TEST PLAN- STUDENT RECORD SYSTEM

[**Test Plan Identifier:**](#_i8cnzmvszut0) **2**

[**References:**](#_hmma3pdssdcy) **2**

[**Introduction:**](#_8fgkpi6pupjo) **2**

[Purpose:](#_sydatgu0uxbw) 2

[Problem Statement:](#_dtfhr7wfxfqw) 2

[**InScope:**](#_c0ub25udecy3) **3**

[**Features to be Tested:**](#_jpyulugss85g) **4**

[**Out of Scope:**](#_kz0id6myqjx3) **4**

[**Testing Strategy/Approach**](#_psrs5awy3f2i) **5**

[Test Data Preparation](#_7lsswpu1qlny) 5

[Decision Table](#_3hmwg9ogs7r4) 5

[Functionality Testing](#_vvckq3wwu98e) 5

[**Test Deliverables**](#_yhyr3dcfp7a0) **5**

[**Testing Environment**](#_i4ngbam5lvdz) **6**

[**Assumptions/Risks**](#_i73q0241x5zq) **6**

[Assumptions:](#_he7ejaaugx72) 6

[Risks](#_5i25j59pfy4l) 7

[**Bug Report**](#_cmqb88o4kpnc) **8**

### 

### 

### 

# 

# 

# Test Plan Identifier:

Test Plan for School Administrator v2.0

# References:

* Requirement Analysis
* Mockup (Invision App)

# Introduction:

## **Purpose:**

Purpose of this test plan is to make a login page which gives access to the school administrator. Secondly, an access to fields page which has student credentials and score, and a search option for easy access.Finally, is the student detail view, along with a profile photo and a convert to CSV option.

## **Problem Statement:**

Schools in the present day and age have the problem of a huge number of students and a mismanagement of student directories which results in the delay in access of the scores of the students. The school administrator has presented the view where the school requires a planning, monitoring, organizing and controlling of the student database with respect to the scores.This existing plan will mitigate the problem of slow access of student database. This plan also strives to make the access and overviewing functioning of the school administrator easier and more flexible by also giving a functionality of converting the database into CSV for easier access.

# InScope:

Describes all the tests that needs to be done in this test phase.

* UI Testing:
  + Movement of arrow keys and its functionality is monitored.
* Smoke Testing:
  + Critical functioning of the system is seen i.e. for login and the application is interactive.
* Database and Integrity Testing:
  + Is done in order to ensure database access methods and the processes function properly and without data corruption.
* Functionality Testing:
  + for overall functionality of the website
* Usability Testing:
  + Is done to ensure that the content on the pages will have accuracy, logical/structural display

## **Features to be Tested:**

Features of the application are:

* Login Page
  + Email Validation
  + Password Validation
  + Login button
* Fields Page
  + Student credentials
  + Sorting of ACT and SAT scores
  + Search function for name and ID
  + Student administrator login
  + Profile Photo verification of student administrator.
* Student detail view
  + Profile photo of student
  + Student credentials
  + Export as CSV function

# Out of Scope:

All the tests that do not need to be done at this phase.

* Security Testing
* Potability Testing
* Load Testing
* Performance testing

# Testing Strategy/Approach

## **Test Data Preparation**

The test data that is created is a prototype of the valid sores of students in a school. The data has names, ID, SAT and ACT scores.

### **Decision Table**

Email and password field can be checked if required conditions are fulfilled using decision tree module. These have invalid, empty or incorrect email address will be evaluated.

* Invalid email valid password.
* Invalid password and valid email.
* Invalid email and invalid password.
* Valid email and valid password.

### **Functionality Testing**

Functionality testing can be done for test cases.The Test Cases will be prepared and testing will be done for various functionalities. For. example. Selenium can be used for this browser based web application.

# Test Deliverables

* Test plan document.
* Test cases.
* Summary Reports
* Sanity Checklist
* Test matrix
* Test result

# 

# Testing Environment

|  |  |
| --- | --- |
| Browser | Chrome, Safari, Firefox, Opera, Internet Explorer |
| Operating System | Windows, Linux, IOS |
| Version | Windows 79.0.3945.88  MacOs Catalina |

# Assumptions/Risks

## **Assumptions:**

* 1. Prepared sets of test data will be sufficient for the database.
  2. Test data will be prepared in such a way that most of the criteria will be fulfilled i.e. no field is left.
  3. Data for the criteria will not be corrupted.
  4. User requirements will not have massive changes.
  5. User story will be true and permanent.

## 

## 

## **Risks**

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Impact | Probability | Mitigation Plan |
| Test data prepared might not be sufficient and may not cover all the criteria | High | High | Need to verify if all test data is sufficient with the client and stakeholder. |
| Incorrect estimation of resources when testing is to begin | High | Medium | Proper estimation regarding project time and buffer time needs to be done before project begins. |
| Changes in original requirements or designs. | Medium | Medium | Requirements and designs should be referred from requirements document and any changes needs to be evaluated before integrating into system. |
|  |  |  |  |

# Bug Report

Bug tickets will be created for the issues reported during the test execution and will be kept in the sprint according to the priority of the task.

The issues reported in the tasks from UAT/Production will be marked as the highest priority and will be verified by the QA.