|  |  |
| --- | --- |
|  | |
| CAESAR-logo.png | |
| Test Plan | |
| Version: 1.0  12/11/2009 | |
|  | |
| **Authors:** | **Reviewers:** |
| Paul Montagna | Matthew Close |
|  | Mark Isherwoord |
|  | George Ng |
|  | Jamie Stevens |
|  | Paul Hames |

|  |
| --- |
|  |

# 

# Executive Summary

This document outlines the testing procedures and functions to be completed in regards to the project being produced by Coordinated Assistance Enterprise for Students at Risk (CAESAR) Team. They will develop a software solution and a networked database for Students at Risk with the ICT section of the engineering department of multiple Swinburne campuses. The test plan document will define the procedures which will be used throughout the development process of the project.

Table of Contents

[Executive Summary 2](#_Toc240772285)

[1 Introduction 4](#_Toc240772286)

[1.1 Document Scope 4](#_Toc240772287)

[1.2 Intended Audience 4](#_Toc240772288)

[1.3 Boundary of Testing Plans 4](#_Toc240772289)

[1.4 System Overview 4](#_Toc240772290)

[2 General Testing Procedures 5](#_Toc240772291)

[2.1 Testing Process 5](#_Toc240772292)

[2.1.1 Approach 5](#_Toc240772293)

[2.1.2 Tools and Techniques 5](#_Toc240772294)

[2.1.3 Risks and Contingencies 5](#_Toc240772295)

[3 Testing Deliverables 6](#_Toc240772296)

[3.1 Testing Environment 6](#_Toc240772297)

[3.1.1 Software Requirements 6](#_Toc240772298)

[3.1.2 Hardware Requirements 6](#_Toc240772299)

[4 Functional Test Plan 7](#_Toc240772300)

[4.1 Maintain Students 7](#_Toc240772301)

[4.2 Maintain Users 8](#_Toc240772302)

[4.3 Maintain CSV File 8](#_Toc240772303)

[4.4 Maintain Meetings 9](#_Toc240772304)

[4.5 Generate Reports 10](#_Toc240772305)

[4.6 Prompt User with Updates 10](#_Toc240772306)

[4.7 Application Tools 11](#_Toc240772307)

[4.8 User Tools 11](#_Toc240772308)

[5 Non-Functional Test Plan 12](#_Toc240772309)

[6 Client Acceptance Test Plan 13](#_Toc240772310)

[7 Requirements Traceability 14](#_Toc240772311)

[8 References 15](#_Toc240772312)

# 

# 1 Introduction

## 1.1 Document Scope

The purpose of this document is to provide a plan of testing procedures that will be used throughout the development of the Students at Risk database application. The testing procedures will ensure that the application will function as intended for the ICT department.

## 1.2 Intended Audience

The intended audience of this document is the Coordinated Assistance Enterprise for Students at Risk (CAESAR) Team.

## 1.3 Boundary of Testing Plans

The Test Plan will include the following:

* Integration Test Plan
* System Test Plan
* Client Acceptance Test Plan
* Unit Test Plan

## 1.4 System Overview

The current system seeks to identify students at risk and contact them to offer assistance and/or address any problems they may be having. There is evidence to suggest that the earlier a student is identified and contact is made the more likely a student will continue their studies.

The system that the CAESAR Team will be implementing will help to maintain a list of students who are missing classes repeatedly or a teacher believes is having trouble in class. The system will be able to add students automatically via the ‘Allocate+’ attendance system or they can be added manually by teachers.

Once a student has been entered into the system a message will be sent to the administrator, who will then arrange a meeting time and enter it into the system. A confirmation email will be sent to both parties confirming the meeting details. After the meeting is completed the teacher/team leader will record the outcome of the meeting. The outcome possibilities include; if the student withdraws, another meeting is required or if the student is no longer flagged as at risk.

# 

# 2 General Testing Procedures

## 2.1 Testing Process

### 2.1.1 Approach

The testing process will be conducted by members of the CAESAR Team during the course of the project and by the client during the Client Acceptance Test.

The Client Acceptance Test will be carried out after the completion of application coding but prior to application implementation. This testing will be carried out to ensure application usability and working functionality as per the application requirements.

CAESAR Team will be responsible for testing the application code and functions throughout the development lifecycle by conducting white box testing. This will require close analysis of the developed code to ensure that it is error free and functions as desired.

### 2.1.2 Tools and Techniques

Coding standards will be used by CAESAR Team throughout the development of the application and will be reviewed regularly to ensure that the standards are being adhered to. We will also be performing tests which include:

* Unit Testing
* System Testing
* Usability Testing
* Security Testing
* Standards Compliance

### 2.1.3 Risks and Contingencies

For more risks and how they are managed please refer to our Risk Management Plan.

|  |  |
| --- | --- |
| **Risks** | **Contingency** |
| Data loss | Extra work load for team members. |
| Project running overtime | If project is at risk of non completion then the team will re-examine the project scope. Team Members will need to work overtime to complete the required tasks. |
| Client requests changes to functional requirements | The request is then put forward at a team meeting and is evaluated and discussed before reporting back to the client. |

# 

# 3 Testing Deliverables

The following are testing deliverables that will be produced throughout the course of the project:

* Testing Plan
* Testing Templates
* Client Acceptance Test
* Testing Report

## 3.1 Testing Environment

Tests will be conducted onsite at Swinburne’s Hawthorn campus. Tests will also be conducted offsite at team member’s homes to ensure the application functions correctly.

### 3.1.1 Software Requirements

The following lists the minimum software requirements the proposed application will be tested on:

* Windows XP
* Windows Vista
* Windows 7

### 3.1.2 Hardware Requirements

The following lists the minimum hardware requirements the proposed application will be tested on:

* Intel Celeron D, 2.66 GHz
* 512 MB Ram
* Novell Network System

.

# 

# 4 Functional Test Plan

## 4.1 Maintain Students

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Functionality** | **Use Case ID** | **Description** | **Expected Result** |
| **FT0.1.a** | Add Student | UC0101 | Student details are entered. | Student details are validated and processed. |
| **FT0.1.b** |  |  | Student details are saved. | Student is added to the list of students at risk and a confirmation message is displayed. |
| **FT0.1.c** |  |  | Student details are unable to be saved. | Appropriate error message is displayed and the detail entering screen is cleared. |
| **FT0.2.a** | Search Student | UC0102 | Student information is entered. | Data is validated and processed and appropriate results are returned. |
| **FT0.2.b** |  |  | Details cannot be displayed. | Appropriate error message is displayed and a new search form is displayed. |
| **FT0.3** | View Student | UC0103 | Student is selected from search result. | The selected student’s information is displayed in the view form. |
| **FT0.4.a** | Edit Student | UC0104 | Select edit. | Edit form is shown with students details ready to be changed. |
| **FT0.4.b** |  |  | Changes are made. | Student details are edited by the user via the edit student form. |
| **FT0.4.c** |  |  | Save button is selected. | Student details are updated in the database and the new information is displayed. |
| **FT0.4.d** |  |  | Data entered is incorrect. | Error message is displayed and student’s information on the form is restored to normal. |
| **FT0.5** | Archive Student | UC0105 | Archive option is chosen. | Student’s information is archived. |

## 

## 4.2 Maintain Users

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Functionality** | **Use Case ID** | **Description** | **Expected Result** |
| **FT1.1** | Edit Admin Permissions | UC0201 | Add a user to the administrator user group. | User is added to the admin user group, only a current admin can perform this operation. |
| **FT1.2.a** | Add User Availability | UC0202 | Times of availability are added. | Data is validated and saved to database then user is returned to main screen. |
| **FT1.2.b** |  |  | Times entered are not valid. | Error message is displayed and the Add User Availability form is cleared. |
| **FT1.3.a** | Edit User Availability | UC0203 | New updated times are entered. | New times are saved to the database and status is updated. |
| **FT1.3.b** |  |  | New updated times are not valid. | Error message is displayed and the user is returned to the main form. |

## 4.3 Maintain CSV File

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Functionality** | **Use Case ID** | **Description** | **Expected Result** |
| **FT2.1.a** | Import CSV File | UC0301 | Select the CSV file to upload. | File is validated to make sure it is in the appropriate format and then confirm the upload before it is imported and the students added the MySQL database. |
| **FT2.1.b** |  |  | CSV file not formatted correctly. | CSV import fails and the user is returned to the main screen. |

## 

## 4.4 Maintain Meetings

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Functionality** | **Use Case ID** | **Description** | **Expected Result** |
| **FT3.1.a** | Add Meeting Time and Location | UC0401 | Admin enters the designated meeting time and location. | Meeting time and location is stored in database and user is returned to view student form. |
| **FT3.1.b** |  |  | Time and location data is not valid. | Invalid data is cleared from the form and a fresh Meeting Time form is displayed. |
| **FT3.2.a** | Record Meeting Outcome | UC0402 | Select meeting number. | The meeting details for the respective meeting number are displayed. |
| **FT3.2.b** |  |  | User enters meeting outcome. | The meeting details are updated with the record. |
| **FT3.2.c** |  |  | User cancels the update. | If the user chooses to cancel instead of confirm the update, he is returned to the view student form. |

## 

## 4.5 Generate Reports

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Functionality** | **Use Case ID** | **Description** | **Expected Result** |
| **FT4.1** | View All Students | UC0501 | Select view all students. | Once selected it will query the database for all students then display them. |
| **FT4.2.a** | View All Students by Teacher | UC0502 | Select view all students by teacher option and enter a teacher’s name. | It will then search the database for the appropriate details and display a report. |
| **FT4.2.b** |  |  | No result is found, either teacher or students. | User is returned to the main form. |
| **FT4.3.a** | Custom Search | UC0503 | User enters search criteria. | Program will search the database for matching fields and then display the appropriate results. |
| **FT4.3.b** |  |  | No results are found. | If no results are found the user is returned to the search form. |

## 4.6 Prompt User with Updates

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Functionality** | **Use Case ID** | **Description** | **Expected Result** |
| **FT5.1.a** | New Student at Risk Report | UC0601 | New students at risk are found. | System will search the database for new students and display notifications to the user in regards to the new students at risk. |
| **FT5.1.b** |  |  | No new students found. | If no students are found the notification area will be kept empty. |
| **FT5.2.a** | Prompt user to complete outstanding reports | UC0602 | Select notification. | If a notification is selected it will display the outstanding reports details. |
| **FT5.2.b** |  |  | No notifications found. | If there are no notifications then there will in turn be no outstanding reports so it will show the main form. |

## 

## 4.7 Application Tools

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Functionality** | **Use Case ID** | **Description** | **Expected Result** |
| **FT6.1.a** | Validation |  | System doesn't accept the data entry. | If entered incorrectly an error message will be shown stating why it was not accepted. |
| **FT6.1.b** |  |  | System accepts the data entry. | Data entry is accepted and user is shown appropriate screen. |

## 

## 4.8 User Tools

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Functionality** | **Use Case ID** | **Description** | **Expected Result** |
| **FT7.1** | FAQ / Online Documentation |  | System will have a link to documentation that will help the user of the system resolve any queries or issues they have. |  |
| **FT7.2** | Mouse Over Tool tips |  | When user moves mouse onto a certain control or section of the form a tool tip will appear at the mouse’s position displaying some helpful hint/tips to the user. |  |

# 

# 

# 5 Non-Functional Test Plan

|  |  |  |  |
| --- | --- | --- | --- |
| **Test ID** | **Functionality** | **Description** | **Expected Result** |
| **NTF0.1** | Speed of Application | System performs application and database tasks in reasonable time dependant on the task at hand. | System runs efficiently and effectively in the given time. |
| **NTF0.2.a** | Stability | Attempt to perform multiple tasks at the same time. | System does not crash, or freeze. |
| **NFT02.b** |  | Multiple users performing tasks at the same time. | System does not crash, or freeze. |
| **NFT0.3** | Layout | Should be intuitive, clean, user-friendly, clear and informative. | System will convey information in a clear sense for the user. |
| **NFT0.4** | Ease of use | System should be easy to navigate and use. | User can navigate through the system easily and effectively without getting lost. |
| **NFT0.5** | MySQL database | Use of MySQL database with the system to be able to run on existing Swinburne servers. | System runs without error and connects to the server/database with succession. |
| **NFT0.6** | Room for expansion | System can be edited and modified by people outside the current project team to add extra functionality to the database and application. | System and database can be expanded easily and existing information/source is available to help the new project team. |
| **NFT0.7.a** | Consistency | System is consistent and all forms meet a standard and are generic in certain ways. | User can navigate through the system without being confused. |
| **NFT0.7.b** |  | Font specifically is consistent. | User should be able to have no problem reading any of the information or links in the system. |
| **NFT0.8** | Database Design | Database should be normalized to avoid problems when modifying at run-time. | User can modify the database without difficulties. |

# 

# 6 Client Acceptance Test Plan

The Client Acceptance Test will be conducted before the system goes live. Tests that will be conducted are available in the Client Acceptance Test document. This can be found on the cit3 server. The Client Acceptance Test will be used to receive client feedback on the CAESAR system.

# 

# 7 Requirements Traceability

The Requirements Matrix is a set of tables that must be checked off as the tests mentioned above are conducted. This will ensure all test have been conducted and signed off.

# 

# 8 References

1. **Cit3 main site:** <http://cit3.ldl.swin.edu.au/>
2. **Team Documents:** Located on team forum.
3. **Swinburne Major projects:** Past project documents