be reviewed and test approach should be documented.
* All test scenarios should be documented in JIRA.
* All source code should be unit tested

**Exit Criteria :**

All defects (up to P3, inclusively) found during Testing/Re-testing efforts should be fixed.

* 100% automation of functional (E2E, Integration) tests, unless they are not automate-able.
* Application performance should not be degraded.
* Planned deliverables are ready.

### 4. Test Environment

* Windows or Mac OS.
* Java 1.8u1.
* Chrome Browser 50.0 or above

### 5. Test deliverables

* Test design document in wiki
* Test scenarios and cases in JIRA
* Automated test cases in scheduled Jenkins CI.
* Bug reports if any, in Jira.

### 6. Schedule

| **Item to Test** | **Creation Effort** | **Creation Start Date** | **Creation End Date** | **Execution Start Date** | **Execution End Date** |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

### 7. Review

This document needs to be reviewed by the following stakeholders.

|  |  |
| --- | --- |
| **Stakeholder name** | **Review/Date** |
| Pavithra Jayaraman |  |
| Product Owner |  |
| Project Manager |  |
| Team Member#1 |  |
| Team Member#2 |  |
| Team Member#3 |  |
| Team Member#4 |  |