Course Code: 102007052
Course Title: Software Engineering
Instructor: Engr. Ghassan Husnain
Program: BS CS (Software Engineering)   **Total Marks: 30 Time Allowed: 4 Hours**
Note: Attempt all Questions

Student name: syed bashir ahmad shah

Student ID: 12996

Q.1: What are the four important attributes that all professional software should have? Suggest four other attributes that may sometimes be significant.   *(4 marks)*

Answer: Maintainability: Software should be written in such a way so that it can evolve to meet the changing needs of customers. This is a critical attribute because software change is an inevitable requirement of a changing business environment.

Dependability and security: Software dependability includes a range of characteristics including reliability, security and safety. Dependable software should not cause physical or economic damage in the event of system failure. Malicious users should not be able to access or damage the system.

Efficiency: Software should not make wasteful use of system resources such as memory and processor cycles. Efficiency therefore includes responsiveness, processing time, memory utilization, etc

Acceptability: Software must be acceptable to the type of users for which it is designed. This means that it must be understandable, usable and compatible with other systems that they use.

Four other attributes that can be useful are, usability, portable, robust, appropriate etc.

Q.2: Explain why professional software is not just the programs that are developed for a customer.   *(4 marks)*

Answer: Professional software is not just the programs developed for a customer because the software is almost always packaged with associated documentation such as requirements, design models, and user manuals.

A good or professional software goes beyond software developed solely for a customer. It is usually composed of executable code coupled with documentation and configuration of data that is required to make these programs operate correctly. A professionally developed software system is often more than a single program. The system usually consists of a number of separate programs and configuration files that are used to set up these programs. It may include system documentation, which describes the structure of the system, user documentation, which explains how to use the system, and websites for users to download recent product information. Example a word processing system consist of executable program, user manual and the document such as requirements and the design needed to produce the executable program.

Q.3: Giving reasons for your answer based on the type of system being developed, suggest the most appropriate generic software process model that might be used as a basis for managing the development of the following systems: (*12 marks*)

- A system to control anti-lock braking in a car
- A virtual reality system to support software maintenance
- A university accounting system that replaces an existing system
- An interactive travel planning system that helps users plan journeys with the lowest environmental impact

Answer: (a) Anti-lock braking system: This is a safety critical system so requires a lot of up-front analysis before implementation. It certainly needs a plan-driven approach to development with the requirements carefully analysed. A waterfall model is therefore the most appropriate approach to use, perhaps with formal transformations between the different development stages.

(b) Virtual reality system: This is a system where the requirements will change and there will be an extensive user interface components. Incremental development with, perhaps, some UI prototyping is the most appropriate model. An agile process may be used.

(c) University accounting system: This is a system whose requirements are fairly well known and which will be used in an environment in conjunction with lots of other systems such as a research grant management system. Therefore, a reuse-based approach is likely to be appropriate for this.

(d) Interactive travel planning system: System with a complex user interface but which must be stable and reliable. An incremental development approach is the most appropriate as the system requirements will change as real user experience with the system is gained.

2

Q.4: Explain why incremental development is the most effective approach for developing business software systems. Why is this model less appropriate for real-time systems engineering? *(5 marks)*

Answer: Business software systems are usually complex, software intensive, and frequently being changes when business goals or processes are changed. So incremental development is better.

Real-time systems usually involve many hardware components which are not easy to change and cannot be incremental. Also real-time systems usually safety critical which needed to be built based on well planned process.

Q.5: Suggest why it is important to make a distinction between developing the user requirements and developing system requirements in the requirements engineering process. *(5 marks)*

Answer: There is a fundamental difference between the user and the system requirements that mean they should be considered separately.

(a) The user requirements are intended to describe the system's functions and features from a user perspective and it is essential that users understand these requirements. They should be expressed in natural language and may not be expressed in great deal, to allow some implementation flexibility. The people involved in the process must be able to understand the user's environment and application domain.

(b) The system requirements are much more detailed than the user requirements and are intended to be a precise specification of the system that may be part of a system contract. They may also be used in situations where development is outsourced and the development team need a complete specification of what should be developed. The system requirements are developed after user requirements have been established.

_____