

RFID – Student Tracking System

Software Test Cases

Supervisor:

Dr. Usman Bhatti

Project Manger:

Fahad Tariq (06-0132)

Group Members:

AbdulRehman S. Durrani (06-0104)

Usman Zafar (06-0088)

M. Ahsan Iqbal (06-0102)

BS (CS) Final Year Project - II

Date: 14/03/2010

National University Of Computer and Emerging Sciences

Department of Computer Science

Lahore, Pakistan

# Test case Design and description

Following test cases have been made to ensure the quality of software. By quality we mean that all constraints must be followed regarding language that is C#, database that is SQL server, environment that is Visual Studio .Net and hardware that is RFID transceiver and tags. There should not be any issue for the users regarding data manipulation, use of software and hardware, starting and shutting down of application, time constraints, interfaces, navigation etc. Every user is given appropriate rights for the use of system, these test cases also verifies and validates those rights and privileges by applying boundary conditions in the testing methodology. After applying a test case, if the required result is not achieved then that test fails, otherwise it passes.

Every use case has been covered in these test cases and complete application has been tested. There are few test cases that cover more than one use case. There were few problems that were identified through testing and corrections were made and then test cases were applied again. Following is a list of test cases with detailed description given in ‘test case templates’ given below.

**12.1. Login**

**12.2. Student’s attendance**

**12.3. View (or search) attendance (or information records)**

**12.4. Insert data**

**12.5. Delete data**

**12.6. Modify/update data**

**12.7. Generate report**

**12.8. User blacklist**

**12.9. Navigation & Logout**

**12.10. Time constraints**

## Test case template

**Login**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID:** | | TC-1 | | QA Test Engineer | | Ahsan Iqbal |
| **Test case Version:** | | Version 1.1 | | **Reviewed By** | | Dr. Usman Bhatti |
| **Test Date:** | | 4/3/2010 | | **Use Case Reference(s)** | | UC-7 |
| **Revision History** | | N/A | | | | |
| **Objective** | | This test should only be passed in case of valid login | | | | |
| **Product/Ver/Module** | | Version 1.1/Login module | | | | |
| **Environment:** | | Visual Studio .Net | | | | |
| **Assumptions:** | | This use case is run for all users like student, faculty, staff and administrator for signing into the system for using software application and its functionalities. | | | | |
| **Pre-Requisite:** | | User has opened the application | | | | |
| **Step No.** | Execution description | | | | **Procedure result** | |
|  | Enter username, password and select user category. Then press Login button.  In case of successful login, appropriate GUI should be displayed according to user category. | | | | In case of valid username, password and user category, login is successful  After successful login, appropriate GUI is displayed | |
| **Comments**  If any of the username, password or user category entered is wrong, system does not login. For instance if a student enters correct username and password but selects user category as faculty, staff or administrator then system gives an error. Also appropriate GUI is displayed according to user category after successful login. | | | | | | |
|  | | | *Passed* *Failed* *Not Executed* | | | |

**Student’s attendance**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID:** | | TC-2 | | QA Test Engineer | | Ahsan Iqbal |
| **Test case Version:** | | Version 1.1 | | **Reviewed By** | | Dr. Usman Bhatti |
| **Test Date:** | | 6/3/2010 | | **Use Case Reference(s)** | | UC-1 |
| **Revision History** | | N/A | | | | |
| **Objective** | | This test should only be passed in case of successful attendance of student | | | | |
| **Product/Ver/Module** | | Version 1.1/Attendance module | | | | |
| **Environment:** | | RFID hardware & SQL server | | | | |
| **Assumptions:** | | This test is run for all use cases involving RFID tag identification through transceiver. For example: Student’s entrance in labs, examination hall, library, hostels etc. | | | | |
| **Pre-Requisite:** | | RFID transceiver is in working order | | | | |
| **Step No.** | Execution description | | | | **Procedure result** | |
|  | Take RFID tag in range of transceiver.  In case of successful attendance, appropriate message should be displayed through some beep/text to the student. | | | | In case of valid student roll number, after successful attendance, appropriate message is displayed to the student through transceiver screen and a beep. | |
| **Comments**  If RFID tag of a student is not blacklisted by the administrator and the transceiver is in working condition, attendance is marked in database and a response is given to the student by the transceiver in case of successful attendance. In case of an invalid tag id, it does not respond to the student and does not mark attendance in database. | | | | | | |
|  | | | *Passed Failed Not Executed* | | | |

**View (or search) attendance (or information records)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID:** | | TC-3 | | QA Test Engineer | | Ahsan Iqbal |
| **Test case Version:** | | Version 1.1 | | **Reviewed By** | | Dr. Usman Bhatti |
| **Test Date:** | | 6/3/2010 | | **Use Case Reference(s)** | | UC-8 |
| **Revision History** | | N/A | | | | |
| **Objective** | | This test should only be passed in case of successful view of attendance/records | | | | |
| **Product/Ver/Module** | | Version 1.1/View attendance/records module | | | | |
| **Environment:** | | Visual Studio .Net | | | | |
| **Assumptions:** | | This test case is run for users like student, faculty or hostel superintendent for viewing student’s attendance records and for administrator or librarian for viewing other information records of students. | | | | |
| **Pre-Requisite:** | | User has successfully logged into the system | | | | |
| **Step No.** | Execution description | | | | **Procedure result** | |
|  | Enter roll number of student.  Then press ‘view attendance/records’ button.  In case of valid student, attendance/records should be displayed to the user. | | | | In case of valid roll number of student, attendance/records are displayed to the user. | |
| **Comments**  If roll number of a student is blacklisted by the administrator then user will be given a message that attendance/records of this student is not available or if it’s available then it’s not up to date. | | | | | | |
|  | | | *Passed Failed Not Executed* | | | |

**Insert data**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID:** | | TC-4 | | QA Test Engineer | | Ahsan Iqbal |
| **Test case Version:** | | Version 1.1 | | **Reviewed By** | | Dr. Usman Bhatti |
| **Test Date:** | | 7/3/2010 | | **Use Case Reference(s)** | | UC-10 & UC-12 |
| **Revision History** | | N/A | | | | |
| **Objective** | | This test should only be passed in case of successful insertion of data in database | | | | |
| **Product/Ver/Module** | | Version 1.1/View attendance/edit information module | | | | |
| **Environment:** | | Visual Studio .Net | | | | |
| **Assumptions:** | | This use case is run for users like faculty, staff and administrator for inserting attendance or other information records respectively. | | | | |
| **Pre-Requisite:** | | User has successfully logged into the system | | | | |
| **Step No.** | Execution description | | | | **Procedure result** | |
|  | After following UC-8, UC-9 or UC-11, user can insert information/attendance according to the rights given to him/her. User enters the information in available fields of forms and then press ‘insert’ button.  In case of successful insertion of data, an appropriate message should be given to the user. | | | | In case of data satisfying all rules and checks, it is inserted in respective tables of database.  After successful insertion, an appropriate message is displayed to the user that data has been ‘saved’. | |
| **Comments**  If the data being inserted does not follow all the restrictions applied by database like primary key constraints, data types, length of data, mandatory fields etc, then data is failed to be inserted. In such case, an error message is displayed to the user. | | | | | | |
|  | | | *Passed Failed Not Executed* | | | |

**Delete data**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID:** | | TC-5 | | QA Test Engineer | | Ahsan Iqbal |
| **Test case Version:** | | Version 1.1 | | **Reviewed By** | | Dr. Usman Bhatti |
| **Test Date:** | | 9/3/2010 | | **Use Case Reference(s)** | | UC-10 & UC-12 |
| **Revision History** | | N/A | | | | |
| **Objective** | | This test should only be passed in case of successful deletion of data from database | | | | |
| **Product/Ver/Module** | | Version 1.1/View attendance/edit information module | | | | |
| **Environment:** | | Visual Studio .Net | | | | |
| **Assumptions:** | | This use case is run for users like faculty, staff and administrator for deleting attendance or other information records respectively. | | | | |
| **Pre-Requisite:** | | User has successfully logged into the system | | | | |
| **Step No.** | Execution description | | | | **Procedure result** | |
|  | After following UC-8, UC-9 or UC-11, user can delete information/attendance according to the rights given to him/her. User deletes the information available in fields of forms and then press ‘delete’ button.  In case of successful deletion of data, an appropriate message should be given to the user. | | | | In case of data satisfying all rules and checks, it is deleted from respective tables of database.  After successful deletion, an appropriate message is displayed to the user that data has been ‘deleted’. | |
| **Comments**  If the data being deleted does not follow all the restrictions applied by database like primary key constraints, mandatory information etc then data is failed to be deleted. In such case, an error message is displayed to the user. | | | | | | |
|  | | | *Passed Failed Not Executed* | | | |

**Modify/update data**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID:** | | TC-6 | | QA Test Engineer | | Ahsan Iqbal |
| **Test case Version:** | | Version 1.1 | | **Reviewed By** | | Dr. Usman Bhatti |
| **Test Date:** | | 9/3/2010 | | **Use Case Reference(s)** | | UC-10 & UC-12 |
| **Revision History** | | N/A | | | | |
| **Objective** | | This test should only be passed in case of successful updation of data in database | | | | |
| **Product/Ver/Module** | | Version 1.1/View attendance/edit information module | | | | |
| **Environment:** | | Visual Studio .Net | | | | |
| **Assumptions:** | | This use case is run for users like faculty, staff and administrator for updating attendance or other information records respectively. | | | | |
| **Pre-Requisite:** | | User has successfully logged into the system | | | | |
| **Step No.** | Execution description | | | | **Procedure result** | |
|  | After following UC-8, UC-9 or UC-11, user can modify information/attendance according to the rights given to him/her. User modifies the information available in fields of forms and then press ‘update’ button.  In case of successful updation of data, an appropriate message should be given to the user. | | | | In case of data satisfying all rules and checks, it is updated in respective tables of database.  After successful modification, an appropriate message is displayed to the user that data has been ‘updated’. | |
| **Comments**  If the data being updated does not follow all the restrictions applied by database like primary key constraints, data types, length of data, mandatory fields etc, then data is failed to be updated. In such case, an error message is displayed to the user. | | | | | | |
|  | | | *Passed Failed Not Executed* | | | |

**Generate report**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID:** | | TC-7 | | QA Test Engineer | | Ahsan Iqbal |
| **Test case Version:** | | Version 1.1 | | **Reviewed By** | | Dr. Usman Bhatti |
| **Test Date:** | | 9/3/2010 | | **Use Case Reference(s)** | | UC-13 |
| **Revision History** | | N/A | | | | |
| **Objective** | | This test should only be passed in case of successful generation of reports | | | | |
| **Product/Ver/Module** | | Version 1.1/View(or search) attendance/ information records module | | | | |
| **Environment:** | | Visual Studio .Net | | | | |
| **Assumptions:** | | This use case is run for all users like student, faculty, staff and administrator for generating reports of their required data. | | | | |
| **Pre-Requisite:** | | User has successfully retrieved data following some use case | | | | |
| **Step No.** | Execution description | | | | **Procedure result** | |
|  | After following UC-8, UC-9 or UC-11, user can generate reports of the available data by pressing ‘generate report’ button.  In case of successful generation of report, an appropriate message should be given to the user. | | | | In case of data satisfying all rules and checks for the generation of report, the preview of required report is displayed to the user.  After successful generation of report, an appropriate message is displayed to the user that ‘report has been generated’. | |
| **Comments**  If the data does not pass the criteria to be generated as a report, an error message is displayed to the user. If data is in the right format, its preview is displayed to the user and the user may get print out of that report. | | | | | | |
|  | | | *Passed Failed Not Executed* | | | |

**User blacklist**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID:** | | TC-8 | | QA Test Engineer | | Ahsan Iqbal |
| **Test case Version:** | | Version 1.1 | | **Reviewed By** | | Dr. Usman Bhatti |
| **Test Date:** | | 11/3/2010 | | **Use Case Reference(s)** | | UC-14 |
| **Revision History** | | N/A | | | | |
| **Objective** | | This test should only be passed in case of unsuccessful login and RFID access | | | | |
| **Product/Ver/Module** | | Version 1.1/Attendance/login module | | | | |
| **Environment:** | | Visual Studio .Net & RFID hardware | | | | |
| **Assumptions:** | | This use case is run for users like student, faculty or staff whose tag id or login information (username & password) has been blocked by the administrator. | | | | |
| **Pre-Requisite:** | | User has NOT logged into the system | | | | |
| **Step No.** | Execution description | | | | **Procedure result** | |
|  | If user has been blocked by the administrator, his/her tag should not be identified by the RFID transceiver and attendance should not be marked in database. Similarly user should not be able to sign into the software application. | | | | In case of a blacklisted user, an error message is displayed while accessing RFID hardware or software application. | |
| **Comments**  If a user has been blocked by the administrator but that user can still mark his/her attendance using RFID tag or can log into the application then this test case fails. | | | | | | |
|  | | | *Passed Failed Not Executed* | | | |

**Navigation & Logout**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID:** | | TC-9 | | QA Test Engineer | | Ahsan Iqbal |
| **Test case Version:** | | Version 1.1 | | **Reviewed By** | | Dr. Usman Bhatti |
| **Test Date:** | | 11/3/2010 | | **Use Case Reference(s)** | | N/A |
| **Revision History** | | N/A | | | | |
| **Objective** | | This test should only be passed in case of accurate navigation & successful logout | | | | |
| **Product/Ver/Module** | | Version 1.1/Logout module | | | | |
| **Environment:** | | Visual Studio .Net | | | | |
| **Assumptions:** | | This use case is run for all users like student, faculty, staff and administrator who are using the software application by signing into the system and navigating through various GUIs. | | | | |
| **Pre-Requisite:** | | User has successfully logged into the system | | | | |
| **Step No.** | Execution description | | | | **Procedure result** | |
|  | User should be given such an interface where navigation through GUIs and logging out of the system should not be a problem. Each GUI should have buttons like ‘LOGOUT’, ‘NEXT’, ‘PREVIOUS’. | | | | When user presses NEXT or PREVIOUS buttons, GUIs are displayed accordingly.  In case of pressing LOGOUT button, main/home page of application is displayed to the user where interface for signing into the system is provided | |
| **Comments**  If the user cannot navigate through different GUIs of the software application or if an option for logging out of the system is not provided on every GUI, then this test case fails. | | | | | | |
|  | | | *Passed Failed Not Executed* | | | |

**Time constraints**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID:** | | TC-10 | | QA Test Engineer | | Ahsan Iqbal |
| **Test case Version:** | | Version 1.1 | | **Reviewed By** | | Dr. Usman Bhatti |
| **Test Date:** | | 11/3/2010 | | **Use Case Reference(s)** | | N/A |
| **Revision History** | | N/A | | | | |
| **Objective** | | This test should only be passed if all processes meet their required time bounds | | | | |
| **Product/Ver/Module** | | Version 1.1/Timing module | | | | |
| **Environment:** | | Visual Studio .Net & RFID hardware access | | | | |
| **Assumptions:** | | This use case is run for all processes involving RFID hardware access and software application | | | | |
| **Pre-Requisite:** | | All software and hardware components of system are in working order | | | | |
| **Step No.** | Execution description | | | | **Procedure result** | |
|  | User should be able to access RFID hardware and software in limited time and should be able to get system response without much delay | | | | When user starts any process, processing is done in constant time by the system and instant response is given to the user. | |
| **Comments**  This testing involves RFID hardware access like attendance of student using tag and transceiver. Quick response from transceiver should be given to student within 1-2 seconds and attendance should be marked in database. Similarly other user processes should also be completed without any delay such as opening software application, login and logout, accessing database for purpose of insertion, editing, searching or retrieval of data, generating reports, navigation through GUIs, confirmation or error messages in case of successful or unsuccessful completion of a task respectively. | | | | | | |
|  | | | *Passed Failed Not Executed* | | | |