System Testing
User Acceptance Testing

CSCI 3300/5300

Review

- What are some of the testing types we talked about?
- What are some strategies for constructing unit tests?
- What types of integration testing techniques do you know?

System Testing

- What is the objective of System Testing?
- Starts after Integration testing is complete
- Requires a Test Plan – a document describing:
  - Scope:
    - Features to be tested
    - Non-functional testing to be done
    - Exit criteria
  - Approach
    - Test environment
    - Resources
    - How will the testing be done (hopefully, not manually)

System Test Scope

- Test against requirements
  - Expected scenario
  - The unexpected cases
- Test against user stories

As a [*user role*],
I need [*the ability to something*],
so that I can [*get some benefit or avoid some consequence*].

Business need

Given [*some initial context*],
when [*this certain type of event occurs*],
then ensure [*this particular outcome*].

Test description

Given [input | preconditions],
when [actions | triggers],
then [output | consequences].

Test specification template

Example

- [User Story] As an engineer, I want to run the simulation with a particular day's data rates, so that I can compare actual system performance with the simulation result.
- [Feature 1] Run simulation
- [Feature 2] Compare results of the simulation
- Construct separate test cases for each feature
  - Focused tests
  - Easier to identify and diagnose issues

[Feature 1] Run simulation

- [Test Description]
  - [Given] Particular day's data rates and system design
  - [When] Data rates are run through the simulation.
  - [Ensure that] Simulation runs to completion.
- [Test Case 1]
  - [Inputs] Monolithic system, full day of data rates
  - [Actions] Run simulation
  - [Consequences] Simulation processes all rates and terminates
- [Test Case 2]
  - [Inputs] Monolithic system, empty data rates file
- Construct more test cases

[Feature 2] Compare results of the simulation

- [Test Description]
  - [Given] Particular day's data rates and system design
  - [When] Data rates are run through the simulation and simulation completes
  - [Ensure that] Simulation produces correct results
- [Test Case 1]
  - [Inputs] Monolithic system, constant data rates
  - [Actions] Run simulation to completion
  - [Consequences] Reported latency and throughput are correct.
    - How do we know if they are correct?
- [Test Case 2]
  - [Inputs] Monolithic system, increasing data rates
- Construct more test cases

Non-Functional testing includes:

Non-Functional Testing

- **Security**: how well is the system safeguarded against deliberate and sudden attacks from internal and external sources,
- **Reliability**: the extent to which any software system continuously performs the specified functions without failure.
- **Survivability**: The parameter checks that the software system continues to function and recovers itself in case of system failure.
- **Availability**: The parameter determines the degree to which user can depend on the system during its operation.
- **Usability**: The ease with which the user can learn, operate, prepare inputs and outputs through interaction with a system.

Non-Functional Testing

- **Scalability**: the degree in which any software application can expand its processing capacity to meet an increase in demand.
- **Interoperability**: how well does a software system interface with other software systems.
- **Efficiency/Performance**: the extent to which software system can handle capacity, quantity and response time.
- **Flexibility**: The term refers to the ease with which the application can work in different hardware and software configurations. Like minimum RAM, CPU requirements.
- **Portability**: the flexibility of software to transfer from its current hardware or software environment.
- **Reusability**: It refers to a portion of the software system that can be converted for use in another application.

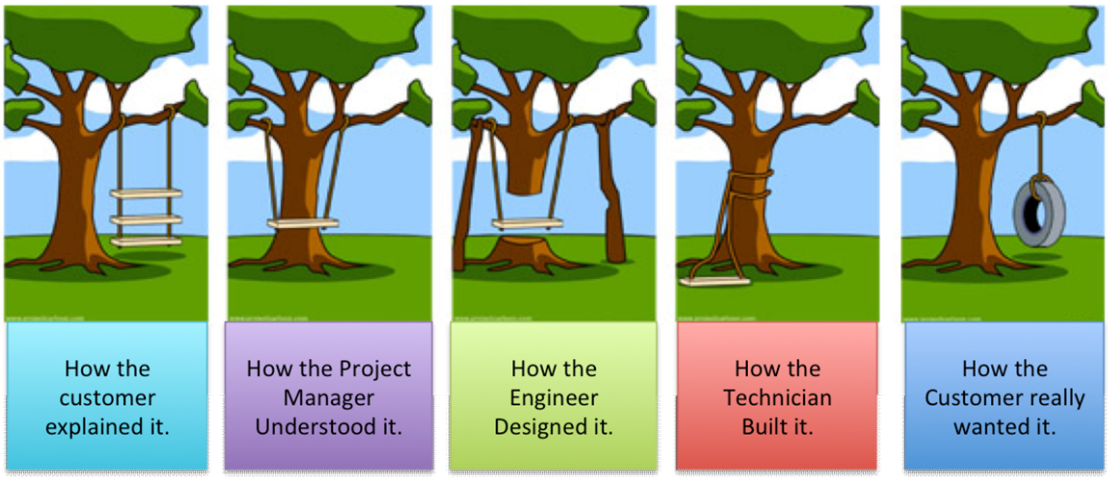
Examples of non-functional testing

Test Case #	Test Case	Domain
1	Application load time should not be more than 5 secs up to 1000 users accessing it simultaneously	Performance Testing
2	Software should be installable on all versions of Windows and Mac	Compatibility Testing

When do we stop testing?

- Can continue infinitely
- Completion/Exit/Success criteria must be defined as part of the test plan
  - Must be defined in objective terms
  - Concrete numbers/ranges (especially for non-functional testing)

From Requirements to User Acceptance Testing



User Acceptance Testing

- Sometimes known as *Beta* testing
- Testing of software by end-users to determine if it solves their need
- When does this happen in the Waterfall model?
- When does this happen in Agile?

Testing Summary

- Testing is a complex task
- Main Goal of testing: to identify flaws
- High Quality
  - takes time and commitment
  - does not happen by accident
- Testing must be a planned effort
- The sooner a flaw is identified, the easier it is to fix
- Aim to push your tests to the lowest possible level of the "testing pyramid"

Software Testing Trivia

The main focus of acceptance testing is

A. Finding faults in the system
B. Testing the system with other systems
C. Testing for a business perspective
D. Testing by an independent test team

Which testing phase tests individual software modules combined together as a group?

A. Module/Unit testing
B. Integration testing
C. White box testing
D. Black box testing
E. Software testing

A test harness is:

A. A high level document describing the principles, approach and major objectives of the organization regarding testing
B. A set of test activities collected into a manageable phase of a project
C. A test environment comprised of stubs and drives needed to conduct a test
D. A set of several test cases for a component or system under test

Which of the following is not part of performance testing?

A. Measuring response time
B. Measuring transaction rates
C. Recovery testing
D. Simulating many users
E. Generating many transactions

Which of the following would be a valid measure of test progress?

A. Number of undetected defects
B. Number of test cases not yet executed
C. Total number of defects in the product
D. Effort required to fix all defects

What is the purpose of test completion criteria in a test plan?

A. To know when a specific test has finished its execution
B. To ensure that the test case specification is complete
C. To set the criteria used in generating test inputs
D. To know when test planning is complete
E. To know when to stop testing

Which is NOT true: a block-box tester…

A. Should be able to understand a functional specification or requirements document
B. Should be able to understand the source code
C. Is highly motivated to find faults
D. Is creative to find the system's weaknesses

Beta testing is

A. Performed by customers at their own site
B. Performed by customers at their software developer's site
C. Performed by an independent test team
D. Performed as early as possible in the lifecycle