Ryan Somma

The following assignments for Week Two are taken from the Review Questions, Problems and Exercises sections of your textbook.

Chapter Three, page 114:

1. *The Capability Maturity Model (CMM) was developed by the Software Engineering Institute at Carnegie Mellon, and is widely used by both the private and public sectors. What is the purpose of the CMM framework and how does it achieve this ?*

The CMM aims to help organizations reduce costs and project timelines, while improving productivity and quality. It provides five maturity levels for organizations to work towards, with the first level having no consistent standards or methodologies, the fifth level representing optimal systems development, and the levels in between establishing employee experience, standard operating procedures, and metrics gathering on the process.

2. Systems development methodology *and* system life cycle *are two terms that are frequently used and just as frequently misused. What is the difference between the two terms?*

A system life cycle happens in all projects, and it involves the development, deployment, and ultimate obsolescence and refactoring of the project. A system development methodology is an established, standardized process for developing systems. It allows for repeatability and documentation to easily integrate new users into the process.

3. *A number of underlying principles are common to all systems development methodologies. Identify these underlying principles and explain them.*

Principle 1, Get the System Users Involved: No matter how well the user requirements are written out, the users will find new issues in any system developed. To avoid these unanticipated issues, user input must be gathered throughout the development process.

Principle 2, Take a Problem-Solving Approach: define the problem and determine what any solution must achieve to solve the problem.

Principle 3, Establish Phases and Activities: Break the project down into phases where different stakeholders are involved and different goals and deliverables achieved.

Principle 4, Document, Document, Document: Documented user requirements provide a contract to refer to when new requirements arise mid-process. Documented programming code explains why a programmer developed things a particular way. Documented processes establish standards and provide a guide to new employees.

Principle 5, Establish Standards: standard database, software, and interfaces must be decided upon.

Principle 6, Manage: Process and projects require oversight to ensure quality standards are met

Principle 7: Information Systems as Capital Investments: IS must be considered a business asset just like an office building, personnel, or other equipment, and should be managed for cost effectiveness.

Principle 8, Be Willing to Kill the Project or Redefine It: regularly evaluate the project to make sure it is still feasible. If not, consider all project costs lost and kill or redefine the project scope.

Principle 9, Divide and Conquer: The project is a solution to a problem, but the project itself is a problem requiring solution. Take the large problem of the project, and break it down into smaller manageable problems.

Principle 10, Design for Growth and Change: System entropy is inevitable as technology and business requirements change. Any system should be developed to anticipate such change.

4. *The PIECES framework was developed by James Wetherbe as a means to classify problems. Identify the categories, then categorize the following problems using the PIECES framework:*
    a. *Duplicate data is stored throughout the system.*
    b. *There is a need to port an existing application to PDA devices.*
    c. *Quarterly sales reports need to be generated automatically.*
    d. *Employees can gain access to confidential portions of the personnel system.*
    e. *User interfaces for the inventory system are difficult and confusing, resulting in a high frequency of incorrect orders.*

PIECES stands for Performance Information Economics Control Efficiency Service

a) Is a problem of Information. Duplicate information means information must be updated in multiple places, and the potential for such information to get out of synch arises.
b) Service Problem. Porting the application to a new device is providing a new service to the user.
c) There are information and service elements to this, but it sounds primarily like a problem of Efficiency. Automatically generating the report will eliminate personnel hours, and ensure the report is ready on schedule for management.
d) Security. Usenames and passwords with assigned privileges must be instituted to prohibit employees from viewing their coworker's personal details.
e) Performance. A more intuitive UI would reduce errors and costs.

5. *Scope definition is the first phase of the* FAST *methodology and it is either the first phase or part of the first phase of most methodologies. What triggers the scope phase, which stakeholders are involved in this phase, what two essential questions need to be answered and what three important deliverables come out of this phase?*

A combination of problems, opportunities, and directives triggers the scope phase. System owners, project managers, and system analysts are involved in this phase. User input would be too detailed and system builders would lack a design to build off of. The scope phase must answer the questions of whether the problem is worth looking at, and, if so, determine the extent and boundaries of the project. A problem statement, a concise definition of the problem and project constraints, the project scope, which defines budget, timeline, resources, and project boundaries, and a statement of work, a contract to develop the system, are the three deliverables that must come out of this phase.

6. *What is the essential purpose of the logical design phase? How does it accomplish this? How are technological solutions incorporated in this phase? What are some common synonyms for this phase used by other methodologies? Who are the typical participants in this phase? What is agile modeling and what is its purpose? What are the deliverables coming out of this phase? In terms of the development team what critical transition takes place by the end of this phase?*

The logical design phase translates business requirements into models. This process is independent of technological solutions, and only maps the business processes in preparation for the following phases, which will begin to translate them into technical solutions. The term "system modeling" is also used to describe this phase. System Analysts, Users, and Project Managers are included in this phase to ensure a proper level of detail is achieved and standards are adhered to. Agile modeling reduces the amount of time spent on this phase, and methodologies such as Extreme Programming de-emphasize this phase under the assumption that, no matter how clearly defined, project requirements will change and the process should account for this. Data, Process, and interface models should be drawn out of this phase from the preceding phases. At the end of this phase, the project transfers from the system analysis to system design portion of the system development.

Chapter Four, page 151:

1. *Of the many different reasons that projects fail, what is the major cause of project failure?*

Project managers who are not trained or educated to be project managers.

2. *What are PERT and GANTT charts? How do we decide which one to use?*

The two charts communicate the project tasks and schedule. PERT charts communicate the relationships between tasks, while GANTT charts are better for visualizing a project schedule.

3. Why is negotiating scope important? What is the deliverable in the process of negotiating the scope?

A clearly-defined project scope is crucial to managing System Owner and System User expectations for the project, as well as allowing Project Managers and System Designers to properly plan out the system development. At the end of the project scope phase, a statement of work is delivered, which serves as a contract between the interested parties for what will be delivered in this system development, when, and for what cost.

4. What is a popular tool used to identify tasks in the project management life cycle?

A work breakdown structure takes an entire project and segments it into phases, with sub-tasks, deliverables and milestones.

5. What are the factors to consider in estimating task durations?

The skills and experience of the employees tackling the task, availability and involvement of the users, employees available for the task, and complexity of the task are some factors to consider in estimating task durations.

6. What are the differences between forward scheduling and reverse scheduling?

Forward scheduling begins with a project development start date and determines the project completion date based on the sum of all task durations. Reverse scheduling begins with a project completion date, and then determines all task durations based on meeting this deadline.

7. What are the categories of resources to be allocated to the project?

People (anyone involved in the project), services (such as quality review), facilities and equipment (workspace and technologies necessary for the project), supplies and materials (office supplies), and money to support all aspects of the project development.

8. What should project managers do to manage changes that occur and/or are requested during a project?

They should familiarize the System Owner and Project Manager with the Management Expectations Matrix, explaining how increasing the project scope will impact one or more aspect of the project development, and ask Management to define which aspects of the project, scope, quality, or cost, should be allowed adjustment.

9. Why is critical path analysis important?

Understanding the critical path for a project, the series of tasks that are interdependent and have no slack time, is crucial for establishing priorities for tasks in order to keep a project on schedule.