A Kamil Khan

Testing Strategies for Object Oriented Software

Unit Testing in an Object Oriented Context:

1. Smallest testable unit is the encapsulated class or object.
2. Because a class can contain different number of operations and a particular operation may exist as part of a number of different classes, the meaning of the Unit testing changes dramatically. This means the traditional unit testing approach is ineffective in the object oriented context.
3. Class testing for OO software is driven by the operations encapsulated by the class and the state behaviour of the class.

Integration Testing in an Object Oriented Context:

Because OO software does not have hierarchical control structure, Conventional Bottom up or Top down integration testing have little meaning.

Two Types of Strategies:

1. Thread Based Testing
   Integrates the set of classes required to respond to one input or event for the system. Each thread is integrated and tested individually
2. Use Based Testing
   Starts by testing classes (independent classes) that uses very few (if any) of server classes. After that dependent classes will be tested. Use of drivers and stub should be avoided.

Cluster Testing is one step in the integration testing of OO software. Here cluster of collaborated classes is exercised by designing test cases that attempt to uncover errors in the collaborations.

Validation Testing in an Object Oriented Context:

1. Validation of OO software focuses on user-visible actions and user-recognisable output from the system.
2. Tester should draw upon the use-cases that part of the analysis model.
3. The use-case provides a scenario that has a high likelihood of uncovered errors in user interaction requirements.
4. Conventional Black box testing can be used to drive validation tests.
5. In Addition, test cases may be derived from the object-behavioural model and from the even flow diagram.

Test case Design:

1. Each test case should be uniquely identified and explicitly associated with the class to be tested.
2. Purpose of the test should be stated.
3. A list of testing steps should be prepared.
   A. list of specific states for the objects to be tested.
   B. list of messages and operations that will be exercised as a consequence of the test.
   C. list of exceptions that may occur as the object is tested.
   D. list of external conditions
   E. supplementary information that will aid in understanding or implementing the steps.

Testing Methods available at the Class Level

1. Random Testing
2. Partition Testing at the Class Level
   A. State Based Partitioning
   B. Attribute Based Partitioning
   C. Category Based Partitioning

Testing Strategies for Web Applications:

The approach for WebApp testing adopts the basic principle for all software testing and applies a strategy and tactics that have been recommended for object oriented systems. The following steps summarises the approach

1.The content model for the WebApp is reviewed to uncover errors

a. similar to copy editing a written document

b. Might enlist the services of professional copy editor to uncover typographical errors, grammatical mistakes, errors in context consistency, errors in graphical representation and cross referencing errors

2. The design model for the WebApp is reviewed to uncover navigation errors

a. Use cases allow a web engineer to exercise each usage scenario against the architectural and navigation design

b. Navigation links are reviewed to ensure that they correspond with those specified in SNU for each User role.

3. Selected processing components and web pages are unit tested

a. Smallest testable Unit is web page.

b. driven by content, processing and links encapsulated by the Web Page

4. Architecture and Integration tests are conducted.

a. in Linear, grid or simple hierarchical structure, integrate web pages as same as integrate modules in conventional software.

b. Mixed Hierarchy or network architecture, integrating testing is similar to OO system.

c. Thread Based testing can be used to integrate set of web Pages. Each thread is integrated and tested separately. Regression testing is applied to make sure no side effects occur.

d. cluster testing integrates set of collaborating pages

5. Assembled WebApp is tested for overall functionality and content delivery

a. User visible actions and user recognisable output from the system is tested.

b. Use cases are written.

6. Tested for different environmental configuration

a. Cross reference matrix that defies all probable operating system, browsers, hardware platforms and communication protocols is created.

b. Tests are conducted to uncover errors associated with each possible configuration.

7. WebApp is tested by controlled and monitored population of end users.

a. population of users that encompasses every possible user is chosen.

b. WebApp is exercised by these users and the results of their interaction with the system are evaluated for the content and navigation errors, usability concerns, and webApp reliability and performance