

MIDTERM PROJECT

TOPIC: Test Plan for Student Registration System for a University.

Name

ID

Tashriful Islam Khan

16-31004-1

Submitted to:

Farzana Bente Alam (Assistant Professor of AIUB)

Test Plan for Student Registration System for a University

1. TEST PLAN IDENTIFIER: FSF-SQT1.4

2. REFERENCES:

- ❖ Web based online course registration system
- ❖ Unit testing
- ❖ Functional testing
- ❖ Integration testing
- ❖ System testing
- ❖ Acceptance testing

3. INTRODUCTION:

This is a test plan for testing web-based online course registration for American International University-Bangladesh (AIUB). The primary focus of this plan is to develop a schema which can accommodate all the listed requirements and provide level of information that ensures that the system works properly.

There will be three level of testing such as Unit Testing, System/Integration Testing and Acceptance Testing in this System. The details for each level are addressed below and will be further defined in level specific plans.

The acceptance testing is required to take one month from the date of application delivery from system test and is to be done in parallel with the present application process.

4. TEST ITEMS

The list of the following items should be tested is given below:

A. Course Management

Courses are made from the basis of online registration system. All requirements regarding course management should be tested.

B. Section Selection

Every section belongs to a specific course in a specific semester of a specific year and every section should have a maximum number of slots for the students to register. All functionalities regarding Section Management should be tested.

C. Faculty Management

Any faculty can belong to single/multiple departments. All information regarding Faculty Management should be captured correctly and has to be tested properly.

D. Student Management

Every student can belong to single/multiple departments. All information regarding Faculty Management should be captured correctly and has to be tested properly.

E. Verification

Only verified users can use this system. Verification of every user should be tested.

F. Fault tolerance

System should be fault tolerant. The whole system work properly and not crash due to unexpected behaviour from user. If the system crashes, it should have the capability of restore the most important information.

5. SOFTWARE RISK ISSUES

Every software is linked with some unexpected risk issues that should be tested thoroughly. In this system the following risk may be appear so that risks have to test properly.

- A. The ability to restart the application in the middle of the process is acritical factor to application reliability. This is especially true in the case of the transmission file as once data is pulled from the mail box it is no longer available there and must be protected locally.
- B. Backup and recovery of files, local databases and rest restart of the process must be carefully checked.
- C. Database security and access must be defined and verified. All the basic security functionality has to be checked.
- D. Runtime data has to be preserved so that a student's data remains preserved in the case of a sudden system failure.

6. FEATURES TO BE TESTED:

We will perform testing on use case specification, functional requirement, and non-functional requirement of all use cases and functions. They include:-

- A. User (Teacher, Student, Administration) login and logout.
- B. Student shall be able to edit their profile (Name and ID are not editable).
- C. Student shall be able to display the previous semester courses, results and Faculty.
- D. Student shall be able to display the previous semester courses, results and Faculty
- E. Student shall be able to register for the upcoming Semester.
- F. Student shall select the Academic year and the Semester to register.
- G. Student shall be able to add courses to registration.
- H. Student shall be able to choose preferable section for a course.
- I. Student can remove a course after he/she has added this to system.

- J. Student can view all the courses and course schedule he/she has added for the upcoming semester.
- K. Advisor can permit a student to add/remove a course if the student has any constraints.
- L. Administrator can login to system and add a new course, delete an existing course or edit a course description for the selected semester.
- M. Administrator can open a new section or close a section for a course.
- N. Administrator can make a course list, edit a course list, and adjust number of credits per course list.
- O. A user can change password.
- P. The system shall display the schedule for any student (provided the semester details).
- Q. The system shall display all the details for a section (Professor Name, Course Title, Suggested text-books, reference text-books, Room Number, Class schedule etc)
- R. The system shall allow the users to identify the number of students in each section of a course.

7. FEATURES NOT TO BE TESTED

A. Network Security.

B. Performance Testing.

Performance Testing covers a broad range of engineering or functional evaluations where a material, product, or system is not specified by detailed material or component specifications: Rather, emphasis is on the final measurable performance characteristics.

C. Load Testing.

Load testing is the process of creating demand on a system or device and measuring its response.

D. Stress Testing.

Verify system response during maximum student logins.

E. Volume Testing.

Verify system response when Database at 90% capacity.

8. APPROACH

8.1 Testing Levels

The testing for the Student Registration System project will consist of Unit, System/Integration(combined) and Acceptance test levels. It is hoped that there will be at least one full time independent test person for system/integration testing. However, with the budget constraints and time line established; most testing will be done by the test manager with the development team's participation.

UNIT Testing will be done by the developer and will be approved by the development team leader. Proof of unit testing (test case list, sample output, data printouts and defect information) must be provided by the programmer to the team leader before unit testing will be accepted and passed on to the test person. All unit test information will also be provided to the test person.

SYSTEM/INTEGRATION Testing will be performed by the test manager and development team leader with assistance from the individual developers as required. No specific test tools are available for this project. Programs will enter into System/Integration test after all critical defects have been corrected. A program may have up to two Major defects as long as they do not impede testing of the program (I.E. there is a work around for the error).

ACCEPTANCE Testing will be performed by the actual end users with the assistance of the test manager and development team leader. The acceptance test will be done in parallel with the existing manual ZIP/FAX process for a period of one month after completion of the System/Integration test process.

Programs will enter into Acceptance test after all critical and major defects have been corrected. A program may have one major defect as long as it does not impede testing of the program (I.E. there is a work around for the error). Prior to final completion of acceptance testing all open critical and major defects MUST be corrected and verified by the user test representative.

8.2 Configuration Management/Change Control: Movement of programs from the development portion of the 'RED' system to the test portion of the 'RED' system will be controlled through the existing Configuration Management application process, 'EXTRACT'. This will ensure that programs under development and those in full test will have the same version controls and tracking of changes. The same extract process will be used to migrate the programs from the Development/Test 'RED' system to the production 'BLUE' system once all testing has been completed according to published plans and guidelines. All Unit and initial system testing will be performed on the Development AS/400 'RED' system. Once the system has reached a reasonable level of stability, no critical or major defects outstanding, initial pilot testing will be done on the production AS/400 'BLUE' system. All testing done on the 'BLUE' system will be done in a parallel mode with all controls set to prevent actual updating of the production files.

8.3 Test Tools

The only test tools to be used are the standard AS/400 provided utilities and commands.

- A. The Program Development Manager (PDM) will be used as the source version configuration management tool in conjunction with the in-house check-in/check-out control utility. The check-in/out utility is part of each developer's standard AS/400 access menu.
- B. The initial prototypes for the new screens will be developed using the AS/400 Screen Design Aid (SDA). The initial layout and general content of the screens will be shown to the sales administration staff prior to proceeding with testing and development of the screens.
- C. All editing, compiling and debugging will be done using the Source Entry Utility (SEU).
- D. Data acquisition will be from actual production files where available using the AS/400 data base copy command CPYF and its various functions. Additional data will be created and modified as needed using the Data File Utility (DFU). No changes will ever be made to actual production files under any circumstances.

8.4 Meetings

The test team will meet once every two weeks to evaluate progress to date and to identify error trends and problems as early as possible. The test team leader will meet with development and the project manager once every two weeks as well. These two meetings will be scheduled on different weeks. Additional meetings can be called as required for emergency situations.

8.5 Measures and Metrics

The following information will be collected by the Development team during the Unit testing process. This information will be provided to the test team at program turnover as well as be provided to the project team on a biweekly basis.

- A. Defects by module and severity.
- B. Defect Origin (Requirement, Design, Code)
- C. Time spent on defect resolution by defect, for Critical & Major only. All Minor defects can be totalled together.

The following information will be collected by the test team during all testing phases. This information will be provided on a biweekly basis to the test manager and to the project team.

- A. Defects by module and severity.
- B. Defect Origin (Requirement, Design, Code)
- C. Time spent on defect investigation by defect, for Critical & Major only. All Minor defects can be totalled together.
- D. Number of times a program submitted to test team as ready for test.
- E. Defects located at higher levels that should have been caught at lower levels of testing.

9. TEM PASS/FAIL CRITERIA

- A. All test cases completed.
- B. All major functionality works properly.
- C. 99% percentage of cases completed with a 2% of minor defects.

10. TEST DELIVERABLES

- A. Test strategy.
- B. Test Plan.
- C. Test cases.
- D. Test data.
- E. Defect or bug report.
- F. Test incident report.
- G. Test summary report.

11. REMAINING TEST TASKS

| Tasks | Assign to |
|---|----------------|
| Making budget and time schedule | Test Manager |
| Execute all the test cases and report defects | QA tester |
| Define severity and priority for each defect | AA tester |
| Create acceptance test plan | TM ,PM ,Client |

12. ENVIRONMENTAL NEEDS

For the test environment, a key area to set up includes-

- A. System and application.
- B. Test data.
- C. Database server.
- D. Front-end running environment.
- E. Client operating system.
- F. Browser.
- G. Hardware includes server operating system.
- H. Network.
- I. Documentation required like reference documents/configuration guides/installation guides/user manuals.

13. STAFFING AND TRAINING NEEDS

- A. There will be minimum one test manager and a tester during whole testing process.
- B. Test manager divide the test tasks and distribute among the test team members.
- C. For new members there should have a training session for learning about the environment and test application and tools.
- D. Train the members before new testing tools are used.

14. RESPONSIBILITIES

| Tasks | QA leader | Test leader | DEV. TEAM | TEST TEAM | Client |
|--|--------------|----------------|--------------|--------------|--------|
| Deciding the test budget and schedule | | X | X | X | X |
| Identifying the testing activities | | X | X | X | X |
| Planning the entire testing process | | X | X | X | X |
| Sharing updates on testing with PM | | X | X | X | X |
| Implementing test process | X | | X | X | X |
| Ensuring that test product documentation is complete | X | | X | X | X |
| Unit and integration testing | X | X | | X | X |

15. SCHEDULE

Time has been allocated within the project plan for the following testing activities. The Specific dates and times for each activity are defined in the project plan time line. The persons required for each process are detailed in the project time line and plan as well. Coordination of the personnel required for each task, test team, development team, management and Customer will be handled by the project manager in conjunction with the development and test Team leaders.

- A. Time allocate for review the projects
- B. Time allocate for test plan with review
- C. Time allocate for documentation make by test team personnel.
- D. Time allocated for both System/Integration and Acceptance test processes.

16. PLANNING RISKS AND CONTINGENCIES

- A. Lack of personnel resources when testing is to begin.
- B. Lack of hardware and software.
- C. Lack of tools and Application.
- D. Late delivery of software, hardware and tools.
- E. Changes on the design.
- F. Staff turnover.