[**References**](#_pugfgfeq40ae) **2**

[**Introduction**](#_dsdlmnv0bhd) **2**

[Purpose](#_qacb3ig60il2) 2

[Project Overview](#_lyi3i21uco2n) 2

[**Test Items**](#_3rgrbp8b5hqr) **2**

[**In scope**](#_6104z6o94hq1) **3**

[Smoke testing](#_60i0frtgu8ba) 3

[UI/UX Testing](#_nkhy5ulzulaq) 3

[Functional Testing](#_h3cee15zf46l) 3

[Sanity Testing](#_eji0zkdiqf3o) 3

[API Testing](#_og1b1kren053) 3

[Features to be tested](#_qau1mcywfyf4) 3

[**Out of scope**](#_3o1kisde2yva) **3**

[**Test Strategy**](#_jyiskgr2omau) **4**

[Test Data Preparation](#_hkh6e7h151s6) 4

[Testing Type](#_uws04zmu308s) 4

[Tools](#_wp4cp6okldq1) 4

[**Testing Environment**](#_1garrhvc9ade) **5**

[**Assumption and Risks**](#_abj2fy9b79zh) **5**

[Assumption](#_2fm0tlz3uo3d) 5

[Risks](#_l0jyu062sopa) 5

[**Test Deliverables**](#_3e5zjwfwq7te) **6**

[**Bug Reporting**](#_su8lcs9aei7v) **6**

Administration Module Test Plan v1.0

# References

* Mockups (Invision app)
* Requirement document.

# Introduction

## Purpose

The purpose of this document is to describe the plan for validating all the requirements so that School Management System application can be used by the administrator to fulfill the needs. The document will also describe the activities need to perform during testing of features mentioned in user stories. The document talks about scope (in and out), limitations and risk along with mitigation plans. The document also describe additional tests and strategies could be performed to enhance the quality.

## Project Overview

The administration module of the School Management System can be utilized by the school administrator to view relevant information related to students of ABC school, such as Name, Id, ACT and SAT scores.

The functionality of the module expands from viewing the student details, search information based on name and score, sort student list through scores and export detail of individual student in CSV format.

# Test Items

Test items are mentioned on the basis of test activities that align with the features.

The modules that are to be tested include:

1. Login module
2. Admin Dashboard
3. Details export module

# In scope

The following are the testing to be performed for the following sprint

## Smoke testing

* Perform to test the critical functionalities are working.
* Further rigorous testing can be avoided if smoke testing fails

## UI/UX Testing

* Test performed to validate whether UI objects are located as per the mock ups provided
* Responsiveness of web application is also verified

## Functional Testing

* Functionalities are verified and validated against the user requirements

## Sanity Testing

* Before release of application major functionalities are check to detect any issues or bugs

## API Testing

* Testing to verify communication and data exchange

## Features to be tested

1. Administration login (High Priority)
   1. Email field validation
   2. Password field validation
   3. Login button accessibility
2. Students details visibility
3. Search student according to names and scores
4. Sort according to ACT and SAT scores
5. Export student details as CSV files

# Out of scope

* Performance testing
* Security testing
* Portability testing

# Test Strategy

Agile methodology is to be followed throughout the project.

## Test Data Preparation

Data required for username and password are prepared based on Equivalence Class Partitioning where data will be partitioned as invalid and valid for both positive and negative test cases.

Similarly, Decision Table will be implied to test various scenarios which could be testing login with all empty fields, either one of the fields is kept empty.

## Testing Type

Smoke testing is performed for the Login feature with test cases derived from Decision table.For examples following combinations are used

* Valid username and password
* Invalid username and valid password
* Valid username and invalid password
* Invalid username and valid password
* Empty fields

UI/UX testing is performed as black box testing to ensure that the list appears on the dashboard and if the interface is responsive by using in with different browser sizes.

Database testing is performed for search module with CRUD operation List should only contain information in the database.

In API testing the status of request and data are verified through the use of automation tool, POSTMAN.

## Tools

|  |  |
| --- | --- |
| Browsers | Chrome (High priority), Firefox (Medium priority) |
| Version | 79.0.3945.88 (Official Build) (64-bit) |
| API automation tool | POSTMAN |

# Testing Environment

|  |  |
| --- | --- |
| Development | Development team |
| QA | QA team, TL, PM |
| Client | School Admin |

# Assumption and Risks

## Assumption

* Requirements do not abruptly change
* The user stories are valid.
* The prepared test cases and data will be sufficient for testing

## Risks

Following are considered to the risk with the mitigation plan

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Impact** | **Probability** | **Mitigation Plan** |
| Test data prepared for the fields might not be sufficient | High | High | Review the data with TL and inform the client |
| User might have late changes in requirements | Medium | High | Clarify the user stories at the beginning and avoid any vagueness |
| Test environment may not be available as per client requirement | Medium | Medium | Inform the client and negotiate with the environment |

# Test Deliverables

* Test Plan
* Test Cases
* Bug report
* Test Data
* Test Metrics
* Summary report

# Bug Reporting

Bug detected in QA environment will be reported to the Development Team with priority and severity mentioned. Bug tracking tool as JIRA will be used to keep the QA, Development team and Project Managers align