OTHER TESTING APPROACHES

CS-HU 374 Lecture 10

https://www.slideshare.net/software-engineering-book/ch8testing – material provided for this class

TESTING IS A VIBRANT AND DIVERSE FIELD

In this class we just covered a few essential testing concepts and approaches

Overview of the following topics:

- Development testing
- Release testing
- Performance testing

DEVELOPMENT TESTING

Unit testing

- Individual program units or object classes are tested. Focus on testing the functionality of objects and methods

Component/integration testing

- Several individual units are integrated to create a composite unit. Focus on testing component interfaces.

System testing

- Some or all components in a system are integrated and tested as a whole. Focus on testing component interactions.

COMPONENT TESTING

Software components are often composite components that are made up of several interacting objects.

The functionality of these objects are accessed via established component interfaces.

Testing composite components should therefore focus on showing that the component interface behaves according to its specifications.

SYSTEM TESTING

System testing during development involves integrating components to create a version of the system and then testing the integrated system.

The focus in system testing is on the interactions between components.

System testing checks that components are compatible, interact correctly and transfer the right data at the right time across their interfaces.

System testing checks the emergent behavior of a system.

- Reusable components that have separately developed and off-the-shelf systems may be integrated
- Components developed by different team members or sub-teams may be integrated
- System testing is a collective rather than an individual process
- System testing may involve a separate testing team with no involvements from designers and programmers.

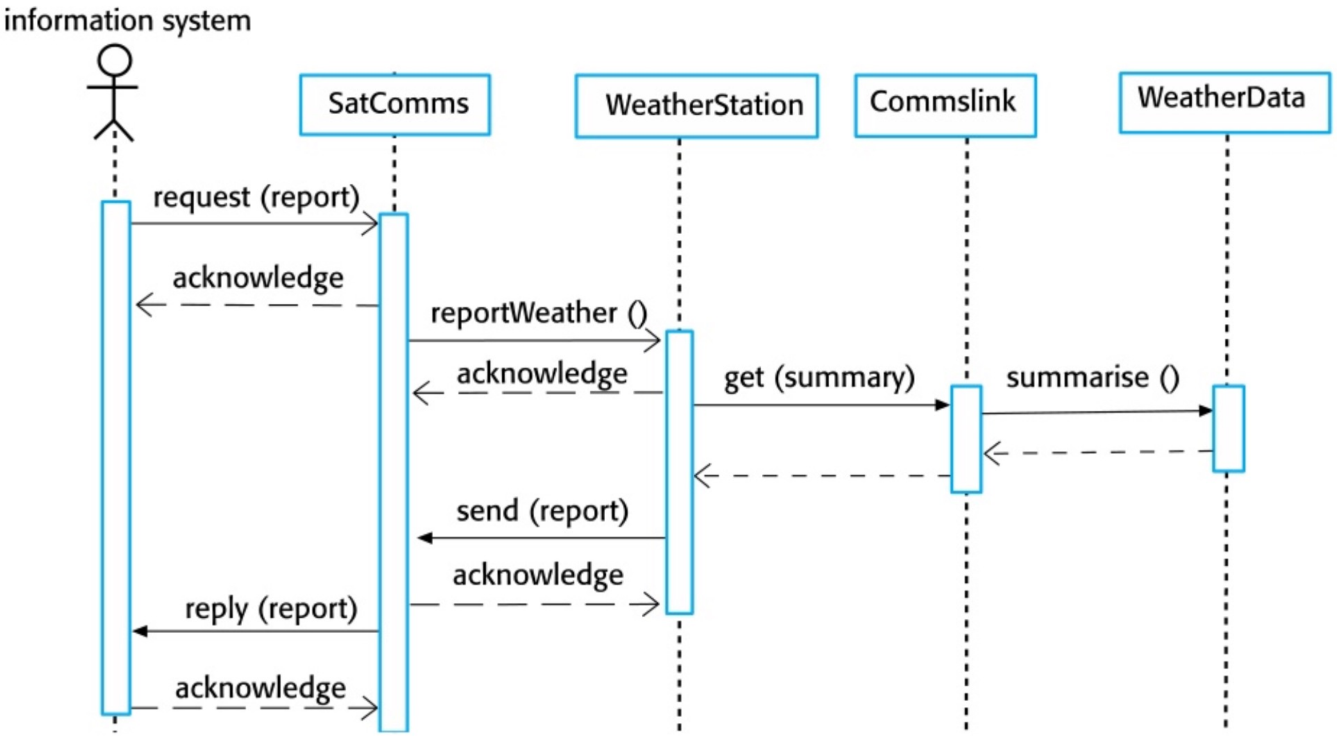
USE-CASE TESTING

The use-cases developed to identify system interactions can be used as a basis for system testing

Each use case involves several system components so testing use cases forces their interactions to occur

The sequence diagrams associated with the use case documents the components and interaction that are being tested.

A sequence diagram of a collect weather use case



RELEASE TESTING

Release testing is the process of testing a particular release of a system that is intended for use outside of the development team.

The primary goal of the release testing process is to convince the supplier of the system that it is good enough for use.

- Release testing, therefore, must show that the system delivers its specified functionality, performance and dependability, and that it does not fail during normal use.

Release testing is usually a black-box process where tests are only derived from the system specification.

Release testing is a form of system testing where

- A separate independent team is responsible for testing
- Focuses on meeting requirements and is good enough for external use (validation)

PERFORMANCE TESTING

Part of release testing may involve testing the emergent priorities of a system, such as performance and reliability.

Tests should reflect the profile of use of the system.

Performance tests usually involve planning a series of tests where the load is steadily increased until the system performance becomes unacceptable.

Stress testing is a form of performance testing where the system is deliberately overloaded to test its failure behavior.

USER TESTING

User or customer testing is a stage in the testing process in which users or customers provide input and advise on system testing.

User testing is essential, even when comprehensive system and release testing have been carried out.

- Users own working environments have a major effect
  - Reliability
  - Performance
  - Usability
  - Robustness
- Difficult to implement in a testing environment

Types of user testing

- Alpha – user works with the development team at the developer's site
- Beta – a release of the software made for users to experiment and raise problems
- Acceptance – customers test a system to decide whether it is ready to by deployed in the customer environment.

REFLECTION ON TESTING

In Canvas 20 to 30 minutes

Write a reflection on different test case generation approaches and testing strategies. Describe and compare them. (about 2 to 3 paragraphs).

In the last paragraph share your thoughts on what worked for you in this class or what did not. Focus mainly on the course content, assignments and activities.

Please complete course evaluation!