FEDERAL UNIVERSITY OF KASHERE
FACULTY OF SCIENCE
Department of Computer Science
www.fukashere.edu.ng

P.M.B 0182,
GOMBE,
GOMBE STATE

SYSTEM ANALYSIS AND DESIGN (CSC 3307)

MODULE (2)

Module Content:

- **Project Management**
  - Project Identification
    - *System Request*
  - Feasibility Analysis
    - *Technical Feasibility*
    - *Economic Feasibility*
    - *Organizational Feasibility*
  - Project Selection
  - Traditional Project Management Tools
    - *A Work-Breakdown Structure*
    - *Gantt Chart*
    - *Network Diagram*
  - Project Effort Estimation
  - Creating and Managing the Workplan
    - *Evolutionary Work Breakdown Structure and Iterative Workplans*
    - *Managing Scope*
    - *Timeboxing*
    - *Refining Estimate*
    - *Managing Risk*
  - Staffing the Project
    - *Staffing Plan*
    - *Motivation*
    - *Handling Conflict*
  - Environment and Infrastructure Management
    - *CASE Tools*
    - *Standards*
    - *Documentation*

Referral material: *Systems Analysis and Design-An Object-Oriented Approach with UML-2015.*

This module primarily describes the project management workflow of the Unified Process.

- The first step in the process is to identify a project that will deliver value to the business and to create a system request that provides basic information about the proposed system.
- Second, the analysts perform a feasibility analysis to determine the technical, economic, and organizational feasibility of the system; if appropriate, the system is selected and the development project begins.

- Third, the project manager estimates the functionality of the project and identifies the tasks that need to be performed.
- Fourth, the manager staffs the project.
- Finally, the manager identifies the tools, standards, and process to be used; identifies opportunities for reuse; determines how the current project fits into the portfolio of projects currently under development; and identifies opportunities to update the overall structure of the firm's portfolio of systems current in use.

INTRODUCTION

Most projects occurring in people's lives, such as weddings or graduation celebrations, require planning and management. Months are spent in advance identifying and performing all the tasks that need to get done, such as sending out invitations and selecting a menu, and time and money are carefully allocated among them. In the end, the success of any party has a lot to do with the effort that went into planning along the way. System development projects can be much more complicated than the projects we encounter in our personal lives usually, more people are involved (e.g., the organization), the costs are higher, and more tasks need to be completed.

Owing to the complexity of software and software development, it is virtually impossible to "know" all of the possible things that could happen during system development projects. Therefore, it is not surprising that "party planners" exist for information systems projects: They are called **project managers.**

PROJECT MANAGEMENT

Is the process of planning and controlling the development of a system within a specified time frame at a minimum cost with the right functionality. In general, a **project** is a set of activities with a starting point and an ending point meant to create a system that brings value to the business. **A project manager** has the primary responsibility for managing the hundreds of tasks and roles that need to be carefully coordinated.

During the inception phase of the Unified Process (*based on the enlargement and refinement of a system through multiple iterations, with cyclic feedback*) of a new systems development project, someone a manager, staff member, sales representative, or systems analyst—typically identifies some **business value** that can be gained from using information technology. New systems development projects should start from a **business need or opportunity**.

Ultimately, information systems need to affect the organization's bottom line (in a positive way!). To ensure that a real business need is being addressed, the affected business organization (called the **project sponsor),** proposes the new systems development project using a **system request**. **The system request effectively kicks off the inception phase for the new systems development project**. The request is forwarded to an **approval committee** for consideration. The approval committee reviews the request and makes an initial determination of whether to investigate the proposal or not. If the committee initially approves the request, the systems development team gathers more information to determine the feasibility of the project.

F**easibility analysis:** plays an important role in deciding whether to proceed with an information systems development project. It examines the technical, economic, and organizational pros and cons of developing the system, and it gives the organization a slightly more detailed picture of the advantages of investing in the system as well as any obstacles that could arise. In most cases, the project sponsor works closely with the development team to develop the **feasibility analysis**. Once the feasibility analysis has been completed, it is submitted to the **approval committee**, along with a **revised system request**. The committee then decides whether to approve the project, decline the project, or table it until additional information is available. Projects are selected by **weighing risks** and **returns** and by making **trade-offs** at the organizational level.

Once the committee has approved a project, the development team must carefully plan for the actual development of the system. Because we are following a Unified Process-based approach, the systems development workplan will evolve throughout the development process. Given this evolutionary approach, one critical success factor for project management is to start with a realistic assessment of the work that needs to be accomplished and then manage the project according to that assessment. This can be achieved by carefully creating and managing the workplan, estimating the effort to develop the system, staffing the project, and coordinating project activities.

PROJECT IDENTIFICATION

A project is identified when someone in the organization identifies a **business need** to build a system. This could occur within a business unit or IT, come from a steering committee charged with identifying business opportunities, or evolve from a recommendation made by external consultants. Examples of business needs include supporting a new marketing campaign, reaching out to a new type of customer. Sometimes, needs arise from some kind of "pain" within the organization, such as a drop in market share.

The **project sponsor** is someone who recognizes the strong business need for a system and has an interest in seeing the system succeed. Examples, include from small departments in an organization to even the CEO themselves.

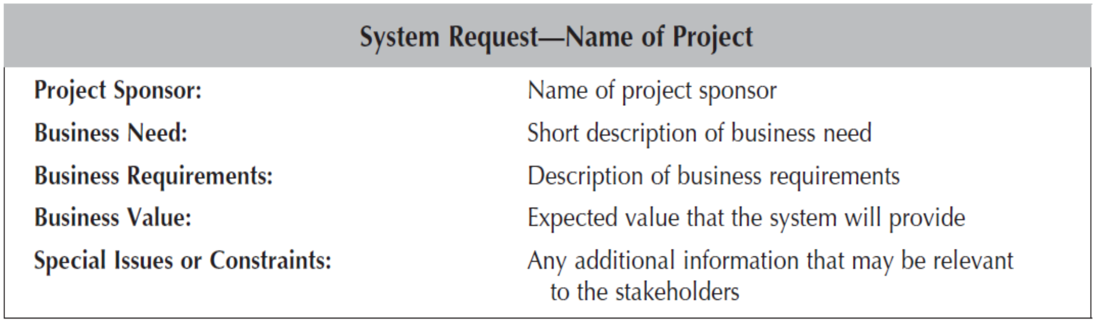
The **business need** drives the high-level **business requirements** for the system. **Requirements** are what the information system will do, or the **functionality** it will contain. They need to be explained at a high level so that the approval committee and, ultimately, the project team

The **project sponsor** also should have an idea of the **business value** to be gained from the system, both in tangible and intangible ways. **Tangible value** can be quantified and measured easily (e.g., 2 percent reduction in operating costs). An **intangible value** results from an intuitive belief that the system provides important, but hard-to-measure, benefits to the organization (e.g., improved customer service or a better competitive position). Once the project sponsor identifies a project that meets an important business need and he or she can identify the system's business requirements and value, it is time to formally initiate the project. In most organizations, project initiation begins with a document called a **system request**.

System Request

A system request is a document that describes the business **reasons** for building a system and the **value** that the system is expected to provide. Most system requests include five elements:

- **Project sponsor:** The sponsor describes the person who will serve as the primary contact for the project
- **Business need:** business need presents the reasons prompting the project
- **Business requirements:** The business requirements of the project refer to the business capabilities that the system will need to have.
- **Business value:** business value describes the benefits that the organization should expect from the system.
- **Special issues:** special issues are included on the document as a catch-all for other information that should be considered in assessing the project. For example, the project may need to be completed by a specific deadline. Project teams need to be aware of any special circumstances that could affect the outcome of the system. Figure 1 shows a template for a system request.



System Request—Name of Project	
Project Sponsor:	Name of project sponsor
Business Need:	Short description of business need
Business Requirements:	Description of business requirements
Business Value:	Expected value that the system will provide
Special Issues or Constraints:	Any additional information that may be relevant to the stakeholders

Figure 1: System Request Template

FEASIBILITY ANALYSIS

Feasibility analysis guides the organization in determining whether or not to proceed with a project. Feasibility analysis also identifies the important **risks** associated with the project that must be addressed if the project is approved. Most project teams will revise their feasibility study throughout the development process and revisit its contents at various checkpoints during the project. If at any point the **project's risks and limitations outweigh its benefits**, the project team may decide to **cancel the project or make necessary improvements.**

Each organization has its own process and format for the feasibility analysis, but most include three types:

1) Technical feasibility,
2) Economic feasibility, and
3) Organizational feasibility.

The results of these analyses are combined into a **feasibility study**, which is given to the approval committee.

1) Technical feasibility

The first type of feasibility analysis addresses the technical feasibility of the project: the extent to which the system can be successfully designed, developed, and installed by the IT group. Technical feasibility analysis is in essence **a technical risk analysis** that strives to answer this question: Can we build it?

2) Economic feasibility

The second element of a feasibility analysis is to perform an economic feasibility analysis (also called a **cost–benefit analysis**), which identifies the **financial risk** associated with the project. Economic feasibility is determined by identifying costs and benefits associated with the system, assigning values to them, and then calculating the cash flow and return on investment for the project. The more expensive the project, the more rigorous and detailed the analysis should be.

3) Organizational feasibility

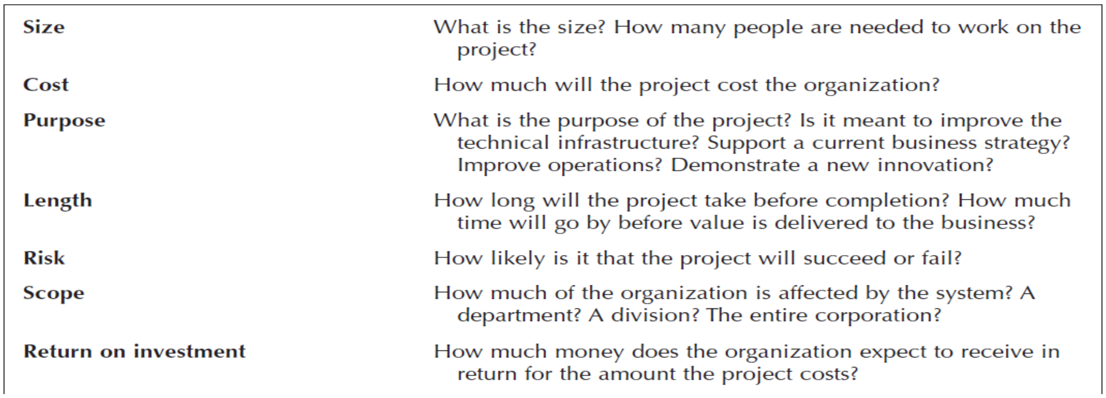
The final type of feasibility analysis is to assess the organizational feasibility of the system, how well the system ultimately will be accepted by its users and incorporated into the ongoing operations of the organization. In essence, an organizational feasibility analysis attempts to answer the question, If we build it, will they come? Ways to assess the organizational feasibility of the project include:

- To understand how well the goals of the project align with business objectives.
- To assess organizational feasibility is to conduct a stakeholder analysis. A stakeholder is a person, group, or organization that can affect (or will be affected by) a new system. i.e., system users, and organizational management.

PROJECT SELECTION

Once the **feasibility analysis** has been completed, it is submitted to the **approval committee**, along with a **revised system request**. The committee then decides whether to approve the project, decline the project, or table it until additional information is available. At the project level, the committee considers the value of the project by examining the **business need** (found in the system request) and the **risks of building the system** (presented in the feasibility analysis).

Before approving the project, however, the committee also considers the project from an organizational perspective; it has to keep in mind the company's entire **portfolio of projects**. This way of managing projects is called **portfolio management**. Portfolio management takes into consideration the different kinds of projects that exist in an organization—large and small, high risk and low risk, strategic and tactical. See Figure 2.



Size	What is the size? How many people are needed to work on the project?
Cost	How much will the project cost the organization?
Purpose	What is the purpose of the project? Is it meant to improve the technical infrastructure? Support a current business strategy? Improve operations? Demonstrate a new innovation?
Length	How long will the project take before completion? How much time will go by before value is delivered to the business?
Risk	How likely is it that the project will succeed or fail?
Scope	How much of the organization is affected by the system? A department? A division? The entire corporation?
Return on investment	How much money does the organization expect to receive in return for the amount the project costs?

Figure 2: Organizational project portfolio

The approval committee must be selective about where to allocate resources. This involves trade-offs in which the organization must give up something in return for something else to keep its portfolio well balanced.

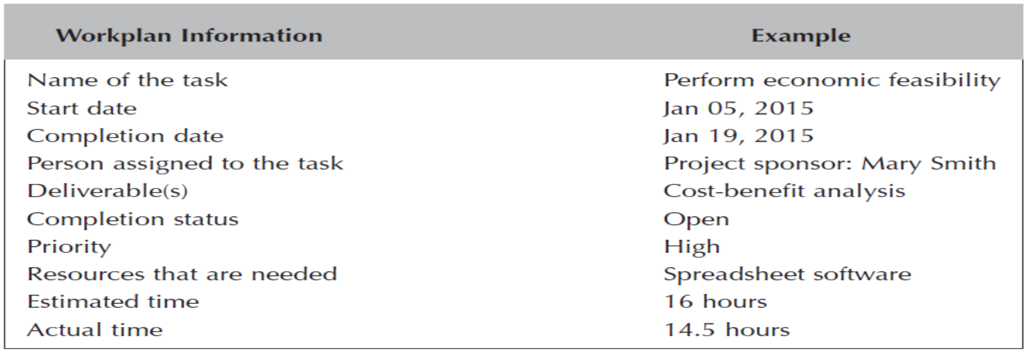
TRADITIONAL PROJECT MANAGEMENT TOOLS

Tools that have successfully been applied to **managing** project include the:

1) A work-breakdown structure,
2) A Gantt chart, and
3) A network diagram.

To understand these tools appropriately, we need to introduce the concept of a task.

A task: is a unit of work that will be performed by a member or members of the development team, such as feasibility analysis. Figure 3 present the components of a task.



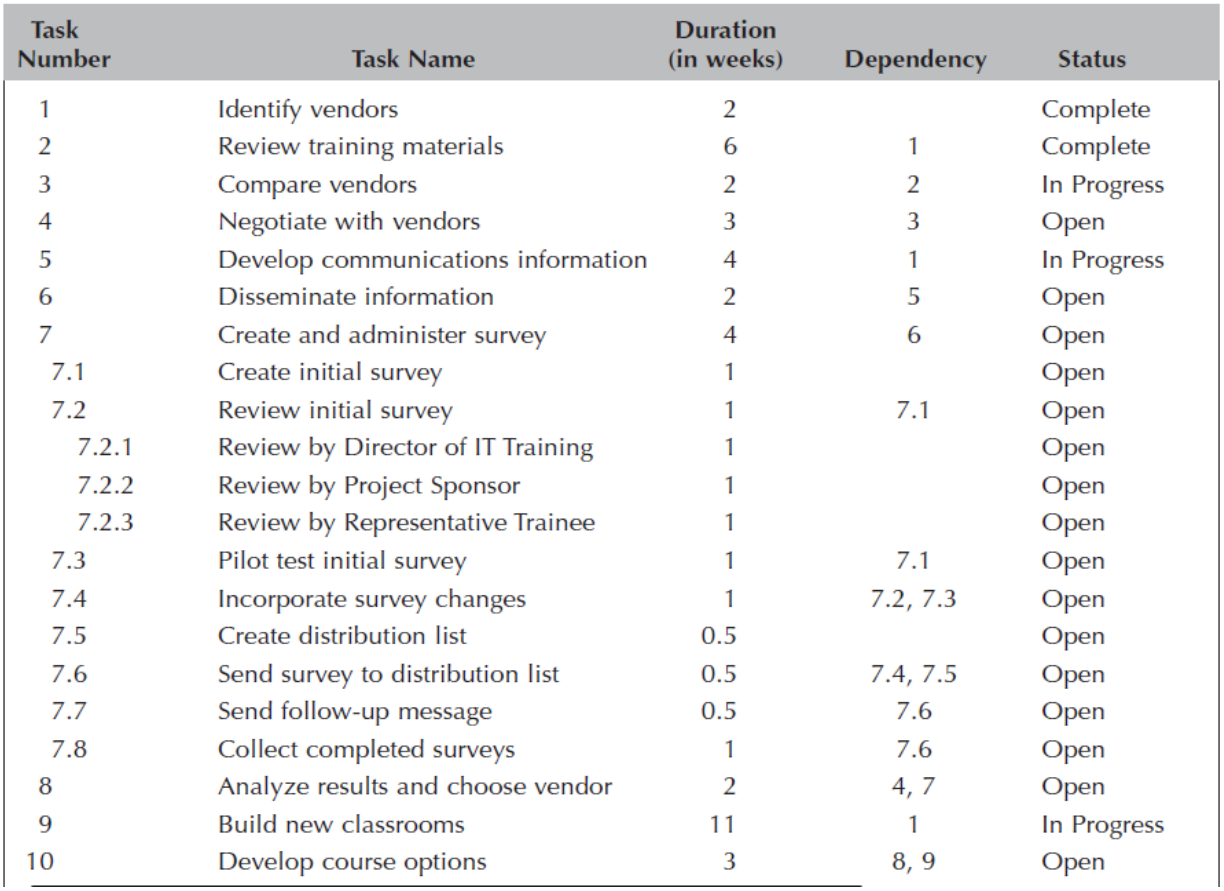
Workplan Information	Example
Name of the task	Perform economic feasibility
Start date	Jan 05, 2015
Completion date	Jan 19, 2015
Person assigned to the task	Project sponsor: Mary Smith
Deliverable(s)	Cost-benefit analysis
Completion status	Open
Priority	High
Resources that are needed	Spreadsheet software
Estimated time	16 hours
Actual time	14.5 hours

Figure 3: Structure of a task

The first thing a project manager must do is to **identify the tasks** that need to be accomplished and determine how long each task will take. Tasks and their identification and documentation are the basis of all three of these tools. Once the tasks have been identified and documented, they are organized within a work breakdown structure that is used to drive the creation of Gantt charts and network diagrams that can be used to **graphically portray** a traditional workplan.

1) A work-breakdown structure

A project manager can use a structured, top-down approach whereby high-level tasks are first defined and then broken down into subtasks. A list of tasks **hierarchically numbered** as in figure 4 is called a work breakdown structure (WBS). The number of tasks and level of detail depend on the complexity and size of the project. Key milestones, or important dates, are also identified on the workplan.

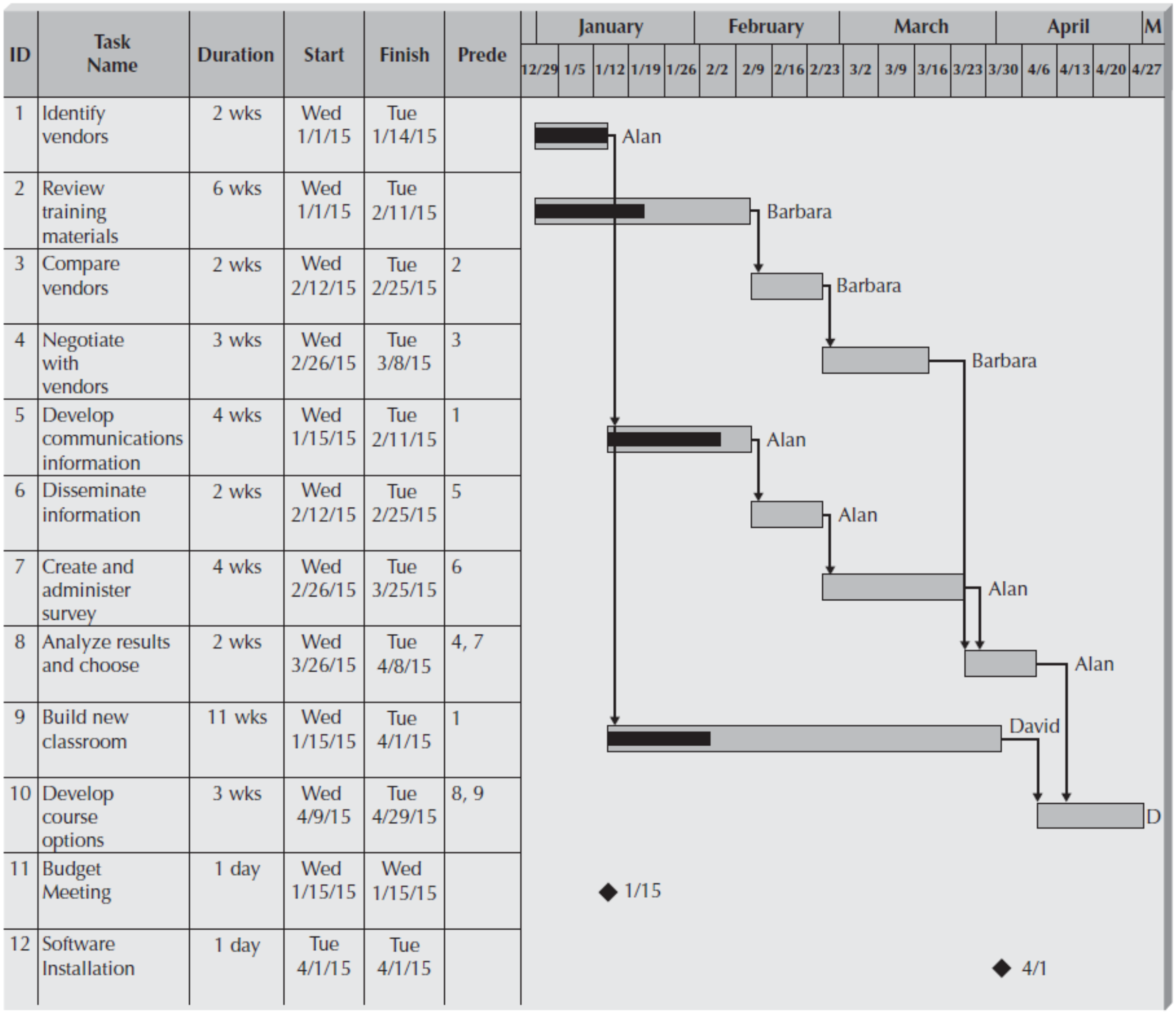


Task Number	Task Name	Duration (in weeks)	Dependency	Status
1	Identify vendors	2		Complete
2	Review training materials	6	1	Complete
3	Compare vendors	2	2	In Progress
4	Negotiate with vendors	3	3	Open
5	Develop communications information	4	1	In Progress
6	Disseminate information	2	5	Open
7	Create and administer survey	4	6	Open
7.1	Create initial survey	1		Open
7.2	Review initial survey	1	7.1	Open
7.2.1	Review by Director of IT Training	1		Open
7.2.2	Review by Project Sponsor	1		Open
7.2.3	Review by Representative Trainee	1		Open
7.3	Pilot test initial survey	1	7.1	Open
7.4	Incorporate survey changes	1	7.2, 7.3	Open
7.5	Create distribution list	0.5		Open
7.6	Send survey to distribution list	0.5	7.4, 7.5	Open
7.7	Send follow-up message	0.5	7.6	Open
7.8	Collect completed surveys	1	7.6	Open
8	Analyze results and choose vendor	2	4, 7	Open
9	Build new classrooms	11	1	In Progress
10	Develop course options	3	8, 9	Open

Figure 4: Work Breakdown Structure

2) Gantt Chart

A Gantt chart is a horizontal bar chart that shows the same task information as the project WBS but in a graphical way. Sometimes a picture really is worth a thousand words, and the Gantt chart can communicate the high-level status of a project much faster and easier than the WBS. Creating a Gantt chart is simple and can be done using a spreadsheet package, graphics software, or a project management package.



ID	Task Name	Duration	Start	Finish	Prede
1	Identify vendors	2 wks	Wed 1/1/15	Tue 1/14/15	
2	Review training materials	6 wks	Wed 1/1/15	Tue 2/11/15	
3	Compare vendors	2 wks	Wed 2/12/15	Tue 2/25/15	2
4	Negotiate with vendors	3 wks	Wed 2/26/15	Tue 3/8/15	3
5	Develop communications information	4 wks	Wed 1/15/15	Tue 2/11/15	1
6	Disseminate information	2 wks	Wed 2/12/15	Tue 2/25/15	5
7	Create and administer survey	4 wks	Wed 2/26/15	Tue 3/25/15	6
8	Analyze results and choose	2 wks	Wed 3/26/15	Tue 4/8/15	4, 7
9	Build new classroom	11 wks	Wed 1/15/15	Tue 4/1/15	1
10	Develop course options	3 wks	Wed 4/9/15	Tue 4/29/15	8, 9
11	Budget Meeting	1 day	Wed 1/15/15	Wed 1/15/15	
12	Software Installation	1 day	Tue 4/1/15	Tue 4/1/15	

Figure 5: Gantt Chat

3) Network Diagram

A second graphical way to look at project workplan information is the network diagram that lays out the project tasks in a **flowchart**. The network diagram is drawn as a **node-and-arc type of graph** that shows **time estimates in the nodes and task dependencies on the arcs**. Each **node**

represents an individual task, and a **line connecting two nodes represents the dependency between two tasks**. Network diagrams are the best way to communicate task dependencies because they lay out the tasks in the order in which they need to be completed. See Figure 6 below.

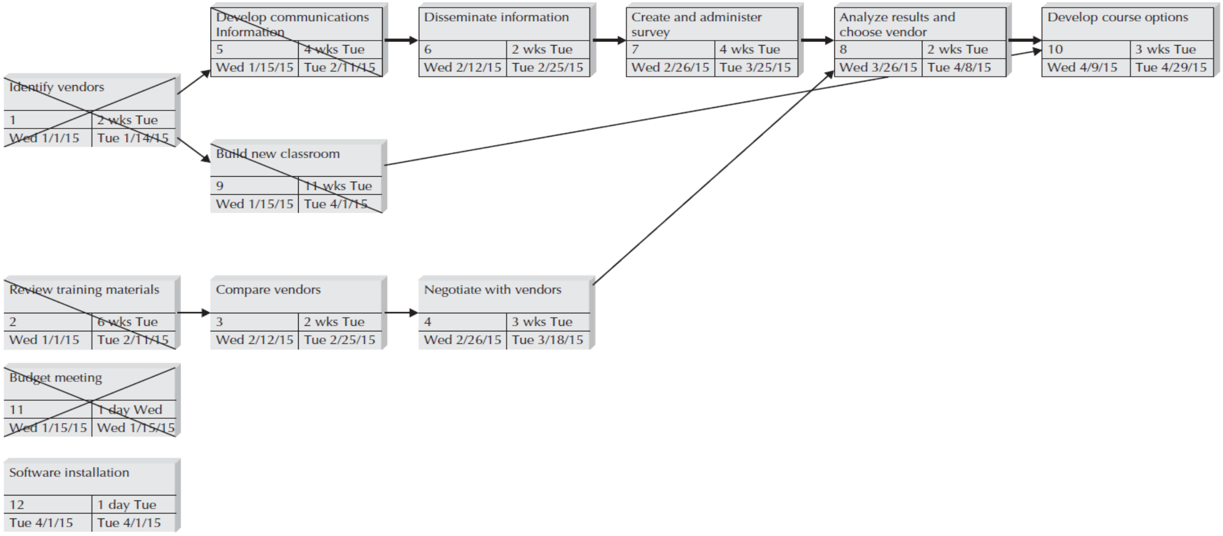


Figure 6: Network Diagram

PROJECT EFFORT ESTIMATION

Estimation is the process of assigning projected values for **time and effort**. The estimates developed at the start of a project are usually based on a range of **possible** values and gradually become more **specific** as the project moves forward. That is, the range of values for the inception phase will be much greater than for the transition phase.

There are a variety of ways to estimate the time required to build a system. Because the Unified Process is **use-case** (*a list of actions or event steps, typically defining the interactions between a role (an actor) and a system, to achieve a goal*) driven, we use an approach that is based on use cases: **use-case points**. Use-case models have two primary constructs: **actors** and **use cases**.

An **actor represents a role** that a user of the system plays, not a specific user. Actors can also represent other systems that will interact with the system under development.

A **use case** represents a major business process that the system will perform that benefits the actor(s) in some manner.

Depending on the number of unique transactions that the use case must address, a use case can be categorized as being simple, average, or complex. A use case is classified as **simple** if it supports one to three transactions, as **average** if it supports four to seven transactions, or as **complex** if it supports more than seven transactions.

CREATING AND MANAGING THE WORKPLAN

Once a project manager has a general idea of the **functionality** and **effort** for the project, he or she creates a **workplan**, which is a **dynamic schedule that records and keeps track of all the tasks that need to be accomplished over the course of the project**.

The workplan 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. There are **standard lists of tasks, or methodologies**, that are available for use as a **starting point**. One approach for identifying tasks is to get a list of tasks that has already been developed and to modify it. Others include to purchase from consultants or vendors, or books than can serve as a guide.

Evolutionary Work Breakdown Structures and Iterative Workplans

Because object-oriented systems approach to systems analysis and design support incremental and iterative development, any project planning approach for object-oriented systems development also requires an incremental and iterative process. **Evolutionary WBSs** as an object-oriented systems approach to designing a workplan, allow the analyst to develop an **iterative workplan.**

Managing Scope

An analyst may assume that a project will be safe from scheduling problems because he or she carefully estimated and planned the project up front. However, the most common reason for schedule and cost overruns—**scope creep**—occurs after the project is under way. Scope creep happens when new requirements are added to the project after the original project scope was defined and frozen. It can happen for many reasons:

1) Users might suddenly understand the potential of the new system and realize new functionality that would be useful.
2) Developers might discover interesting capabilities to which they become very attached.
3) A senior manager might decide to let this system support a new strategy that was developed at a recent board meeting.

Fortunately, using an **iterative and incremental development process** allows the team to deal with changing requirements in an effective way. The key in avoiding schedule and cost overruns is to identify the requirements as much as possible in the beginning of the project and to apply analysis techniques effectively. Of course, some requirements may be missed no matter what precautions are taken. However, the project manager should allow only absolutely necessary requirements to be added after the project begins. Sometimes changes cannot be incorporated into the present system even though they truly would be beneficial. In this case, these additions should be recorded as future enhancements to the system.

Timeboxing

Another approach to scope management is a technique called timeboxing. This technique sets a fixed deadline for a project and delivers the system by that deadline no matter what, even if functionality needs to be reduced. Timeboxing 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 short time frame. Several steps are involved in implementing timeboxing on a project. First, set the date of delivery for the proposed goals. The deadline should not be impossible to meet, so it is best to let the project team determine a realistic due date.

Refining Estimates

The estimates that are produced during inception need to be refined as the project progresses. This does not mean that estimates were poorly done at the start of the project; rather, it is virtually impossible to develop an exact assessment of the project's schedule at the beginning of the development process. A project manager should expect to be satisfied with broad ranges of estimates that become more and more specific as the project's product becomes better defined.

During planning, when a system is first requested, the project sponsor and project manager attempt to predict how long the development process 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 elaboration, 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 it is delivered.

Managing Risk

One final facet of project management is **risk management**, the process of assessing and addressing the risks that are associated with developing a project. Many things can cause risks: **weak personnel, scope creep, poor design, and overly optimistic estimates**. The project team must be aware of potential risks so that problems can be avoided or controlled well ahead of time. Typically, project teams create a **risk assessment**, or a **document that tracks potential risks** along with an evaluation of the likelihood of each risk and its potential impact on the project.

A paragraph or two is also included to explain potential ways that the risk can be addressed. There are many options including:

1) The risk could be publicized (give out information for advertising or promotional purposes),
2) Avoided, or
3) Even eliminated by dealing with its root cause.

For example, imagine that a project team plans to use new technology but its members have identified a risk in the fact that its members do not have the right technical skills. They believe that tasks may take much longer to perform because of a high learning curve. One plan of attack could be to eliminate the root cause of the risk—the lack of technical experience by team members—by finding the time and resources needed to provide proper training to the team.

STAFFING THE PROJECT

Staffing the project includes a series of activities, these include:

1) Determining how many people should be assigned to the project,
2) Matching people's skills with the needs of the project,
3) Motivating them to meet the project's objectives, and
4) Minimizing the conflict that will occur over time.

The deliverables for this part of project management are a **staffing plan**, which describes the number and kinds of people who will work on the project, the **overall reporting structure**, and the **project charter**, which describes the project's objectives and rules.

Staffing Plan

The first step to staffing is determining the average number of staff needed for the project. Many times, the temptation is to assign more staff to a project to shorten the project's length, but this is not a wise move. Adding staff resources does not translate into increased productivity; staff size and productivity share a disproportionate relationship, mainly because it is more difficult to coordinate a large number of staff members. The more a team grows, the more difficult it becomes to manage. Imagine how easy it is to work on a two-person project team: The team members share a single line of communication. But adding two people increases the number of communication lines to six, and greater increases lead to more dramatic gains in communication complexity. Figure 7 illustrates the impact of adding team members to a project team.

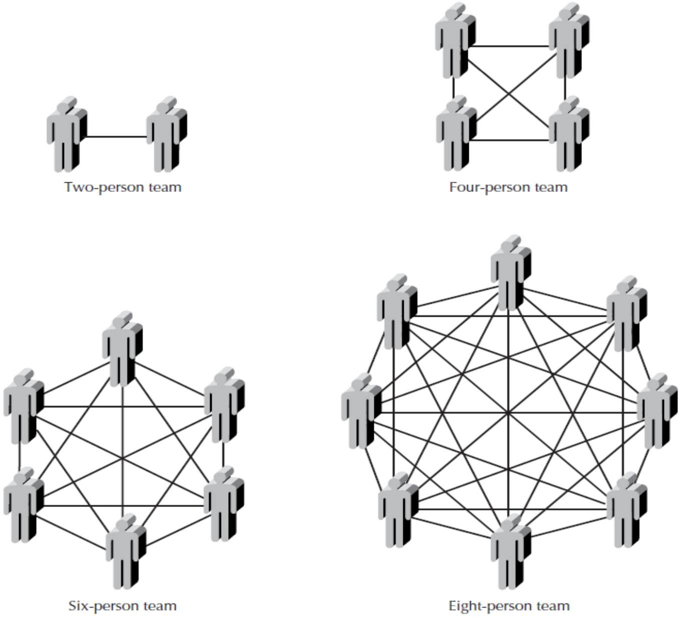


Figure 7: Increasing Complexity with Larger Teams

The general rule is to keep team sizes to fewer than eight to ten people; therefore, if more people are needed, create sub-teams. In this way, the project manager can keep the communication effective within small teams, which, in turn, communicate to a contact at a higher level in the project. Typically, a project has one project manager who oversees the overall progress of the development effort, with the core of the team comprising the various types of analysts described in module 1. A **functional lead** is usually assigned to manage a group of analysts, and a **technical lead** oversees the progress of a group of programmers and more technical staff members. There are many structures for project teams; Figure 8 illustrates one possible configuration of a project team.

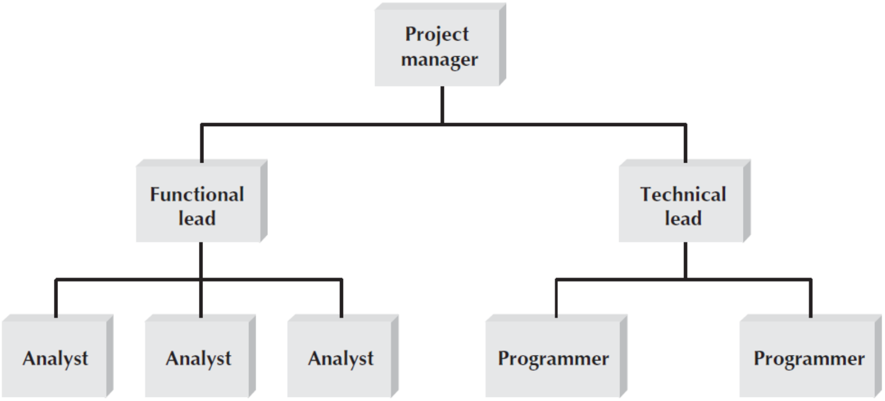


Figure 8: Possible Reporting structure

Motivation

Assigning people to tasks isn't enough; project managers need to motivate the people to ensure a project's success. **Motivation** has been found to be the number one influence on people's performance, but determining how to motivate the team can be quite difficult. You might think that good project managers motivate their staff by rewarding them with money and bonuses, but most project managers agree that this is the last thing that should be done. The more often managers reward team members with money, the more they expect it and most times monetary motivation won't work. Pink (a researcher) has suggested a set of principles to follow to motivate individuals in twenty-first century firms:

1) Pink suggests considering using some form of the 20 percent time rule to motivate individuals. This rule suggests that 20 percent of an employee's time should be spent on some idea in which he or she believes.

2) That firms should be willing to fund small "Now That" awards. These awards are given as small signs of appreciation for doing a great job. However, these awards are not given by a manager to an employee but from an employee to a peer of the employee.

3) If an employee (or team member) refers to the firm (the team) as "they," then there is the real possibility that the employee feels disengaged or possibly alienated. On the other hand, when employees refer to the firm as "we," they obviously feel like they are part of the organization.

4) Pink suggests that management should periodically consider giving each employee a day on which he or she can work on anything he or she wants. In some ways, this is related to the 20 percent rule.

Handling Conflict

The third component of staffing is organizing the project to minimize conflict among group members. **Group cohesiveness** (the attraction that members feel to the group and to other members) contributes more to productivity than do project members' individual **capabilities or experiences**.

Clearly defining the roles on the project and holding team members accountable for their tasks are a good way to begin mitigating potential conflict on a project.

Some project managers develop a **project charter***,* which lists the project's norms and **ground rules.** For example, the charter may describe:

- When the project team should be at work,
- When staff meetings will be held,
- How the group will communicate with each other, and
- What are the procedures for updating the workplan as tasks are completed.

ENVIRONMENT & INFRASTRUCTURE MANAGEMENT

The **environment and infrastructure management** workflows support the development team throughout the development process.

- **The environment workflow:** primarily deals with choosing the correct set of **tools** that will be used throughout the development process and identifying the appropriate set of **standards** to be followed during the development process.
- **Infrastructure workflow:** deals with choosing the appropriate **level** and **type of documentation** that will be created during the development process.

CASE Tools

Computer-aided software engineering (CASE) is a category of software that automates all or part of the development process. Some CASE software packages are used primarily to support the analysis workflow to create **integrated diagrams** of the system and **to store information regarding the system components**, whereas others support the design workflow that can be used to generate code for database tables and system functionality. Other CASE tools contain functionality that supports tasks throughout the system-development process.

The benefits of using CASE are numerous. With CASE tools, tasks can be

- Completed and altered faster,
- Development documentation is centralized, and
- Information is illustrated through diagrams, which are typically easier to understand.
- Reduce maintenance costs,
- Improve software quality, and

- Enforce discipline.

Examples of CASE tools include the requirement analysis tools, structure analysis tools, software design tools, code generation tools, test case generation tools, document production tools, reverse engineering tools etc. **(Find out examples under each)**

Standards

One way to make certain that everyone is performing tasks in the same way and following the same procedures is to create **standards** that the project team must follow. Standards can include formal rules for naming files, forms that must be completed when goals are reached, and programming guidelines.

When a team forms standards and then follows them, the project can be completed faster because task coordination becomes less complex. Standards work best when they are created at the beginning of each major phase of the project and communicated clearly to the entire project team. As the team moves forward, new standards are added when necessary.

Documentation

Finally, during the inception phase of the infrastructure workflow, project teams establish good **documentation standards** that include detailed information about the tasks of the Unified Process. Typically, the standards for the required documentation are set by the development organization. The development team only needs to ascertain which documentation standards are appropriate for the current systems development project.