**Chapter 3 ALL Review Questions**

1. Why do many projects end up having unreasonable deadlines? How should a project manager react to unreasonable demands?

Some of the main reasons that unreasonable deadlines are set are due to:

-Upcoming Holidays

-Winning a project with a low (or lower than normal) bid

-Funding opportunity pressures where project managers are pushed to promise a complete system for a date that is long before they will actually be able to deliver it

In reaction, Project managers should first start with a realistic assessment of the work that will need to be done to complete the project. It is up to the manager to follow four crucial developmental steps (ID project size, create/manage workplan, staff the project, coordinate project activities) to ensure that appropriate deadlines are set…and hopefully met.

See middle two ¶ on page 70.

2. What are the trade-offs that project manager must manage?

The trade-offs are the size of the system (in terms of what it does), the time to complete the project (when the project will be finished), and the cost of the project. Kind of like Functionality, Time, and Money

**3. What are the two basic ways to estimate the size of a project?**

The two ways to estimate the size of a project are relying on past experiences or industry standards or by calculating the function points, a measure of program size based on the number and complexity of inputs, outputs, queries, files, and program interfaces.

**4.** *What is a function point and how is it used?*

A **function point** is a measure of program size based on the system’s number and complexity of inputs, outputs, queries, files, and program interfaces. This is used in the function point approach of identifying project size.

To calculate the function points for a project, components are listed on a worksheet to represent the major elements of the system (inputs, outputs, etc). After each line on the worksheet is filled in and the complexity is added in (low, medium, high complexity); a number of points is calculated per line by multiplying each number of components by a complexity index. The line totals are added to determine the total unadjusted function points for the project.

5. The 3 steps of the **Function Point Approach** are:

a. Estimate System Size – needs to be done using function points. A function point is a measure of program size based on the system’s number and complexity of inputs, outputs, queries, files, and program interfaces. To calculate the function points for a project, components are listed on a worksheet to represent the major elements of the system. The project manager records the total number of each component that the system will include, and then he or she breaks down the number to show the number of components that have low, medium, and high complexity. After each line is filled in, a total number of points’ is calculated per line by multiplying each number by a complexity index. The line totals are added to determine the total unadjusted fuction points (TUFP) for the project. Sometimes a shortcut can be used to determine the complexity of the project. An APC value can be assigned (ranges from .65 for very simple systems, to 1.00 for normal systems, to as much as 1.35 for complex systems); then they multiply the value to the TUFP score. Once the number of function points has been estimated you need to convert the number of function points into the lines of code that will be required to build the system. The choice of development language has a significant impact on the time and cost of projects.

b. Estimate Required Effort – effort is a function of the system size combined with production rates (how much work someone can complete in a given time). The *COCOMO model* is the most popular algorithm used to convert a lines-of-code estimate into a person-month estimate. There are different COCOMO models, which vary based on the complexity of the software, the size of the system, the experience of the developers, and the type of software being developed. For small-to moderate-sized business software projects (i.e. – 100,000 lines of code and 10 or fewer programmers), the model is:

effort (in person-months) = 1.4 \* thousands of line of code

c. Estimate Time Required – Historical data or estimation software may be used as aids. A rule of thumb equation used is:

schedule time (months) = 3.0 \* person-months^1/3

The equation suggests that a project that has an effort of 14 person-months should be scheduled to take a little more than 7 months to complete.

**6.** What is the formula for calculating the effort for a project?

Effort (in person-months) = 1.4 x thousands of lines of code

(i.e.- If a project requires 10,000 lines of code the effort required would be: 1.4 x 10 = 14 months)

The formula can be found on page 76.

7. Name two ways to identify the tasks that need to be accomplished over the course of a project?

One approach is to get a list of tasks that has already been developed and to modify it. There are standard lists of tasks, or methodologies, which are available for use as a starting point.

Another approach used if the project manager prefers to begin from scratch, is to use a structured, top-down approach whereby high-level tasks are first defined and then broken down into sub-tasks.

**8. What is the difference between a methodology and a work plan? How are the two terms related?**

A work plan is a dynamic schedule that records and keeps track of all the tasks that need to be accomplished over the course of the project. It lists each task, along with important information about it, such as when it needs to be completed, the person assigned to do the work, and any deliverables that will result. A methodology is a formalized approach to implementing an SDLC. They can be purchased from consultants, or venders, or books that can serve as a guide or the project manager can create one. The two terms are related because they both break down the tasks that need to be implemented. A methodologies breakdown of tasks can serve as the backbone for the project work plan.

**9.** *Compare and contrast the Gantt chart with the PERT chart.*

A **Gantt chart** is a horizontal bar chart that shows the same task info as the project work plan but in a graphical way. This can be made with a spreadsheet package, graphic software (Visio), or project management package.

First tasks are listed in rows and the time is listed in columns by increments based on the needs of the project. As the tasks are finished, the appropriate cells are filled in showing when the task started and when it was finished.

10. The **Hurricane Model** is based around the same way forecasters watch behaviors of storms and predict when they will hit and the amount of damage they could cause. The predictions become more accurate the closer the storm gets and then finally arrive. In planning, when a system is first requested, the project sponsor and project manager attempt to predict how long the SDLC will take, how much it will cost, and what it will ultimately do when it is delivered (i.e. – its functionality). However, the estimates are based on very little knowledge of the system. As the system moves into the analysis, more information is gathered, the system concept is developed, and the estimates become even more accurate and precise. As the system moves closer to completion, the accuracy and precision increase, until the final system is delivered.

1. What is scope creep, and how can it be managed?

Scope creep is the most common reason for schedule and cost overruns in a project. It occurs when new requirements are added to a project after the original requirements have been established and “frozen”.

This can be best managed by:

-Identifying the requirements as well as possible in the beginning of the project so that there are no “new requirements” that can be added

- Apply analysis techniques effectively

-Use a combination of meetings with users and prototyping so that any unrecognized requirements can be discovered by users and implemented into new prototypes

-Project managers should only allow absolutely necessary requirements to be added after a project begins

- Use Timeboxing techniques (described in question 12)

See pp. 83-85

12. What is time boxing, and why is it used?

It is another approach to scope management that sets a fixed deadline for a project and delivers the system by that deadline no matter what, even if functionality needs to be reduced. Time boxing ensures that project teams don’t get hung up on the final finishing touches that can drag out indefinitely, and it satisfies the business by providing a product within a relatively fast time frame.

**13. What are the problems associated with the conventional WBSs?**

1. They tend to be focused on the design of the information system being developed. The WBS can sometimes force the premature decomposition of the system design and the tasks associated with creating the design of the system.
2. They tend to force too many levels of detail very early on in the SDLC for large projects or they tend to allow too few levels of detail for small projects. This causes the cost and schedule estimation for many information systems development projects tends to be inaccurate.
3. Because they are project specific, they are very difficult to compare across projects. This leads to ineffective learning across the organization. It allows project managers to make the same mistakes twice, not being able to learn from previous projects managed by others.

**14.** *What is an evolutionary WBS? How does it address the problems associated with a conventional WBS?*

Evolutionary Work Breakdown Structure is an incremental and iterative process for project planning that can be used with object oriented systems development. Conventional WBS have 3 underlying problems: **1)** they tend to be focused on the design of the information system being developed **2)** they tend to force too many levels of details very early on in the SDLC for large projects or they tend to allow too few levels of detail for small projects **3)** because they are project specific, they are very difficult to compare across projects

This addresses these problems by allowing the development of the iterative work plan. **First**, evolutionary WBS are organized in a standard manner across all projects: by workflows, phases, and tasks.

**Second,** evolutionary WBSs are created in an incremental and iterative manner

**Third,** because the structure of an evolutionary WBS is not tied to any specific project, evolutionary WBSs enable the comparison of the current project to earlier projects and this supports learning from the past successes and failures.

15. An **Iterative Workplan** is where evolutionary Work Breakdown Structures (WBSs) are organized in a standard manner across all projects: by workflows, phases, and then tasks. This decouples the structure of an evolutionary WBS from the structure of the design of the product. This prevents prematurely committing to a specific architecture of a new system. Second, evolutionary WBSs are created in an incremental and iterative manner. The first evolutionary WBS is typically only done for the aspects of the project understood by the analyst. Later on, as the analyst understands more about the evolving development process, more details are added to the WBS. This encourages a more realistic view of both cost and schedule estimation. Third, because the structure of an evolutionary WBS is not tied to any specific project, evolutionary WBSs enable the comparison of the current project to earlier projects. This supports learning from past successes and failures.

1. Describe the differences between a technical lead and a functional lead. How are they similar?

A technical lead usually oversees the progress of a group of programmers and more technical staff members, while a functional lead is usually assigned to oversee a group of analysts.

These two are similar in that they both directly report to the project manager and they both oversee various groups of employees to ensure that projects are being completed on-time. They are both more hands on with those who are directly involved with developing the project.

See pg. 92

17. Describe three technical skills and three interpersonal skills that would be very important to have on any project?

Technical skills: How to use the internet, how to use office products, knowing how an operating system works, some programming experience, experience with database. (These weren’t listed in book, but these are what I think, this question is more opinionated)

Interpersonal skills: Good communication skill, active listening, good leadership skills

**19.** *List three techniques to reduce conflict.*

**1)** Clearly define plans for the project

**2)** Develop a charter

**3)** Develop Schedule commitments ahead of time

**4)** Develop a detailed operating procedures and communicate between members

20. **The difference between upper CASE and lower CASE (Computer-Aided Software Engineering)** is **upper CASE** software packages are used primarily during the analysis phase to create integrated diagrams of the system and to store information regarding the system components. **Lower CASE** software packages are design-phase tools that create the diagrams and then generate code for database tables and system functionality.

1. Describe three types of standards and provide examples of each.
2. Documentation involves ensuring that all documents involved in a project look and feel similar in format and design. Another key is that aspect is document storage.

i.e. - Having the date and name of the project printed in the top right corner of the first page of all documents.

1. Coding can involve various syntax and format rules for all code submitted for the project.

i.e. – Using indentation for all if-else statements and case statements, or having a comment for every X lines of code

1. UI Design involves using format rules and certain text formats for key UI features.

i.e. – Having all labels boldfaced and centered, or Formatting the tab order to move from top left to bottom right.

See pg. 97

22. What belongs in the project binder? How is the project binder organized?

Project binders contain all the deliverables and all the internal communication that takes place-the history of the project. Organize the binder with dividers that separate the content according to the major phases of the project. An additional divider should contain internal communication, such as the minutes from status meetings, written standards, letters to and from the business users, and a dictionary of relevant business terms. Binder should also have a table of contents.

**23. Create a list of potential risks that could affect the outcome of a project.**

* Weak personnel
* Scope creep
* Poor design
* Overly optimistic estimates

**24.** *Some companies hire consulting firms to develop the initial project plans and manage the project but use their own analysis and programmers to develop the system. Why do you think some companies do this?*

I believe that a company would do this if they are inexperienced in this particular project or maybe just a young organization. Being supervised by an experienced firm would give them confidence while they were working on the project and gaining experience. The planning of the project is often the most difficult and time consuming part of the job, so the companies may feel better in their programming skills than their developing skills. Also, by having another firm manage the project will benefit them because the firm may not manage the same way that the company is used to. This will give the team members a new perspective on how to accomplish goals. Lastly, a company would do this because it could possibly be cheaper to have another firm do the planning for them because it would cost more money in labor in time to do it themselves.