**Chapter 1 Review Questions**

1. Phases, steps, techniques, and deliverables are all components of what is known as the System Development Life Cycle (SDLC). Throughout the SDLC, four phases (planning, analysis, design, and implementation) are completed. These phases help determine why and how a system needs to be built, who will use it and for what, how it will operate (infrastructure), and actually building the system. Each phase is completed through a series of steps that more explicitly define the actions to be taken by each team member throughout the build process. Team members use various techniques (methods) to help complete each step of the process and produce deliverables, documents that help explain the system and its use.

**2. Describe the major phases in systems Development Life Cycle (SDLC).**

***Planning***- The planning phases is the fundamental process of understanding why and information system should be built and determining how the project team will go about building it.

***Analysis*** – The analysis phase answers the question of who will use the system, what the system will do and where and when it will be used. During this phase, the project team investigates any current systems, identifies improvement opportunities, and develops a concept for the new system.

***Design*** – The design phase decides how the system will operate, in terms of the hardware, software, and network infrastructure; the user interface, forms and reports; and specific programs, databases, and files that will be needed. The steps in the design phase determine exactly how the system will operate.

***Implementation*** - The implementation phase is where the system is actually built or purchased. Usually this phase gets the most attention, because for most systems it is the longest and most expensive single part of the development process.

Answer on page 3-6.

**3. Describe the principal steps in the planning phase. What are the major deliverables?**

The planning phase has two steps. The first one takes place during project initiation. This is where the system’s business value to the organization is identified. Most ideas for a new system come in the form of a system request. The IS department then works with the department who generated the request to conduct a feasibility analysis. This analysis examines whether the system can be built, will it provide business value, and will it be used if it is built. The system request and the analysis are then presented to an information systems approval committee, which will decide if they should take on the project. Once the project is approved, the second step can take place. The second step is project management. During this step the project manager creates a work plan, staffs the project, and puts techniques into place to help the project team control and direct the project through the entire SDLC. The deliverable for project management is a project plan, which describes how the project team will go about developing the system.

**4. Describe the principle steps in the analysis phase. What are the major deliverables?**

- ONE – **develop an Analysis Strategy** to guide the project team’s efforts. Such a strategy usually includes an analysis of the current system (as-is system) and its problems, and then ways to design a new system (to- be system).

- TWO – **Gathering requirements** through interviews or questionnaires. Using this information leads to the development of a concept for a new system. This concept is used as a basis to develop a set of business analysis models, which describe how the business will operate if the new system is developed

- THREE – **Combination of the Analyses, system concept, and models** are combined into a document called the *system proposal*, which is presented to the project sponsor and other key decision makers.

**5. The principle steps in the Design Phase are:**

a. The design strategy is first developed. It clarifies whether the system will be developed by the company’s own programmers, whether the system will be outsourced to another firm (usually a consulting firm), or whether the company will buy an existing software package.

b. This leads to the development of the basic architecture design for the system, which describes the hardware, software, and network infrastructure to be used. In most cases, the system will add or change the infrastructure that already exists in the organization. The interface design specifies how the users will move through the system (i.e. – navigation methods such as menus and on-screen buttons) and the forms and reports that the system will use.

c. The database and file specifications are developed. These define exactly what data will be stored and where they will be stored.

d. The analyst team develops the program design, which defines the programs that need to be written and exactly what each program will do.

The major deliverables for the Design Phase are:

1. Architecture Design
2. Interface Design
3. Database and File Specifications
4. Program Design

**6. The principal steps in the implementation phase include:**

1. *System Construction*, in which the system is actually built and tested to ensure it performs as designed. Most critical step in implementation.

2. *Installation,* in which the old system is turned off, and the new one’s turned on. This can be done using direct cutover, parallel conversion, or phased conversion approaches*.* A training plan must also be developed to teach users how to use the system and help manage any changes caused by the new system.

3. A *Support Plan* is established which includes a post-implementation review and a way to identify and changes needed for the system.

The major deliverables include the *training plan,* *support plan,* and the system itself.

**7. What are the roles of a project sponsor and the approval committee?**

During the project initiation step of the planning phase in SDLC, the IS department works together with the person or department that generated the system request (called the project sponsor) to conduct a feasibility analysis.

Then the system request and feasibility analysis are presented to an information systems approval committee (sometimes called a steering committee), which decides whether the project should be undertaken.

Answer on page 4

**8. What does gradual refinement mean in the context of SDLC?**

The deliverables of the systems development life cycle that are produced in the analysis phase provide a basic idea of the shape of the new system. These deliverables are used as input to the design phase, which then refines them to produce a set of deliverables that describes in much more detailed terms exactly how the system will be built. These deliverables are then used in the implementation phase to produce the actual system. So each phase refines and elaborates on the work done previously, the system is gradually refined.

**9. Compare and contrast process – centered methodologies with data – centered methodologies.**

- **Process Centered** emphasizes process models as the core of the system concept (Focuses on defining the processes of doing something).

- **Data Centered** emphasizes data models as the core of the system concept (focuses on the contents of the project and how they are organized)

- These methodologies are often balanced depending on the project style and complexity

**10. Structured design-based methodologies** adopt a formal step-by-step approach to the SDLC that moves logically from one phase to the next. Two examples of this type of methodology are the Waterfall Development (the analysts and users proceed in sequence from one phase to the next) and Parallel Development (attempts to address the problem of long delays between the analysis phase and the delivery of the system). **RAD-based methodologies** attempt to address both weaknesses of structured design methodologies by adjusting the SDLC phases to get some part of the system developed quickly and into the hands of the users. In this way, the users can better understand the system and suggest revisions that bring the system closer to what is needed. Three examples of RAD are Phased Development (breaks an overall system into a series of versions, which are developed sequentially), Prototyping (performs the analysis, design, and implementation phases concurrently), and Throwaway Prototyping (similar to Prototyping but is done at a different point in the SDLC; design prototype is built).

1. **Extreme Programming (XP) and Throw-Away Prototyping (TAP) are both development methodologies.** While both focus on getting a system developed quickly and into the hands of users, *XP* is an *agile development methodology* that focuses on continuous coding and integrative testing. It also focuses on keeping code as simple as possible and embracing change. Interacting with users is key to success with developing a system using *XP.* *TAP* is a *Rapid Application development methodology* that involves building design prototypes. These designs only represent a part of the system and only address certain issues. Several prototypes are developed through the design and analysis phases that help understand various important issues and once the system begins implementation, the prototypes are thrown away and not used in the real system.

**12. Describe the major elements and issues with waterfall development.**

With waterfall development-based methodologies, the analysts and users proceed in sequence from one phase to the next. Steps are Planning, Analysis, Design, Implementation, and System. The two key advantages of the structured design waterfall approach are that it identifies system requirements long before programming begins and it minimizes changes to the requirements as the project proceeds. The two key disadvantages are that the design must be completely specified before programming begins and that a long time elapses between the completion of the system proposal in the analysis phase and the delivery of the system (usually many months or years).

Answer on page 8

**13. Describe the major elements and issues with parallel development.**

Parallel development tries to address the problems of long delays between the analysis phase and the delivery of the system. It performs a general design for the whole system and the divides the project into a series of subprojects that can be designed and implemented in parallel. Once all of the subprojects are complete, there is a final integration of all the pieces to create a whole system.

There are a few issues with parallel development. One, there are problems caused by paper documents. Two, sometimes subprojects aren’t always completely independent. Three, decisions made in one subproject may affect another causing the end project to need significant integration efforts.

**14.**  Describe the major elements and issues with **Phased Development.**

**-** Phased Development breaks an overall system into a series of *versions*, which are developed in sequence. The analysis phase identifies the overall system concept, the team, users, and the system sponsor. Then categorizes the requirements into a series of versions. The most important version is the first because it contains the most important and fundamental requirements. The project then transfers into the Design and Implementation Phase. Once the first version is complete the next version starts at the analysis phase and repeats until the system is complete or no longer in use.

**Advantages –** quickly getting the system in the hands of the user.

**Disadvantages –** users begin to work with systems that are intentionally incomplete.

1. The major elements with **Prototyping** are that the analysis, design, and implementation phases are performed concurrently, and all 3 phases are performed repeatedly in a cycle until the system is completed. The basics of analysis and design are performed, and work immediately begins on a system prototype (a “quick-and-dirty” program that provides a minimal amount of features). The first prototype is usually the first part of the system that is used. This is shown to the users and project sponsor, who provide comments. The comments are used to reanalyze, redesign, and reimplement a second prototype, which contains a few additional features. The process continues until all agree that the prototype is functional enough to be installed and used within the organization. Refinement continues until it’s accepted as the new system. The issue with prototyping is that its fast-paced system releases challenge attempts to conduct careful, methodical analysis. Initial design decisions tend to become poor decisions because of all the refinement.
2. **The main elements of Throw-Away Prototyping** include analyzing, designing, and building prototypes that try to address and solve various issues that users face. These prototypes are working systems. They are just products that represents a part of the system that needs refinement. Several prototypes are developed to address the issues and reduce the risk associated with the system. Prototypes are thrown away once all the issues have been addressed and the system is ready to be implemented.

One major issue is that money is spent on prototypes that are not actually built as the system. It is difficult to keep track of each prototype if there are several built. The length of development time is longer than normal prototyping.

**17. What are the key factors in selecting a methodology?**

The key factors in selecting a methodology are: 1) Clarity of user requirements 2) Familiarity of technology 3) System complexity 4) System Reliability 5) Short time Schedule 6) Schedule Visibility

Answers on pg 15-17

**18. What is a use case?**

A use case describes how the user interacts with the system to perform some activity, such as placing an order, making a reservation, or searching for information. They are used to identify and to communicate the requirements for the system to the programmers who much write the system.

**19.**  **What is meant by use–case driven?**

- Use cases are primary modeling tools defining the behavior of the system. The use case describes how the user interacts with the system to perform some activity; also, used to identify and to communicate the requirements for the system to the programmers who must write the system.

**20.** The **Unified Modeling Language** is a standard set of diagramming techniques that provide a graphical representation rich enough to model any systems development project from analysis through implementation.

1. The Object Management Group is the consortium that develops/accepts various technology standards. In November 1997, OMG formally accepted UML as the standard for all object developers.

**22. What is the primary purpose of structure diagrams? Give some examples of structure diagrams.**

Structure diagrams provide a way to represent the data and static relationships in an information system. Some examples of structure diagrams are Class, Object, Package, Deployment, Component, and Composite Structure.

This chart is on page 30 if you want to find the descriptions.

**23. For what are behavior diagrams used? Give some examples of behavior diagrams.**

Behavior diagrams provide the analyst with a way to depict the dynamic relationships among the instances or objects that represent the business information system. They also allow the modeling of the dynamic behavior of individual objects throughout their lifetime. They support the analyst in modeling the functional requirements of an evolving information system.

* Activity – illustrate business workflows independent of classes, the flow of activities in a use case, or detailed design of a method
* Communication – model the behavior of objects within a use case. Focuses on the communication among a set of collaborating objects of an activity
* Behavioral state machine – examine the behavior of one class

**24. Why is it important for an Object Oriented Systems analysis and design (OOSAD) approach to be incremental and iterative?**

- It is important because it enables analysts to break a complex system into smaller more manageable modules, work on the modules individually, and easily piece the modules back together to form an information system. This makes system development easier to grasp, share among members of a project team, and communicate to users.

**25.** If an **OOSAD (Object-Oriented Systems Analysis and Design)** approach is **Incremental and Iterative** it means that the systems analysts develop their understanding of a user’s problem by building up the 3 architectural views little by little. The systems analyst does this by working with the user to create a functional representation of the system under study. Next, the analyst attempts to build a structural representation of the evolving system. Using the structural representation of the system, the analyst distributes the functionality of the system over the evolving structure to create a behavioral representation of the evolving system.

1. **The phases and Workflows of the Unified Process include:**

Phases: Inception, Elaboration, Construction, Transition

Workflows:

-Engineering workflows include 1. Business Modeling, 2. Requirements, 3. Analysis, 4. Design, 5. Implementation, 6. Testing, and 7. Deployment

-Supporting workflows include 1. Project Management, 2. Configuration and Change Management, and 3. Environment

**27. Compare the phases of the Unified process with the phases of the waterfall model.**

The first phase of the Unifies process, Inception, is very similar to the planning phase of a traditional SDLC approach. Planning is the first phase on the waterfall model. The next phase in the Unified process is Elaboration where analysis and design workflows are the primary focus. This is also similar to the waterfall model because analysis and design are the next steps. Then Construction is the next phase in the Unified process which is similar to the Implementation phase of the waterfall model. Last is the Transition phase of the Unified process which tests the system which may result in havening to go back and make changes which is not like the waterfall model, where it doesn’t go back to any phases.

**28. What are the major roles on the project team?**

Project teams will consist of analysts, a project manager, programmers, technical writers and other specialist.

**29. Compare and contrast the role of a systems analyst, business analyst, and infrastructure analyst.**

- **Systems analyst** focuses on the technical IS issues surrounding the system such as developing ideas and suggestions for how information technology can improve business processes, designs the new business processes with help from the business analyst, designs the new IS, and ensures that all IS standards are maintained.

- **Business analyst** – focuses on the business issues surrounding the system. These include identifying the business value that the system will create, developing new ideas and suggestions for how the business processes can be improved, and designing the new processes and policies in conjunction with the systems analyst

- **Infrastructure analyst** focuses on the technical issues surrounding how the system will interact with the organization’s technical infrastructure. The tasks include ensuring that the new information system conforms to organizational standards and identifying infrastructure changes needed to support the system.

**30.** The **most important phase in the SDLC** is the Planning Phase because it is the process of understanding why an information system should be built and determines how the project team should go about building it. If careful planning isn’t done in the early stages of the project it’s going to be difficult to coordinate activities and manage project risks effectively. Also, without a good plan, time may be wasted in areas that do not need a lot of attention and a poor design may be implemented. The plan is the foundation for the rest of the project so it must include thorough details in order to move through the additional phases effectively.

1. **The major elements and issues with an object oriented approach to developing Information Systems are as follows:**

Elements: These methodologies use RAD-based sequence of SDLC phases but attempt to balance the emphasis between process and data by focusing the decomposition of problems on objects that contain both data and processes.

The approach must be:

1. Use-Case Driven- Modeling tools (use-cases) that describe how the user interacts with the system to perform some activity
2. Architecture Centric- The underlying software architecture of the evolving system drives the specification, construction, and documentation of the system.
3. Iterative and Incremental- System undergoes continuous testing and refinement throughout the life of the project. Analysts develop understanding by building three views (structural, functional, behavioral).

Issues: There are more pieces of the project to keep track of. Continuous testing is costly and time consuming. Missing requirements may not be found until late in the development process. Since users can better understand the IT in use, the requirements for a project can expand very quickly.