IS2545 Lecture 4

•••

Test Plans

Bill Laboon/Dustin Iser

The Customer

All stakeholders are customers of the test plan.

- Tester
  - The tester uses the test plan to validate that each of the requirements are tested.
  - The tester uses the test plan to organize her thoughts.
  - The tester uses the test plan as documentation. We need to be able to test features again, months or even years in the future.
- Developer
  - If the developer know what tests you are going to run, the developer can beta test his code before checking it in.
- Project manager
  - The project manager can use test plans as a tool to estimate deadlines or allocate more resources.
- Customer
  - Test plans can give the customer visibility into a project.

The right way to build a test plan

There is no right way to build a test plan.

The tester should build the test plan in such a way that it best serves the project's stakeholders.

Some examples from Dick's:

- Happy path vs. sad path
- BDD style
- Step-by-step

Test plan management

There are dozens of test plan management tools and there are new ones being developed everyday.

- TFS
- Jira
- Word documents
- Excel spreadsheets
- Notepad

Domain driven

The style and content of your test plan is driven by the domain of the software being built.

NASA vs. startup

Health care vs. game developer

Formal Test Plans

We will be building a formal test plan for deliverable 1.

A formal test plan is a sequence of test cases.

A test case is the fundamental ‘unit’ of a test plan.

Typically, a system will require several test plans.

- Login screen test plan
- Role admin test plan
- Checkout test plan
- Regression test plan

Test suite

A group of test plans is known as a test suite.

- DicksSportingGoods.com Test Suite
  - Login test plan
  - Shopping cart test plan
  - Checkout test plan
  - Loyalty point test plan
  - etc.

Test case

A test case mainly consists of:

- Preconditions
- Execution steps
- Postconditions

Example

Assuming an empty shopping cart, when I click ‘Buy Widget’, the number of widgets in the shopping cart becomes one.

Preconditions: Empty shopping cart

Execution steps: Click ‘Buy Widget’

Postconditions: Shopping cart displays one widget

Another example

Assuming that the SORT_ASCENDING flag is set, calling the sort method with [9,3,4,2] will return a new array sorted from high to low, i.e., [2,3,4,9].

Precondition: SORT_ASCENDING flag is set

Execution steps: Call .sort method with argument [9,3,4,2]

Postcondition: [2,3,4,9] is returned

Other test plan attributes

An identifier.

- Often a label, e.g. INVALID-PASSWORD-THREE-TIMES-TEST
- Could be a number

Description

- A description of the test case.

Test run

Once you've written your test plan, it must be executed.

The tester executes a test plan as part of the SDLC in order to validate the system meets requirements.

If a test run fails, a defect should be filed.

- Do not file duplicate defects

Test run statuses

- Passed
- Failed
- Paused
- Running
- Blocked
- Error

Things to consider

Start top-down. Divide the system into features and build a test plan for each.

All of your requirements should be ‘covered’ by the test plan.

- Requirement traceability matrix.

Test plan should ‘cover’ each of the equivalence classes.

Test plan should test happy and sad path.

Test plan should be efficient and shouldn’t include unnecessary tests.

Test cases should be independent of each other.

As a tester, ambiguity is your enemy.