Sessional Assignment

Course Name: Object Oriented Software Engineering

Submitted By:

Musab Awais (13028)

BS (SE-8) Section: A

Submitted To:

Mam Sanaa Jeehan

Dated: 20th June 2020

Department of Computer Science,
IQRA National University, Peshawar Pakistan

Object Oriented Software Engineering
SE – 5th Semester

TERM ASSIGNMENT

Question #1 (10 Marks)

In Software Engineering, there is not a single answer to the question "What should be done first, Coding or Modeling?". Elaborate different scenarios in which all the answers to this questions are justified.

Question #2 (10 Marks)

When carrying out Testing of a Software, a number of techniques are used. Why are they so many in number? Name a few popular Testing Techniques in Software Engineering and state the importance of each one.

Instructions for Assignment Submission

1. Write your names and Ids at the top of each paper of answer sheet.
2. Scan / Take Photo of each paper and save each photo with a number. E.g. photo of page 1 of answer sheet be saved with name 1.jpg, then 2.jpg and so on.
3. Put all answer photos in a folder, name the folder with your Roll Number, Name and Subject Name, e.g. "11512 - Sanaa Jeehan – OOSE". Alternately, you can also make a PDF file of all the pictures and name it as explained.
4. Zip the folder and upload.

Question # 1:
Answer:

Both modeling and coding are very important in software engineering. Mostly modeling is done first because it makes it easier for the coders to code when they know the model of the project they are working on. The more complex your project, the more likely it is that you will fail or that you will build the wrong thing if you do on modeling at all. But sometimes coding is done first because of shortage in time. Doing coding first may result in many errors and failures as we don't have a clear visual of the system. Following are some scenarios for both situations:

<u>Scenario in which Modeling is done first:</u>

1. When a software house is given a big project to make software for an organization, they will do modeling first after gathering the requirements from the client because modeling can help the development team better visualize the plan of the system that is required and allow them to develop more rapidly by helping them build the right thing. After that the software engineer will start coding by following the models given to them.

2. In a university when students are required to build their final year project they are needed to do modeling first in order to describes the software system itself, with the allocation of responsibilities to its various parts, and its behavior and control flow. After that they will start coding which will become easier for them as they will already know the different aspects of their project.

<u>Scenario in which Coding is done first:</u>

1. Consider a client, who wants urgent software for his organization, he wants the software house to build the software in short amount of time. The software house will collect the requirements from the client and will start coding directly as they will have short amount of time to complete their project. Then they will deploy the software into the clients system.

2. Sometimes in universities when project deadline is near and the students are late in starting the project, the students will start coding first as to deliver the project in time. After the project is complete they will build the model of the project at the end in order to give information of their project to their teachers and supervisors.

Question # 2:
Answer:

Why so many Software Testing techniques:

There are so many software testing techniques because they're required to point out the defects and errors that were made during the development phases. Software Testing is necessary because we all make mistakes. Some of those mistakes are unimportant, but some of them are expensive or dangerous. We need to check everything and anything we produce because things can always go wrong. It's essential since it makes sure that the customer finds the organization reliable and their satisfaction in the application is maintained. It is very important to ensure the Quality of the product. Quality product delivered to the customers helps in gaining their confidence. It's important to ensure that the application should not result into any failures because it can be very expensive in the future or in the later stages of the development.

Software Testing Techniques:

- **Functional Testing:**
  This type of testing ignores the internal parts and focuses only on the output to check if it is as per the requirement or not.

- **Non-Functional Testing:**
  It is a type of testing for which every organization having a separate team which usually called as Non-Functional Test (NFT) team or Performance team. Non-Functional Testing involves testing of non-functional requirements such as Load Testing, Stress Testing, Security, Volume, Recovery Testing, etc. The objective of NFT testing is to ensure whether the response time of software or application is quick enough as per the business requirement. It should not take much time to load any page or system and should sustain during peak load.

- **White Box Testing:**
  White Box Testing is based on the knowledge about the internal logic of an application's code. It is also known as Glass box Testing. Internal software and code working should be known for performing this type of testing. Under these tests are based on the coverage of code statements, branches, paths, conditions, etc.

- **Graphical User Interface (GUI) Testing:**
  The objective of this GUI Testing is to validate the GUI as per the business requirement. The GUI Testing includes the size of the buttons and input field present on the screen, alignment of all text, tables, and content in the tables. It also validates the menu of the application, after selecting different menu and menu items, it validates that the page does not fluctuate and the alignment remains same after hovering the mouse on the menu or sub-menu.

- **Gorilla Testing:**
  Gorilla Testing is a testing type performed by a tester and sometimes by the developer as well. In Gorilla Testing, one module or the functionality in the module is tested thoroughly and heavily. The objective of this testing is to check the robustness of the application.

- **Happy Path Testing:**
  The objective of Happy Path Testing is to test an application successfully on a positive flow. It does not look for negative or error conditions. The focus is only on the valid and positive inputs through which application generates the expected output.

- **Incremental Integration Testing:**
  Incremental Integration Testing is a Bottom-up approach for testing i.e. continuous testing of an application when new functionality is added. Application functionality and modules should be independent enough to test separately. This is done by programmers or by testers.

- **Install/Uninstall Testing:**
  Installation and Uninstallation Testing is done on full, partial, or upgrades install/uninstall processes on different operating systems under different hardware or software environment.

- **Integration Testing:**
  Testing of all integrated modules to verify the combined functionality after integration is termed as Integration Testing. Modules are typically code modules, individual applications, client and server applications on a network, etc. This type of testing is especially relevant to client/server and distributed systems.

- **Load Testing:**
  It is a type of Non-Functional Testing and the objective of Load Testing is to check how much load or maximum workload a system can handle without any performance degradation. Load Testing helps to find the maximum capacity of the system under specific load and any issues that cause software performance degradation. Load testing is performed using tools like JMeter, LoadRunner, WebLoad, Silk performer, etc.

- **Monkey Testing:**
  Monkey Testing is carried out by a tester assuming that if the monkey uses the application then how random input, values will be entered by the Monkey without any knowledge or understanding of the application. The objective of Monkey Testing is to check if an application or system gets crashed by providing random input values/data. Monkey Testing is performed randomly and no test cases are scripted and it is not necessary to be aware of the full functionality of the system.

- **Mutation Testing:**
  Mutation Testing is a type of white box testing in which the source code of one of the program is changed and verifies whether the existing test cases can identify these defects in the system. The change in the program source code is very minimal so that it does not impact the entire application, only the specific area having the impact and the related test cases should able to identify those errors in the system.

- **Negative Testing:**
  Testers having the mindset of "attitude to break" and using Negative Testing they validate that if system or application breaks. A Negative Testing technique is performed using incorrect data, invalid data or input. It validates that if the system throws an error of invalid input and behaves as expected.

- **Performance Testing:**
  This term is often used interchangeably with 'stress' and 'load' testing. Performance Testing is done to check whether the system meets the performance requirements. Different performance and load tools are used to do this testing.

- **Recovery Testing:**
  It is a type of testing which validates how well the application or system recovers from crashes or disasters. Recovery Testing determines if the system is able to continue the operation after a disaster. Assume that application is receiving data through the network cable and suddenly that network cable has been unplugged. Sometime later, plug the network cable; then the system should start receiving data from where it lost the connection due to network cable unplugged.

- **Regression Testing:**
  Testing an application as a whole for the modification in any module or functionality is termed as Regression Testing. It is difficult to cover all the system in Regression Testing, so typically Automation Testing Tools are used for these types of testing.

- **Risk-Based Testing (RBT):**
  In Risk-Based Testing, the functionalities or requirements are tested based on their priority. Risk-Based Testing includes testing of highly critical functionality, which has the highest impact on business and in which the probability of failure is very high. The priority decision is based on the business need, so once priority is set for all functionalities then high priority functionality or test cases are executed first followed by medium and then low priority functionalities. The low priority functionality may be tested or not tested based on the available time. The Risk-Based Testing is carried out if there is insufficient time available to test entire software and software needs to be implemented on time without any delay. This approach is followed only by the discussion and approval of the client and senior management of the organization.

- **Sanity Testing:**
  Sanity Testing is done to determine if a new software version is performing well enough to accept it for a major testing effort or not. If an application is crashing for the initial use then the system is not stable enough for further testing. Hence a build or an application is assigned to fix it.

- **Security Testing:**
  It is a type of testing performed by a special team of testers. A system can be penetrated by any hacking way. Security Testing is done to check how the software or application or website is secure from internal and external threats. This testing includes how much software is secure from the malicious program, viruses and how secure and strong the authorization and authentication processes are. It also checks how software behaves for any hackers attack and malicious programs and how software is maintained for data security after such a hacker attack.

- **Static Testing:**
  Static Testing is a type of testing which is executed without any code. The execution is performed on the documentation during the testing phase. It involves reviews, walkthrough, and inspection of the deliverables of the project. Static Testing does not execute the code instead of the code syntax, naming conventions are checked. Static Testing is also applicable for test cases, test plan, design document. It is necessary to perform static testing by the testing team as the defects identified during this type of testing are cost-effective from the project perspective.

- **Stress Testing:**
  This testing is done when a system is stressed beyond its specifications in order to check how and when it fails. This is performed under heavy load like putting large number beyond storage capacity, complex database queries, and continuous input to the system or database load.

- **System Testing:**
  Under System Testing technique, the entire system is tested as per the requirements. It is a Black-box type testing that is based on overall requirement specifications and covers all the combined parts of a system.

- **Unit Testing:**
  Testing of an individual software component or module is termed as Unit Testing. It is typically done by the programmer and not by testers, as it requires detailed knowledge of the internal program design and code. It may also require developing test driver modules or test harnesses.