

Software Test Specification (STS)

Based on IEEE 829 – Adapted for Student Projects

Project Title: [Enter Project Name]

Team Name & Members: [List Members]

Advisor: [Instructor Name]

Version: [e.g., 1.0]

Date: [dd-mm-yyyy]

Table of Contents

- 1. Introduction
- 2. Test Items
- 3. Features to Be Tested
- 4. Features Not to Be Tested
- 5. Test Approach
- 6. Test Cases
- 7. Test Data
- 8. Test Environment
- 9. Responsibilities & Schedule
- 10. Deliverables
- 11. Appendices

1. Introduction

- Purpose of this document.
- Scope of testing (unit, integration, system, acceptance).
- Reference documents (SRS, SDS).

2. Test Items

- List of items to be tested (modules, features, subsystems).
- Example: Authentication Module, Database, Report Generator.

3. Features to Be Tested

- Identify all functionalities from SRS/SDS that will be tested.
- Example: Login, Registration, Report Export.

4. Features Not to Be Tested

- Features out of testing scope (due to time/resources).
- Example: Future extensions, external integrations not implemented yet.

5. Test Approach

- Testing strategy (manual/automated, black-box/white-box).
- Levels of testing (unit, integration, system).
- Tools used (JUnit, Selenium, Postman).

6. Test Cases

- Detailed list of test cases with inputs, expected outputs, and results.

Test Case ID	Feature	Input	Expected Output	Actual Result	Pass/Fail
TC-01	Login	Username=John, Password=123	User logged in	User logged in	Pass

7. Test Data

- Description of test datasets (sample accounts, sample files).
- Example: 10 dummy users with valid and invalid credentials.

8. Test Environment

- Hardware/software required for testing.
- Example: Windows/Linux OS, MySQL DB, Web browser version.

9. Responsibilities & Schedule

- Assign team members to testing tasks.
- Example: Ali → Unit Testing, Sara → Integration Testing.
- Include testing timeline.

10. Deliverables

- List of testing deliverables:
 - Test Cases Document
 - Test Execution Report
 - Defect Log

11. Appendices

- Glossary of terms (e.g., Regression Testing, Test Stub, Mock Data).
- References (IEEE standards, textbooks).

Evaluation

To be completed by the instructor or supervisor.

7. Project STS Report (12 points / 2 marks) – Week 15

Section	Excellent (Full Points)	Good (75%)	Fair (50%)	Poor (25% or below)	Points
1. Introduction (1 pt)	Purpose, scope, and references clearly stated.	Covers most but vague.	Minimal explanation.	Missing.	/1
2. Test Items (1 pt)	All modules/features listed clearly.	Most features listed.	Few features listed.	Missing.	/1
3. Features to Be Tested (1.5 pts)	Comprehensive list mapped to SRS/SDS.	Most key features covered.	Few features listed; weak traceability.	Missing.	/1.5
4. Features Not to Be Tested (0.5 pt)	Clearly explains out-of-scope features.	Partially explains.	Vague or incomplete.	Missing.	/ 0.5
5. Test Approach (1.5 pts)	Clear strategy (levels, tools, methods).	Mostly clear, some gaps.	Very limited explanation.	Missing.	/1.5

Section	Excellent (Full Points)	Good (75%)	Fair (50%)	Poor (25% or below)	Points
6. Test Cases (3 pts)	Well-structured test case table with inputs, expected/actual outputs, pass/fail. Covers positive & negative tests.	Several test cases included, some incomplete.	Few test cases, weak structure.	Missing.	/3
7. Test Data (1 pt)	Relevant datasets defined (valid & invalid).	Some datasets defined.	Minimal test data.	Missing.	/1
8. Test Environment (1 pt)	Hardware/software clearly described.	Partial description.	Very minimal.	Missing.	/1
9. Responsibilities & Schedule (1 pt)	Clear team roles & testing timeline.	Some roles/schedule mentioned.	Very vague or incomplete.	Missing.	/1
10. Deliverables & Appendices (1.5 pts)	All deliverables listed; glossary & references included.	Most listed, minor gaps.	Minimal mention of deliverables.	Missing.	/1.5

Note: If a student is listed as a project member at the beginning of the STS but their **contributions are not reflected** (e.g., no assigned test cases, no role in responsibilities/schedule), that student will **not be awarded the same score** as contributing members.