

SLIATE

SRI LANKA INSTITUTE OF ADVANCED TECHNOLOGICAL EDUCATION

(Established in the Ministry of Higher Education, vide in Act No. 29 of 1995)

Higher National Diploma in Information Technology

Second Year, First Semester Examination – 2015

HNDIT23082- Software Engineering

Marking Schemes

1.

III

- i. List four main differences in software engineering compared to other engineering disciplines.
- It is difficult for a customer to specify requirements completely.
 - It is difficult for the developer to understand fully the customer needs.
 - Software requirements change regularly.
 - Software is primarily intangible; much of the process of creating software is also intangible, involving experience, thought and imagination.
 - It is difficult to test software exhaustively.

04.

[04 Marks]

✓ ii

List four ways how software engineering provides solutions to the differences you have mentioned in (i) above.

- Greater emphasis on systematic, scientific development.
- Computer assistance in software development (CASE)
- A concentration on finding out the user's requirements
- Formal/Semi Formal specification of the requirements of a system.
- Demonstration of early version of a system (prototyping)
- Greater emphasis on development of error free easy to understand code.

08

i

iii What is the main difference between Current usefulness and Potential Usefulness in Boehm's Classification of software quality attributes.

[04 Marks]

03

Current usefulness is the qualities expected from a software system in user's point of view.
Potential Usefulness the qualities expected from a software system in developer's point of view.

[07 Marks]

Q List three qualities for each of the Current usefulness and Potential Usefulness.

- Current usefulness
 - Efficiency
 - Reliability
 - Usability
 - Correctness
 - User friendliness
 - Robustness
- Potential usefulness
 - Maintainability
 - Modularity
 - Reusability
 - Portability

Q6

[06 Marks]

Q Give two (04) reasons why that Software need to be changed.

- Errors in the system.
- Changes in the user requirements
- Availability of new technology
- Changes in the enterprise or Govt. policy.

Q8

[04 Marks]

[Total 25 Marks]

2.

i. Briefly explain three (03) characteristics in software process model.

- Understandability

To what extent is the process explicitly defined and how easy is it to understand the process definition?

2

- Visibility
Do the process activities culminate in clear results so that the progress of the process is externally visible?
- Acceptability
Is the defined process acceptable to and usable by the engineers responsible for producing the software project?
- Reliability
If the process is designed in such a way that process errors are avoided or trapped before they result in product errors?
- Rapidity
How fast can the process of delivering a system from a given specification be completed?
- Robustness
Can the process continue in spite of unexpected problems?
- Maintainability
Can the process evolve to reflect changing organizational requirements or identified process improvements?
- Supportability
To what extent can the process activities be supported by CASE tools?

Any three answers

[06 Marks]

ii. State three problems associated in:

- a. Waterfall Model
- b. RAD Model

Waterfall model

- Real projects rarely follow the sequential flow that the model proposes. Although the Waterfall model can accommodate iteration, it does so indirectly.
- It is often very difficult for the customer to state all requirements explicitly. The Waterfall model has the difficulty of accommodating the natural uncertainty that exists at the beginning of many projects.
- The customers must have patience. A working version of the program(s) will not be available until late in the project time-span. A major blunder, if undetected until the working program is reviewed, can be disastrous.

[03 Marks]

RAD Model

- RAD requires sufficient human resources to create right number of RAD teams

- RAD requires developers and customers who are committed to the rapid-fire activities necessary to get a system completed in a much abbreviated time frame.
- If a system cannot be properly modularized, building the components necessary for RAD will be problematic.
- RAD is not applicable when technical risks are high. This occurs when a new application makes heavy use of new technology or when the new software requires a high degree of interoperability with existing computer programs.

[03 Marks]

iii. List down four processes in RAD model

- Business modelling
- Data Modelling
- Process Modelling
- Application generation
- Testing and turnover

Any four answers

[04 Marks]

iv. Briefly describe about Spiral model

This model is an evolutionary software process model that couples the iterative nature of prototyping with the controlled and systematic aspects of the linear sequential model.

Using the spiral model software is developed in a series of incremental releases. During early iterations, the incremental release might be a paper model or prototype.

[05 Marks]

v. Briefly explain about prototype and list down two techniques used in that model.

A prototype (a small version of the system) can be used to clear the vague requirements. A prototype should be evaluated with the user participation.

There are two types of Prototyping techniques

- * Throw-away Prototyping
- * Evolutionary Prototyping

[04 Marks]

[Total 25 Marks]

3.

i. What is feasibility study?

Short, focused study to check the feasibility of the project considering resources, cost/benefit, technology availability etc.

[02 Marks]

ii. What will technical software development staff find out after “**Requirement elicitation and analysis**”?

- application domain,
- what services the system should provide,
- the required performance of the system,
- Hardware constraints and so on.

[04 Marks]

- iii. At the end of the **requirements specification process** what is the deliverable?
A detailed and precise description of the system requirements is set out or SRS document
- [02 Marks]
- iv. The software requirements document is an official statement of what the system developers should implement. It should include both the user requirements for a system and a detailed specification of the system requirements.
- a. Identify three requirement documents? [03 Marks]
- User requirements (Requirements definition)
System requirements
Software Requirements Specification (SRS)
- b. List two readers from each of above in (a).
User requirements (Requirements definition)
- i. Client managers
ii. System end-users
iii. Client engineers
iv. Contractor managers
v. System architects
Any two from them
- System requirements
- i. System end-users
ii. Client engineers
iii. System architects
iv. Software developers
Any two from them
- Software Requirements Specification (SRS)
- i. System architects
ii. Software developers
- [3*02 = 06 Marks]

- v. What is requirement validation?
Requirements validation is the process of checking that requirements actually define the system that the customer really wants.
- [05 Marks]

vi. During the requirements validation process, different types of checks should be carried out on the requirements in the requirements document. List three of them.

Validity checks, Consistency checks, Completeness checks, Realism checks,
Verifiability

Any three of the above

[03 Marks]

[Total 25 Marks]

4.

i. Briefly explain following Object Oriented design concepts?

- a. Abstraction
- b. Polymorphism
- c. Encapsulation
- d. Modularity

a. Abstraction

- Abstraction is the concept of taking some object from the real world, and converting it to programming terms.
- Abstraction is the concept of moving the focus from the details and concrete implementation of things, to the types of things (i.e. classes), the operations
- Abstraction allows us to proceed with the development work without being held up in low-level implementation details

[02 Marks]

b. Polymorphism

- Refers to a programming language's ability to process objects differently depending on their data type or class
- It is the ability to redefine methods for derived classes
- Concept that refers to the ability of a variable, function or object to take on multiple forms

[02 Marks]

c. Encapsulation

- Minimizing interdependencies among separately written modules by defining strict external interfaces
- The technique of making the fields in a class private and providing access to the fields via public methods
- Protective barrier that prevents the code and data being randomly accessed by other code defined outside the class

d. Modularity [02 Marks]

- Software is divided into separately named, addressable components called modules
- Complexity of a program depends on modularity

[02 Marks]

[02*4=08 Marks]

ii. Why, High Cohesion and Low Coupling characteristics are important to design the good software?

Coupling refers to the relationship of a module with another module. A module is said to be highly coupled with another module if changes to it will result to changes to the other module. And a module is said to be loosely coupled if a module is independent of any other modules. Benefits of low coupling are

- maintainability – changes are confined in a single module
- testability – modules involved in unit testing can be limited to a minimum
- readability – classes that need to be analyzed are kept at a minimum

Cohesion refers to the measure of how strongly-related the functions of a module are. Low cohesion refers to modules that have different unrelated responsibilities. High cohesion refers to modules that have functions that are similar in many aspects. The benefits of high cohesion are

- Readability – (closely) related functions are contained in a single module
- Maintainability – debugging tends to be contained in a single module
- Reusability – classes that have concentrated functionalities are not polluted with useless functions

[04 Marks]

iii. Briefly explain following software architecture models?

- a. Client Server
- b. Repository

Client Server Model

- Distributed system model which shows how data and processing are distributed across a range of processors
- A set of stand-alone servers which offer services to other sub-systems
- A set of clients that call on the services offered by the servers

- A network which allows the clients to access these services

[02 Marks]

Repository Model

- Keep all shared data in a central database that can be accessed by all subsystems
- A system model based on a shared database is called repository model.
- This model is suited to applications where data is generated by one subsystem and used by another.
- It is efficient way to share large amount of data
- Activities such as backup recovery, access control and recovery from error are centralized. They are the responsibility of the repository manager.

[02 Marks]

[2*2 =04 Marks]

iv. State five (05) Interface design principles.

- User familiarity
- Consistency
- Recoverability
- User guidance
- User diversity

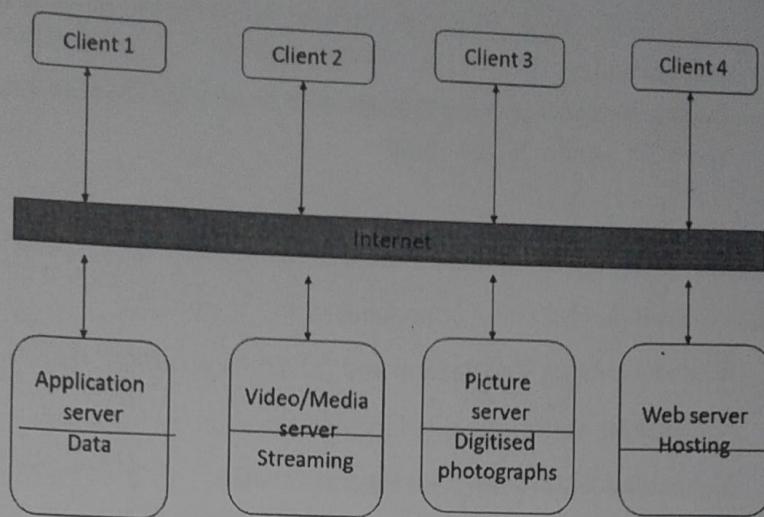
[05 Marks]

v. Draw a diagram showing conceptual view and a process view of the architecture of the following case study.

A Computer-controlled video conferencing system that allows video, audio and computer data to be visible to several participants at the same time. In this system, several servers manage and display the different type of media. Video frames need to be transmitted quickly and in synchrony but at relatively low resolution.

Still picture, however must be maintained at a high resolution, so it is appropriate

to maintain them on separate server.



[04 Marks]

[Total 25 Marks]

5.

i. Briefly explain the following terms?

a. Configuration Management

Discipline for evaluating, coordinating, approving or disapproving, and implementing changes in artifacts that are used to construct and maintain software systems. Development and application of standards and procedures for managing an evolving software product

[02 Marks]

b. Version

An instance of a system which is functionally distinct in some way from other system instances. Simple naming scheme uses a linear derivation V1, V1.1, V1.2

[02 Marks]

c. Release

The simplest thing to do is start your initial development release at 0.1.0 and then increment the minor version for each subsequent release.

[02 Marks]

d. Configuration Management Plan

Describe the standards and procedures which should be used for configuration management.

Include the configuration management items such as set of general, company-wide CM standards, data base

[02 Marks]

- ii. V model is one of the best practices used in the software development projects to identify the errors. Explain what is the importance of the V model?

- V model contain Verification and validation process.
- Verification is the process of identify errors in each phases
- Validation confirm the final product errors
- Consists static and dynamic verification and validation
- Dynamic concepts can be use for the each phase
- Static technique can be used at all stages
- The developer can start with unit testing and end with system testing

(Any 4 answers)

[04 Marks]

Ques

Briefly explain the following three type of maintenance?

a. Corrective

Activities undertaken to detect, isolate, and rectify a fault so that the failed equipment, machine, or system can be restored to its normal operable state.

To repair software faults (fixing bugs in the code)

(Any answer of them)

[02 Marks]

b. Adaptive

Maintenance to add to or modify the system's functionality

Modifying the system to satisfy new requirements

Modifying the system to suit new operation environment

(Any answer of them)

[02 Marks]

c. Perfective

Improving programs performance, structure, and reliability and making changes to avoid future problems or prepare for future changes

Includes monitoring and adjusting the operation of all system components

(Any answer of them)

[02 Marks]

IV

What is stress testing? Give an example?

To ensure the workload of the system. Test beyond the maximum design load until the system fails.

Example

Distributed system based network of processors.

Web service in online application

Transaction processing system

V

[02 Marks]

Network Diagram is the most interesting concept in project scheduling.

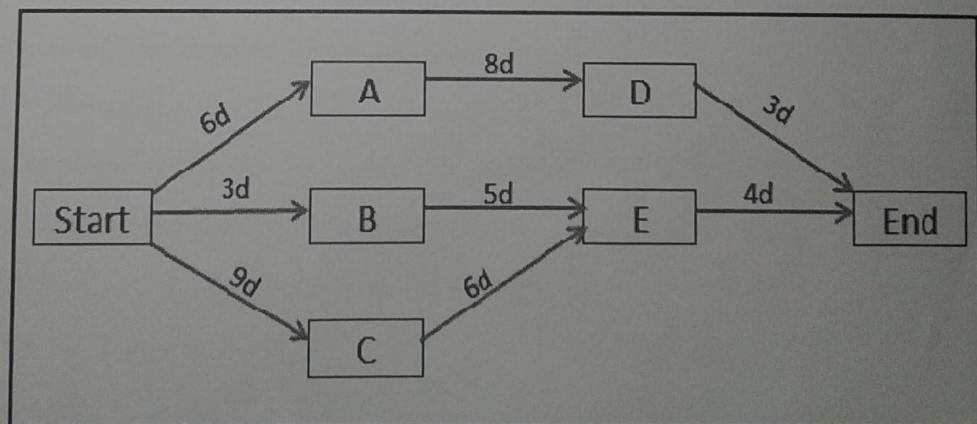
Activity	Predecessor	Duration (days)
A	Start	6
D	A	8
B	Start	3
E	B	5
C	Start	9
E	C	6
End	D	3
End	E	4

a. Draw the activity network diagram for the above table.

[03 Marks]

b. Find the Critical Path?

[02 Marks]



[Correct diagram, 03 Marks]

a. Path Start-C-E-End with the longest duration of 19 days is the critical path.

[02 Marks]

[Total 25 Marks]