1. **Test Plan Identifier:**

   Student Registration System, Version 1.1

2. **References:**

- SRS Document
- Technical Specification
- Requirements specifications
- Project plan
- Requirements and specifications
- Development and test process standards

3. **Introduction:**

The objective of this project is to develop a test plan for a Web-based Online Registration system. It has some general specifications to permit the students to register for the course or for a specific section to a specific course where the system will keep the information about every courses. Students must have to complete the pre-requisite otherwise he/she will not able to register for the course.

System will keep all the records of the students such as their

1. First Name
2. Last Name
3. Student ID
4. Gender
5. Address
6. Department
7. Phone number
8. Email

System will also store the information of the faculties such as their

1. First Name
2. Last Name
3. Faculty ID
4. Gender

5. Address

6. Department

7. Phone number

8. Email

9. Research Interests &

10. Degrees obtained and the corresponding University details

To be enrolled in a specific course details should be verified as following:

1. A student cannot register for two courses which occur at the same time
2. A student can register for courses which will be offered in specific semester one tries to register.
3. Students cannot enroll in a section if a maximum number of students of a section has reached
4. Sections should be deactivated at the end of any semester
5. Students cannot register for two sections belonging to the same course
6. Students should be awarded a grade at the end of every semester

So from above point of view, the interfaces between the following subsystems will be tested:

a) Login module
b) Registration module
c) Course Management module

There could be critical performance measures to be tested. It could be:

a) Response time for remote login
b) Response time when system is loaded with almost its capacity

4. Test Items:

- ✓ Login package, Version 1.1
- ✓ Registration package, Version 1.1
- ✓ Student's info package, Version 1.1
- ✓ Faculty's info package, Version 1.1
- ✓ Course management package, Version 1.1

5. Software Risk Issues:

- Interruption of third party software
- Errors in time of registration
- Absence of backup system and recovery of local databases in case of any failure
- New version of existing software
- Database security and access must be defined and verified

6. Features to be tested:

Student/User:

1. Check student Id and Password.
2. Provided password need to match.
3. When students need to change password, they must follow correct format.
4. Validate the student.
5. Student cannot try to set wrong password more than three times otherwise account will be confiscated automatically by the system.
6. Students are able to add courses, choose any preferable sections for courses.
7. Students can add or drop any courses with following some instructions.

Admin:

1. Can login to the system any time
2. Can open a new section or close a section for a course
3. Can add new course
4. Can make a course list and edit course list
5. Can adjust number of credits per course list

7. Features not to be tested:

- Networks and Hardware.
- Personal Information.
- Email

8. APPROACH (STRATEGY):

The test strategy presents the recommended approach to the testing of software applications. In earlier sections, there has been discussion regarding what will be tested; this section will describe how it will be tested.

8.1: Testing Types:

I. Database Integrity testing

a) Invoke each database access methods for requesting data
b) Check database to ensure that the correct data will be retrieved
c) Testing may require Database development environment to modify data

II. Unit testing

It will be done by developer and will be approved by team lead. All unit test information will also be provided to the test person.

III. Function testing

Testing of the application should focus on target requirements that can be traced directly to use cases. The goals of these tests are to verify correct data acceptance, processing and retrieval. This type of testing is based upon black box techniques that is to verify and validate by interacting with GUI interface.

IV. User Interface testing

This testing verifies a user's interaction with the software. The goal is to ensure that the user interface provides the user with the appropriate access and navigation through the functions of the applications.

V. Performance testing

This testing measures response times. The goal of the testing is to verify and validate the performance requirements have been achieved. It can be used to tune a system's performance as a function of conditions such as hardware configurations. It should be performed on a dedicated machine. This permits accurate measurement. The databases used for this testing should be actual size.

VI. **Load testing**

It measures the workload to evaluate the system's ability to continue to function properly under different workloads. The goal of this testing is to determine and ensure that the system functions properly beyond the expected maximum workload.

VII. Security testing

This testing ensures that only those users granted access to the system are capable of accessing the applications. So for this, the following steps need to be maintained:

a) Identify and list each user type
b) Create tests for each user type and verify permission
c) Access to the system must be reviewed with the system admin.

VIII. Configuration testing:

This testing verifies the operation of the software and hardware configurations.

8.2: Test Tools

- Microsoft Excel will be used to write test cases.
- QA Tool will be used to manage test cases and reporting.

8.3: Meetings

The test team will meet once every one week 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 one week as well. Additional meetings can be called as required for emergency situations.

9. Item Pass/Fail criteria:

Completion criteria are really a communication tool and an important aspect of "quality management" on the project. The test process will be completed when the project leader will be satisfied with the result of the test. At least, 90 percent of the test case must be passed. All functionalities must be covered in this test cases. A percentage with minor defects acceptable but make sure that it will not create major defects. If the software is accepted by the users and they are satisfied with the performance then the software will pass otherwise it will fail and have to check the software again.

10. Test Deliverables:

Test Deliverables are the artifacts which are given to the stakeholders of software project during the software development lifecycle. There are different test deliverables at every phase of the software development lifecycle. The different types of test deliverables are:

- ✓ Test cases
- ✓ Test plans for different kind of testing
- ✓ Testing strategy
- ✓ Test scripts
- ✓ Test data
- ✓ Test reports
- ✓ Test summary report
- ✓ Problem reports and corrective actions
- ✓ Error logs and execution logs.

11. Remaining test tasks:

- Acceptance test plan.
- Integration test plan
- Verify prototypes of screens and report
- Verify execution log file.
- Define turn over procedure.
- Verify summary reports.
- Verify defect logs.

Task	Assigned To
Create Acceptance Test Plan	Project Managers, Clients
Create Integration Test Plan	Project Managers, Developers
Verify Prototypes Of Screens	Project Managers, Clients
Verify Prototypes Of Report	Project Managers, Clients
Define Unit Test rule and procedures	Project Managers, Developers

12. Environmental needs:

- ➢ Hardware is needed to execute our system properly.
- ➢ Database is needed to execute this system.
- ➢ Necessary power Supply & hardware to keep the system running during blackouts.

- Necessary security and encryption tools needed to keep students & teachers information safe in database.
- Reliable communication link with security features.
- Features for handling misinformation or student affairs.

13. Staffing and Training Needs:

This section presents the recommended staffing needed for the registration system.

- Test Manager
- Test Designer
- System Tester
- Test System Admin
- Database Manager/Admin
- Operation Manager
- Implementer/Developer

During the project it is preferred that the test system administrator can assume the job of a full time test engineer with a part time test specialist to help with the system and acceptance testing phase of the project. Operation manager will define operations, attributes and associations of the test classes. If a test specialist is not available to continue the project, the test manager will assume his/her role.

In order to provide complete and proper testing the areas which will need to be addressed in terms of training are following:

I. The developers are needed to be trained for the basic operations of the interfaces.

II. The operations staff will also require on communication process.

III. The Test administrator and Project Manager needs to work together with a system admin and also, train on the internal working of a course assignment flow and find about accessible courses and prerequisite course needs.

14. Responsibilities:

Test Manager:

I. Provide technical direction
II. Acquire correct resources
III. Reporting & Reviewing

Test Designer:

I. Generate test plan
II. Generate test suite

System Tester:

I. Execute tests
II. Log results
III. Recovering errors

Test System Admin:

I. Administer the whole system
II. Manage worker to test systems

Database Manager:

I. Administer database data

Operation Manager:

I. Identify the test class
II. Identify test package

Developer:

I. Create test class and packages implemented in the test suite

The following table will give proper idea about the staffs and their responsibilities in further project breakdown:

Responsibilities	Test Manager	Project Manager	Development Team	Test Team	Client
Acceptance Testing, Documentation and Execution	•	•	•	•	
System Testing , Documentation and Execution	•		•	•	
Integration Testing , Documentation and Execution	•		•	•	
Unit Testing, Documentation and Execution	•		•	•	
System Layout Review	•	•	•	•	
User Interface Review	•	•	•	•	•
Test Procedures	•	•		•	
Data validation	•	•		•	

15. Schedule:

This section will be done later after gathering all the information to finally start the project. It will be created dividing certain mini tasks assigning to particular person for a specific period of time with a Gantt chart or pie chart.

16. Planning Risks and Contingencies:

To keep up right information allegation and for organization to send messages and messages to students, it should have been just editable by some admin with the goal that the update is recorded each time which requires a third client type as data entry personnel. To stay away from that, the developer team can make the plans early before the client proposing it and keep it as highlight and the time table can incorporate a particular slack time to the student management test task.