

**D6 – INFORMATION SYSTEMS DEVELOPMENT.
SOLUTIONS & MARKING SCHEME.**

June 2013

The purpose of these questions is to establish that the students understand the basic ideas that underpin the course. The answers should be largely descriptive and quite short.

Answer A1.

Compare and contrast the role of the systems analyst, business analyst, and infrastructure analyst in a systems development team.

Addresses L01 – AC1.4. The three roles emphasize different perspectives on the system (1). The business analyst represents the sponsor/users interests (1), while the systems analyst knows how to apply IS to support business needs (1). Together, the systems analyst and the business analyst can design a system that conforms to the IS standards while adding value to the business (1). The infrastructure analyst has more technical knowledge and provides the team with technical constraints, or identifies infrastructure changes that the new system will require (1).

(Total 5 marks)

Answer A2.

Identify and distinguish between the phases, steps, techniques, and deliverables associated with a typical systems development lifecycle.

Addresses L01 – AC1.1. Phases are broad groupings of activities performed in the process of developing an information system (1). Generally, we define four phases: planning, analysis, design, and implementation (1). Within each phase, the required activities or tasks are outlined as a series of steps that guide the work to be performed (1). Steps are accomplished by applying the appropriate techniques, or ways to carry out the tasks (1). Deliverables are the understanding and/or specific materials that are produced that represent the accomplishment of a step (1).

(Total 5 marks)

Answer A3.

Explain how a process model can be presented as a set of Dataflow Diagrams. What role does decomposition play in the construction of this type of model?

Addresses L02 – AC2.1. Most business processes are too complex to depict using one diagram (1). Consequently, business processes are typically depicted with a set of DFD, with the first diagram (Context Level) showing a summary of the system, and subsequent DFD showing processes within that system (1). Decomposition is a method for breaking down a business process into smaller, logical processes (1). A simple illustrative example should be used for a top mark (2).

(Total 5 marks)

Answer A4.

Provide an illustrative example to explain why we might use a decision tree and/or a decision table in a process description?

Addresses L06 – AC6.3. A decision tree is useful in that it aids in understanding decision logic pertaining to nodes (questions) and branches (answers (1 plus 1 for example)). A decision table aids in understanding the actions (business policies) that based on a condition or a set of condition (1 plus 2 for example). Many candidates may offer a financial model, but other examples are possible.

(Total 5 marks)

Answer A5.

What type of high-level business rules can be represented in an ERD? Provide two illustrative examples.

Addresses L02 – AC2.4. A business rule is a constraint or guideline to follow during operation of the system (1). Examples of business rules are: an order belongs to just one customer; a customer cannot cancel an order that has been shipped; a backorder can be created for an out of stock product (1 for each suitable example). Business rules are expressed on ERD by the kinds of relationships that the entities share (1).

(Total 5 marks)

Answer A6.

Why is metadata important to system developers? Give some examples of metadata that would be stored when constructing object oriented models.

Addresses L03. Metadata is information we want to collect regarding the components of the object model. This would include names, signatures for methods as well as any pre-conditions, post-conditions or other constraints (3). Metadata helps us more fully understand the meaning and use of the object model components (1). Since there are typically several members of the project team, specifying metadata helps ensure that each team member has a consistent understanding of the data model components (1). Metadata is usually stored in the project repository; CASE tools have their own structures for the entry of metadata.

(Total 5 marks)

Answer A7.

What is the purpose of developing use cases during systems analysis?

Addresses L05 – AC5.3. The purpose of a use case is to illustrate the activities that are performed by the users of the system, (1) and is often thought of as an external or functional view of a business process (1). Use cases are developed during systems analysis activities to help the analysts better understand the situation (1) and simplify later modeling steps in the analysis phase (1) they are also important in developing test plans (1).

(Total 5 marks)

Answer A8.

What are three fundamental parts of most user interfaces?

Addresses L04 – AC4.1

Navigation mechanism - the way the user gives instructions to the system and tells it what to do (1 plus 1 for example).

Input mechanism - the way in which the system captures information (1 plus 1 for example).

Output mechanism - the way the system provides information to the user or to other systems.(1)

(Total 5 marks)

PART B.

Answer B9.

Addresses L01.

What is stakeholder analysis? Discuss three stakeholders that would be relevant for most systems development projects. (6 marks)

Stakeholder analysis is a systematic process that identifies all parties that will be affected by a new information system (1), and attempts to estimate the consequences of the project for each stakeholder group (1). A major goal of stakeholder analysis is to ensure that the consequences of a new system are considered for all parties that will be affected by the system (1). The most common stakeholders to consider for most systems projects are the system champion, the system users, and the organization's management (1). The system champion is the person or group who initiates the project and provides support for it. The users are the individuals who will work with the system once it is implemented (1). The organization management commits resources to the project and has an interest in seeing those resources be used to improve the functioning of the organization (1).

What are the six general skills that all members of a systems development team should have? (14 marks)

[1] Technical skills (knowledge of how to employ technology in development system solutions). (2).

[2] Business skills (knowledge of how to apply IT to business problems to achieve a valuable solution). (3).

[3] Analytical skills (ability to solve complex problems). (2).

[4] Interpersonal skills (oral and written communication skills with both technical and non-technical audiences). (3).

[5] Management skills (ability to manage others and cope with an uncertain environment). (2).

[6] Ethical skills (ability to deal with others honestly and ethically). (2).

(Total 20 marks)

Answer B10.

Addresses L05 and L06

Describe the principal steps in the following phases of systems development. What are some major deliverables associated with each phase?

Systems Analysis (10 marks)

2 marks for each step:

Step 1 – Analysis Strategy: based on the nature of the project, the project team will formulate the approach that will be used to develop the requirements for the new system.

Step 2 – Analyze the current system: gather information from the project sponsor and users of the current system regarding its strengths and weaknesses. Use the problems identified to formulate objectives for the new system.

Step 3 – Create new system concept: based on the gathered information, develop a general concept of the new system, including functions and capabilities it will have.

Step 4 – Modeling activities: express ideas for the new system's processing and data requirements with process models and data models.

Step 5: Prepare and present system proposal: assemble the analysis results, system concept, process model and data model into a proposal for the new system. Project sponsor and/or approval committee will determine if system has enough merit to continue development.

The primary deliverable for the analysis phase is the system proposal, which combines the information generated during this phase into a document that expresses the initial conceptual design for the new system and the basis for the design decisions.

Systems Design (10 marks)

2 marks for each step:

Step 1 – Design Strategy: based on the nature of the project, the project team will determine the appropriate means of developing the system (in-house custom development, purchase of pre-written software, or outsourcing development to a 3rd party. Following this, the steps below outline the various design tasks that must be performed:

Step 2 – Design the system architecture: describe the basic hardware, software, and networking that will be used in the new system.

Step 3 – Design the user interface: the overall structure of the system, the user's navigation through the system; the inputs and outputs of the system, and the appearance of the screens are designed.

Step 4 – Design the database and/or files: develop specifications for the storage structures that will be implemented for the new system.

Step 5: Design the programs: develop plans and outlines for each program that will be written to implement the functions and capabilities of the new system.

The primary deliverable for the design phase is the system specification, which combines all the design specifications mentioned above. The system specification is the basis for the construction work that will be performed by the programmers.

(Total 20 marks)

Answer B11.

Addresses L02.

Define what is meant by the following components of a data flow diagram. Explain what information should be stored about each of these components in a CASE repository.

A process (5 marks)

A process represents actions that are performed for some specific business reason. A process should be named using a verb phrase; information regarding a process to be stored in the CASE repository includes:

Label (name), Type (process), Description (what it is), Process Number, Process Description (Structured English), Notes, A data flow (5 marks)

A data flow represents a single piece of data or a set of logically-related data items that move to or from processes. A data flow should be named using a noun; information regarding a data flow to be stored in the CASE repository includes:

Label (name), Type (flow), Description , Alias, Composition (description of data elements), Notes

A data store (5 marks)

A data store represents a set of data that is stored together - the data store holds the data. A data store should be named using a noun; information regarding a data store to be stored in the CASE repository includes:

Label (name), Type (store), Description, Alias, Composition (description of data elements)

Notes

An external entity (5 marks)

An external entity is something that is outside the scope of our system, but interacts with it. An external entity may be a person, organization, or another system that supplies information to the system and/or receives information from the system. An external entity should be named using a noun; information regarding an external entity to be stored in the CASE repository includes:

Label (name), Type (entity), Description, Alias, Notes (Total 20 marks)

Answer B12.

Assesses L04

What do you think are three common mistakes that novice analysts make in interface design? (6 marks)

The following or similar should be identified:

Failing to focus on the most common paths through the interface (1)

Making the interface too crowded (1)

Failing to think about whether the primary users of the system are casual, occasional users or frequent, experienced users (1)

Being inconsistent from one place in the interface to another in terms of standard design features and terminology (1)

Etc (1)

Compare and contrast the four types of interface evaluation. (14 marks)

*These techniques vary in terms of the degree of formality and the amount of user involvement. Heuristic evaluation involves assessing the interface based on a checklist of design principles (2). This assessment is usually performed by team members, who independently assess the interface and then compare their assessments (1). Weaknesses that are common in all the evaluations then point to areas that need modification (1). Users are not involved in this process. **[Turn over]***

In a walkthrough evaluation, the users see the interface at a meeting presentation (1), and they are "walked-through" the parts of the interface (1). The interactive evaluation can be used when the prototype has been created as an HTML or language prototype (1). The users can actually interact with the interface as if they were using the system (1), and can give direct comments and feedback based on their experience (1). Problems or areas of confusion can be noted and corrected by the team (1). Formal usability testing has the users interacting with the interface without guidance from the project team (2). Every move made by the user is recorded and then analyzed later in order to improve the interface (2).

(Total 20 marks)

Answer B13.

Assesses L06

Where does the analyst find the information needed to create a structure chart?
(4 marks)

One recommendation for creating a structure chart is to begin with the processes depicted on the logical DFD. Each process on a DFD tends to represent one module on the structure chart, and if leveled DFDs are used, then each DFD level tends to correspond to a different level of the structure chart hierarchy. This will be the approach the student is most likely to be familiar with – more complex approaches (e.g. the Yourdon approach) are not discussed in the course text but if a student shows familiarity with them they ought to be given credit.

What is meant by the characteristic of module cohesion? What is its role in structure chart quality?
(8 marks)

Module cohesion refers to how well the lines of code within each structure relate to each other. Ideally, each module should perform one task only which results in smaller, less complex modules that are easier to perfect and maintain, thus contributing to the overall quality of the structure chart (2).

Functional cohesion is the “best” situation, in which a module performs one and only one problem-related task (1). Sequential cohesion involves a module performing more than one task, and the output from one task is used by the next task in the module (1). In communicational cohesion, two or more tasks are combined in a module because both tasks require the same input elements (1). In procedural cohesion, a module incorporates several tasks that are unrelated (1). In temporal cohesion, several non-related tasks are combined in a module because they are performed at the same time (1). Logical cohesion combines several different tasks, and the one to be performed will be chosen by the control module and communicated through a control message passed to the subordinate module. Coincidental cohesion incorporates a number of non-related tasks that have no apparent relationship. This kind of cohesion is the poorest (1).

What is meant by the characteristic of module coupling? What is its role in structure chart quality? (8 marks)

Module coupling refers to how closely modules are interrelated. Ideally, modules are loosely coupled, which means that the design is characterized by a minimal number of interactions (e.g. data passing) between modules. Modules that are loosely coupled can be considered to be fairly independent and the interactions between them relatively easier to track and maintain, thus contributing to the overall quality of the structure chart (2)

Data coupling refers to the situation in which modules pass fields of data in messages. All data that is passed is used by the receiving module (1). Star coupling involves modules passing entire record structures(1). In this case, an entire record will be passed even if only a few fields are needed from the record. Control coupling refers to situations in which a module passes control information to a subordinate module (1). The subordinate modules use the control information to determine the correct processing to perform. Common coupling involves many modules referring to (and changing) the same global data area. This is hard to detect on a structure chart. Content coupling involves one module referring to the inside of another module (1). Data coupling is considered “good” coupling because modules pass parameters or specific pieces of data to each other. This is good because the interaction between the modules is very limited (1). Content coupling is considered “bad” coupling, because one module actually refers to and makes changes to information inside another modules. This is bad because the modules will be highly interactive with each other, and will be much more difficult to maintain in the future (1).

(Total 20 marks)

Question versus LO Grid

A Questions	LO1	LO2	LO3	LO4	LO5	LO6
1	Ac1.4					
2	Ac1.1					
3		Ac2.1				
4						Ac6.3
5		Ac2.4				
6			Ac3.1			
7					Ac5.3	
8				Ac4.1		
B Questions						
9	Ac1.4					
10					Ac5.2-4	Ac6.2-4
11		Ac1.1-2				
12				Ac4.1 & 4.3		
13					√	AC6.2-3