**Chapter 8 : Software testing**

**8.1**



**Testing is a process, which shows a program does what it is intended to do and to discover program defects before it is put into use. "It is not necessary for a program to be completely free of defects before it s delivered to customers "it is that "it is not always (99% of the time) possible to deliver a defect free product to the customer which is of 100% quality.**

**And testing cannot completely validate that a system is fit for its intended purpose as this requires a detailed knowledge of what that purpose will be and exactly how the system will be used. As these details inevitably change between deciding to procure a system and deploying that system, the testing will be necessarily incomplete.**

**In addition, it is practically impossible for all except trivial system to have a complete test set that covers all possible ways that the system is likely to be used.**

**A program need not be completely free of defects before delivery if:**

**1. Remaining defects are minor defects that do not cause system corruption and which are transient i.e. which can be cleared when new data is input.**

**2. Remaining defects are such that they are recoverable and a recovery function that causes minimum user disruption is available.**

**3. The benefits to the customer's business from the system exceed the problems that might be caused by the remaining system defects.**

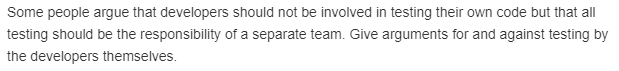
**8.2**



**Testing can detect only the presence of errors, not their absence because the main goal of the testing is “to observe the behavior of the particular software and to check whether it meet its requirement expectation or not”.**

**Testing is a part of broader process of software verification and validation. It consists of a set of activities, where the testers try to make the software behave anomalous in order to detect or anomaly to be later fix. Testing cannot demonstrate the faults other than specified in every circumstance. It is always possible that a test have overlooked could discover further problem with the system.**

**8.3**



**Some people argue that developers should not be involved in testing their own code but that all testing should be the responsibility of a separate team.**

**Arguments for the testing by developers are:**

**• Testing becomes easy as they know the areas of code which need testing.**

**• Generating test case will also becomes easy as they have a clear view of the entire code and the test cases that can be helpful.**

**• Saves time, as they need not learn about the code.**

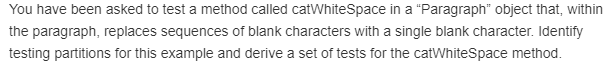
**Arguments against the testing by developers are:**

**• One may not easily identify their own errors.**

**• Testing may be compromised so as to reduce their error rate**

**• Increases the effort of developers and may compromise with quality**

**8.4**



**Given example scenario is to test a method called “catWhiteSpace” in a “Paragraph” object that, within the paragraph, replaces sequences of blank characters with a single blank character.**

**Testing partitions for the scenario:**

**• Sentences with only single blank characters.**

**• Sentences with sequences of blank characters in the middle of the sentence.**

**• Sentences with sequences of blank characters at the beginning of the sentence.**

**• Sentences with sequences of blank characters at the end of the sentence.**

**Set of example tests for the method ‘catWhiteSpace’:**

**• Testing can only show the presence of errors, not their absence. (only single blank characters)**

**• Testing can only show the presence of errors, not their absence. (several blank spaces in the sentence)**

**• Testing can only show the presence of errors, not their absence. (2 blanks at the beginning)**

**• Testing can only show the presence of errors, not their absence . (2 blanks at the end)**

**• Testing can only show the presence of errors, not their absence. (several blanks at the beginning)**

**• Testing can only show the presence of errors, not their absence . (several blanks at the end)**

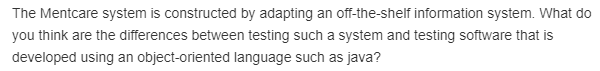
**8.5**



**Regression testing: It is a test suite developed incrementally as a program is developed. Regression testing involves running test sets that have successfully executed after changes have been made to a system.**

**Using the automation frameworks, the entire test suite can be run in a few seconds so it is possible to execute all the tests every time you make a change to the program. Automated testing dramatically reduces the costs of regression testing. These help the regression test to be easier and reduce effort.**

**8.6**



**Difference between testing an off-the-shelf information system and object oriented software exists in their testing methodologies and various tests applied.**

**The differences are:**

**• An application system such as MHC-PMS must be tested by considering each requirement and every requirement carefully and by developing test cases. It should be tested that the applications works well in all the cases.**

**• Object oriented software is tested in the perspective of classes, objects, their binding, dependency etc, whereas off-the-shelf information system applications must be tested in various perspectives such as requirements satisfaction, user acceptance, fault tolerance etc.**

**• The applications adapting an off-the-shelf information system are to be tested from environmental perspectives. For example if drug dosage is given as a wrong input, it affects the patient. All such conditions are to be tested thoroughly, whereas object oriented software may not result in such harmful effects.**

**• These testing process of the off-the shelf systems must concentrate more on scenario testing, performance testing and stress testing. All these tests are to be performed because any error in these applications may affect more and may cause loss.**

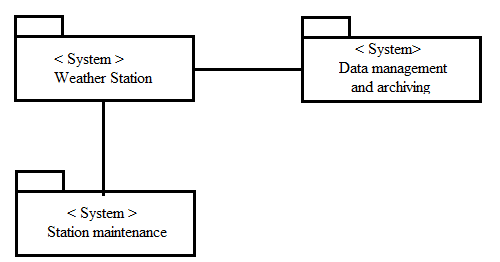
**8.7**



**Scenario-based analysis of wilderness weather station system:**

**The weather station is composed of independent subsystem that communicates by broad casting messages on a conman infrastructure. Weather stations collect data from a set of instruments that measure temperature and pressure, sunshine, rainfall, wind speed and wind direction.**

**The weather station includes a number of instruments that measure weather parameters such as the wind speed and direction, the ground and air temperatures, the barometric pressure and the rainfall over a 24-hour period. Each of these instruments is controlled by a software system that takes parameter readings periodically and manages the data collected from the instruments.**



**In this objects, attributes and methods are identified. A weather station is a package of software controlled instruments which collect data. For designing a test, it is necessary to understand about the interactions between the system and its environment. Testing is performed by checking the data processing and transmits ground thermometers, an anemometer, a wind vane a barometer and a rain gauge.**

**Steps for testing:**

**1. Identify the objects, attributes and methods.**

**2. Understand about the interactions between the system and its environment.**

**3. Check the data processing and transmits ground thermometers, an anemometer, a wind vane a barometer and a rain gauge.**

**4. Check the major components that make up the system and their interactions, and then may organize the components using an architectural pattern such as a layered or client- server model.**

**5. Check whether the command is issued to transmit the weather data, the weather station processes and summaries the collected data .the summation data is transmitted to the mapping computer when a request is received.**

**8.8**



**Stress testing is a form of performance testing where the system is deliberately overloaded to test its failure behavior. The goal of stress testing is to identify application issues that arise or become apparent only under extreme conditions. These conditions can include heavy loads, high concurrency, or limited computational resources.**

**Stress-testing scenarios for a MHC-PMS:**

**MHC-PMS system can be tested by increasing the dosage levels and frequency. Test the system by inputting the dosage levels more than the safe limits. It shows the results how the system reacts under high frequency and high dosage conditions.**

**8.9**



**Benefits of involving users in release testing at an early stage:**

**• As the system release is for customers and users, they can easily identify the modifications needed.**

**• Helps in getting the views of the users and the modifications needed from the users’ perspective.**

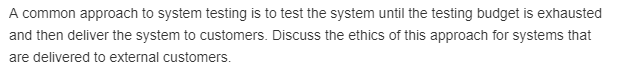
**Disadvantages of user involvement:**

**• May not get a better feedback as the views and ideas of users vary from person to person.**

**• May increase the effort of testing as the users may not have clear view of the system.**

**• Testing perspective may sometimes be deviated to some other view.**

**8.10**



**Ethics of testing the system until the testing budget is exhausted are:**

**• The testing may not be done satisfactorily.**

**• Only few methods of testing are covered with the available budget.**

**• Only few areas of code may be tested because of the running budget. So there may be some uncovered errors.**

**• The testing may not be completed in a satisfactory way by the time the budget exhausts**

**• The quality of the software will be affected by the incomplete testing.**

**• Due to improper testing, errors may occur after delivering the project.**

**• It increases the costs of error correction and maintenance.**

**• Affects the reputation and good will.**

**• The customer may not be satisfied with the product as he may not get a quality product.**