**This references the requirements spreadsheet (due to the fact that we combined the two of them into one big spreadsheet). If necessary West can create a timeline spreadsheet to reference instead but this should suffice. Detail testing as it should’ve/could’ve happened and then describe what we were able to accomplish.**

➢ How do you plan to test your software units, component, or  
system?  
➢ Test, analyze, inspect, or observe (T, A, I, O requirements)  
➢ Multiple test plans  
➢ Detailed Test Plans contain step-by-step action & expected  
response

**SAMPLE TEST PLAN (on canvas):**

You will have multiple test plans. Test plans are normally numbered.  
A test plan should reflect ‘HOW’ you plan to test, along with ‘WHAT’ you plan to test and  
“WHEN” you plan to execute the test.  
Sample Test Plan Outline  
1. Project Title: ABC  
2. Test Plan ###  
3. Test Objectives: Unit, Integration, or System test  
4. Test Approach:  
a. First test with correct input data. (ensures execution)  
b. Second test with out of boundary input date. (defect examination)  
c. Third test functionality. (ensure requirements are successfully met)  
d. Fourth test performance, stress, dependability (  
.  
5. Manual or Automated Test:  
6. Test tools: NONE  
7. Test environment: Joe’s laptop  
8. Test criteria: Requirement ID numbers from the requirements spreadsheet/ requirements  
traceability matrix. Functional or Non-Functional Requirements. Which requirements  
are you testing?  
9. Test schedule: When do you plan to test this. Sunday, March 9, 2025  
10. Test team (list by name): Note: the unit developer can not be the unit tester.

COMMENTS:  
1, Functional testing objective – Ensure software works as it should. Validate user workflows,  
data processing, and verify input/output parameters. Check for out of boundary alerts.  
2. Performance testing objectives – ensure software is efficient and can handle various loads  
(throughput and scalability)  
3. Security testing objectives – Uncover security flaws  
4. Usability testing – Verify user workflow

CS3300 PROJECT NAME TEST PLAN ###  
DATE  
A Test Strategy or a Detailed Test Plan has EACH input step identified along with the expected  
output. No button is pushed arbitrarily. These are very large documents. One such plan  
normally takes 2 to 3 days to produce, but for large complex systems it may take 7 to 10 days.  
➢