Bahria University-Karachi Campus

Software Project Management

Fall-2024

Week 05

Engr. Majid Kaleem

مدرس: مہندس ماجد کلیم

جامعہ بحریہ، واقعگاہ کراچی

ACTIVITY ANALYSIS - PERT

- A program evaluation review technique (PERT) chart is a graphical representation of a project's timeline that displays all of the individual tasks necessary to complete the project.

SPM-FALL-2024

ACTIVITY ANALYSIS - PERT

1. Lag time is a delay in carrying out activities.
2. Lead time is an advance in carrying out the activities.
3. Expected time is the best estimation of how long a task will take to complete, considering any problems or obstacles that might arise.
4. Optimistic time refers to the minimum time it will take to complete a task.
5. Pessimistic time is the maximum time it will take to finish a task.
6. Most likely time is the best guess of how long a task will take, assuming no problems arise.

SPM - FALL - 2024

THREE-POINT ESTIMATE

- Three-point is a technique that involves people who are professional in the task we are estimating by this technique.
- In three-point estimation, *three figures* are produced initially for every distribution that is required, based on *prior experience* or *best guesses*:
  1. The first is a most likely (M)/best guess which is the *average* amount of work the task might take if the team member performed it 100 (multiple) times.
  2. The second estimate is the pessimistic (P) estimate which is the amount of work the task might take if the *negative factors* they identified do occur.
  3. The third estimate is the optimistic (O) estimate which is the amount of work the task might take if the *positive risks* they identified do occur.

SPM - FALL - 2024

THREE-POINT ESTIMATE

- What is Risk?
  - Risk is simply an *event* which has the potential to *impact* on your objectives (project).
  - Risk is the *possibility* that an outcome will not be as *expected.*

SPM-FALL-2024

THREE-POINT ESTIMATE

- In software development, risks can be categorized as positive (opportunities) or negative (threats).
- Positive Risks (Opportunities):
  - Early Completion of a Project: If the development team is more efficient than expected, the project could be completed ahead of schedule, allowing for earlier release and cost savings.
  - Market Demand Surges: In some cases, an unexpected surge in market demand for a software product can be a positive risk. It could lead to increased revenue and market share.
  - Innovative Technology Adoption: Embracing cutting-edge technology might be seen as a risk, but it can result in a competitive advantage and attract more users or customers.
  - Highly Skilled Team Members: Hiring exceptionally skilled developers can lead to better quality and faster development, reducing time and costs.

SPM-FALL-2024

THREE-POINT ESTIMATE

- *Positive Risks Example:*
  - Every project leader develops a budget for their respective project and its resource needs.
  - However, as in most things in life, there are often adjustments throughout the course of projects.
  - Many times, a project finishes well under budget, *which is technically an error* that is attributable to the project leader's *miscalculation or lack of foresight*.

THREE-POINT ESTIMATE

- *Positive Risks Example (contd):*
  - Businesses are increasingly searching for new ways to incorporate technology for greater efficiency.
  - More and more organizations are finding that, with such technological investments, *they may be eliminating the need for some jobs* within the company.
  - While this type of risk is bad for employees who may risk losing their jobs due to technological efficiency, *it does benefit the company in savings*.

❖ They are "risks" in the sense that they are ***uncertain*** events, and their realization is not guaranteed.

THREE-POINT ESTIMATE

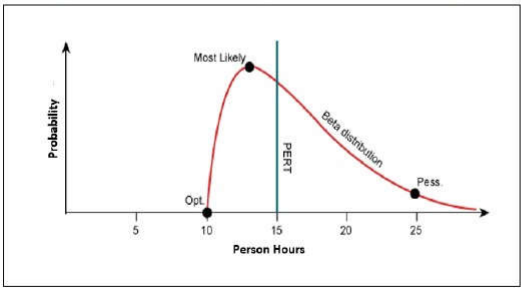
- Positive Risks:
  - Majority of coding can be copied from previous works
  - Client is easy to deal with and specific
  - An already designed theme suits your purpose
  - UNDER BUDGET
- Negative Risks:
  - Client refuses designs
  - Preferred domain name is taken
  - Website gets hacked etc.
  - OVER BUDGET

THREE-POINT ESTIMATE

- Each task receives three estimates:

1. *Optimistic*. Assuming the (NEGATIVE) risks do not occur.
2. *Most likely*. Assuming the risk occurrence level that is most likely to occur. Similar to other estimate types.
3. *Pessimistic*. Assuming that the (NEGATIVE) risks occur.

- Expected Duration (Beta Distribution) Formula



Person-hour: a unit of one hour's work by one person

Standard Deviation for Beta Distribution

$\sigma = (P - O)/6$

LOC-THREE-POINT ESTIMATE

Module	Max Size/Optimistic	Best Guess/ Most Likely	Min Size/Pessimistic
1	20	30	50
2	10	15	25
3	25	30	45
4	30	35	40
5	15	20	25
6	10	12	14
7	20	22	25

SPM-FALL-2024

LOC-THREE-POINT ESTIMATE

$$\text{Module } 1 = \frac{(20 + 4 \times 30 + 50)}{6} = \frac{190}{6} = 31.6$$

$$\text{Module } 2 = \frac{(10 + 4 \times 15 + 25)}{6} = \frac{95}{6} = 15.8$$

$$\text{Module } 3 = \frac{(25 + 4 \times 30 + 45)}{6} = \frac{190}{6} = 31.6$$

$$\text{Module } 4 = \frac{(30 + 4 \times 35 + 40)}{6} = \frac{220}{6} = 36.7$$

$$\text{Module } 5 = \frac{(15 + 4 \times 20 + 25)}{6} = \frac{120}{6} = 20$$

$$\text{Module } 6 = \frac{(10 + 4 \times 12 + 14)}{6} = \frac{72}{6} = 12$$

$$\text{Module } 7 = \frac{(20 + 4 \times 22 + 25)}{6} = \frac{133}{6} = 22.17$$

SPM-FALL-2024

LOC-THREE-POINT ESTIMATE

The estimate for the whole application is the sum of the estimates for each module :

$$\text{Whole} = 31.6 + 15.8 + 31.6 + 36.7 + 20 + 12 + 22.17 = 170.07 \text{ LOC}$$

The estimate for the standard deviation of the estimate is as follows :

$$\sigma = \sqrt{(50-20)^2 + (25-10)^2 + (45-25)^2 + (40-30)^2 + (25-15)^2 + (14-10)^2 + (25-20)^2}$$

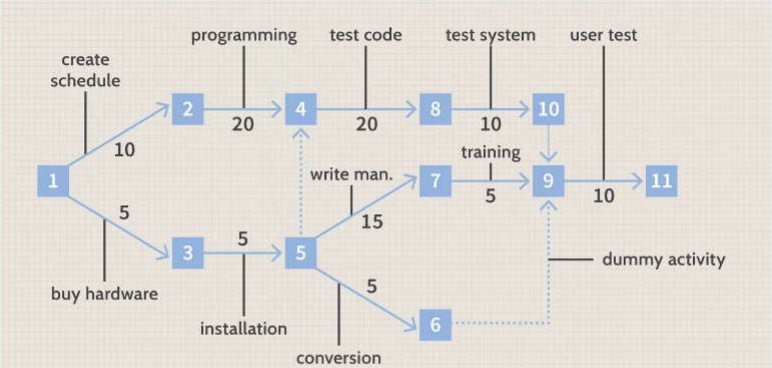
$$\sigma = \sqrt{900 + 225 + 100 + 100 + 16 + 25}$$

$$\sigma = 42.03$$

Mean (Average) + Standard Deviation = 170.07 + 42.03=**212.1**
Mean (Average) - Standard Deviation = 170.7 - 42.03=**128.04**

} OK

ACTIVITY ANALYSIS - PERT

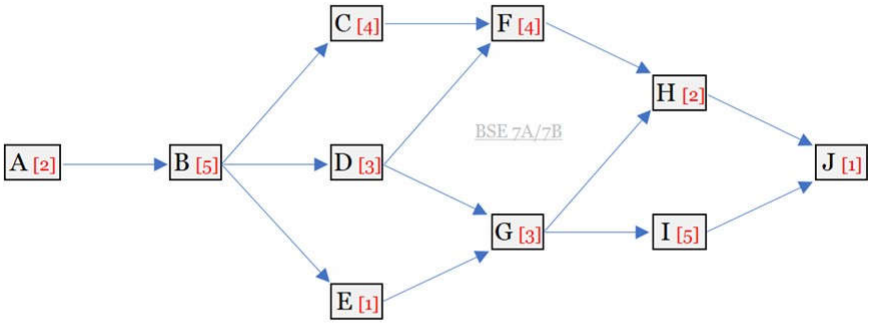


- Numbered rectangles are nodes and represent events or milestones.
- Directional arrows represent dependent tasks that must be completed sequentially.
- Diverging arrow directions (e.g. 1-2 & 1-3) indicate possibly concurrent tasks.
- Dotted lines indicate dependent tasks that do not require resources.

ACTIVITY ANALYSIS - PERT

Activity	Predecessor	Optimistic Estimate *(Days)* 'a'	Most Likely Estimate *(Days)* 'b'	Pessimistic Estimate *(Days)* 'c'	Expected Duration (a+4b+c)/6
A	None	1	2	4	2.2
B	A	3	5	8	5.2
C	B	2	4	5	3.8
D	B	2	3	6	3.3
E	B	1	1	1	1.0
F	C, D	2	4	6	4.0
G	D, E	2	3	4	3.0
H	F, G	1	2	5	2.3
I	G	4	5	9	5.5
J	H, I	5	1	3	1.3

PRECEDENCE DIAGRAM METHOD (PDM/AON)



POSSIBLE ACTIVITY PATHS

Possible paths	Path	Total
Path 1	A + B + C + F + H + J	18.8
	2.2 + 5.2 + 3.8 + 4.0 + 2.3 + 1.3	
Path 2	A + B + D + F + H + J	18.3
	2.2 + 5.2 + 3.3 + 4.0 + 2.3 + 1.3	
Path 3	A + B + D + G + H + J	18.6
	2.2 + 5.2 + 3.3 + 4.0 + 2.3 + 1.3	
Path 4	A + B + D + G + I + J	20.5
	2.2 + 5.2 + 3.3 + 3.0 + 5.5 + 1.3	
Path 5	A + B + E + G + I + J	18.2
	2.2 + 5.2 + 1.0 + 3.0 + 5.5 + 1.3	

SPM-FALL-2024

PRECEDENCE DIAGRAM METHOD (PDM/AON)

Possible paths	Path	Total
Path 1	A + B + C + F + H + J	18
	2 + 5 + 4 + 4 + 2 + 1	
Path 2	A + B + D + F + H + J	17
	2 + 5 + 3 + 4 + 2 + 1	
Path 3	A + B + D + G + H + J	16
	2 + 5 + 3 + 3 + 2 + 1	
Path 4	A + B + D + G + I + J	19
	2 + 5 + 3 + 3 + 5 + 1	
Path 5	A + B + E + G + I + J	17
	2 + 5 + 1 + 3 + 5 + 1	

SPM-FALL-2024

How to Plan a Project?

1. Define Objectives and Scope:
   - Start by clearly defining the project's objectives, goals, and scope. What is the problem you are trying to solve, and what are the desired outcomes? Be as specific and detailed as possible.
2. Gather Requirements:
   - Collect and document all project requirements. This involves understanding the needs of stakeholders, end-users, and any regulatory or compliance requirements. Create a detailed requirements document that outlines functional and non-functional requirements.
3. Create a Project Charter:
   - Develop a project charter that includes the project's purpose, objectives, scope, stakeholders, roles and responsibilities, and a high-level project schedule. This document will serve as a reference point throughout the project.

How to Plan a Project?

4. Build a Project Team:
   - Assemble a team with the necessary skills and expertise to complete the project successfully. Define roles and responsibilities, and ensure that team members understand their tasks and deadlines.
5. Develop a Project Plan:
   - Create a comprehensive project plan that outlines the project's timeline, tasks, dependencies, milestones, and resource allocation. Consider using project management software or tools like Gantt charts to visualize and manage the plan.
6. Risk Assessment and Management:
   - Identify potential risks and uncertainties that may impact the project's success. Develop a risk management plan that outlines how these risks will be mitigated or managed.

How to Plan a Project?

7. Estimate Resources and Budget:
   - Estimate the resources (human, hardware, software, etc.) required for the project and create a budget. Ensure that the budget is realistic and that you have contingency plans in case of unforeseen expenses.
8. Select a Development Methodology:
   - Choose a software development methodology that aligns with the project's goals and constraints. Popular options include Agile, Waterfall, Scrum, and Kanban. The methodology will influence how tasks are executed and managed.
9. Create a Work Breakdown Structure (WBS):
   - Break down the project into smaller, manageable tasks and create a Work Breakdown Structure (WBS). Each task should have a clear description, estimated duration, and assigned team members.

How to Plan a Project?

10. Define Quality Standards:
    - Establish quality assurance and testing criteria. Define how the software will be tested, what constitutes a successful test, and the criteria for acceptance.
11. Communication Plan:
    - Develop a communication plan that outlines how information will be shared among team members, stakeholders, and clients. Regular status meetings and progress reports are essential for keeping everyone informed.
12. Monitoring and Control:
    - Implement project monitoring and control mechanisms. Track progress against the project plan, identify deviations and take corrective actions as needed.

How to Plan a Project?

13. Documentation and Knowledge Sharing:
    - Ensure that all project-related documentation is properly maintained and accessible to the team. Encourage knowledge sharing and document best practices and lessons learned.
14. Testing and Quality Assurance:
    - Execute thorough testing and quality assurance processes as per the defined quality standards. Address and resolve any issues or defects that arise during testing.
15. Deployment and Delivery:
    - Plan for the deployment and delivery of the software to end-users or clients. Ensure that user documentation and training are provided as necessary.

How to Plan a Project?

16. Post-Implementation Review:
    - After deployment, conduct a post-implementation review to assess the project's success, gather feedback, and identify areas for improvement.
17. Project Closure:
    - Close the project by finalizing all documentation, conducting a project closure meeting, and transitioning any remaining tasks or maintenance responsibilities to the appropriate parties.

WORK BREAKDOWN STRUCTURE (WBS)

- A Work Breakdown Structure (WBS) is a hierarchical decomposition of a project into smaller, manageable components or *work packages*.

- ***Types of WBS:***

  1. Deliverable-oriented work breakdown structure
     - A deliverable-oriented WBS (also known as Product-Oriented or Noun-Oriented WBS) decomposes the project scope into smaller and more manageable deliverables.

  2. Phase-oriented work breakdown structure
     - A process-oriented WBS (also known as Process-Oriented or Verb-Oriented WBS) defines what process steps need to be taken to deliver each of the project deliverables.

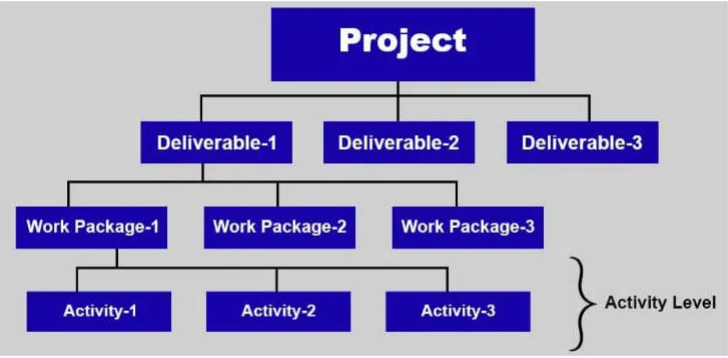
WORK BREAKDOWN STRUCTURE (WBS)

- ***Work Package:***

- It represents a specific, manageable, and well-defined portion of work within a project that can be assigned to a particular individual or team for execution.

- Work packages are typically at a lower level of the project's hierarchy and are further decomposed into smaller *tasks* or *activities* as needed.

WORK BREAKDOWN STRUCTURE (WBS)



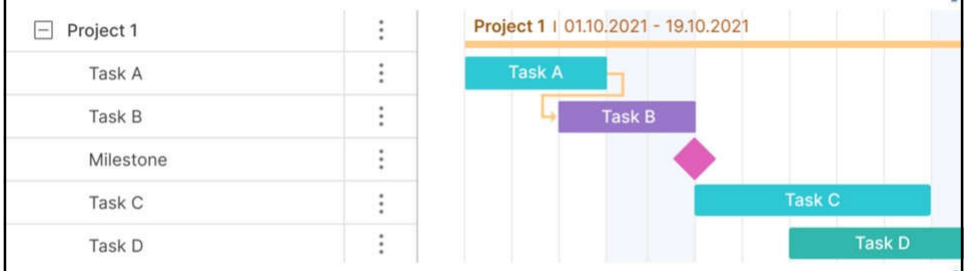
WORK BREAKDOWN STRUCTURE (WBS)

- **Deliverable: Payroll Processing Module**

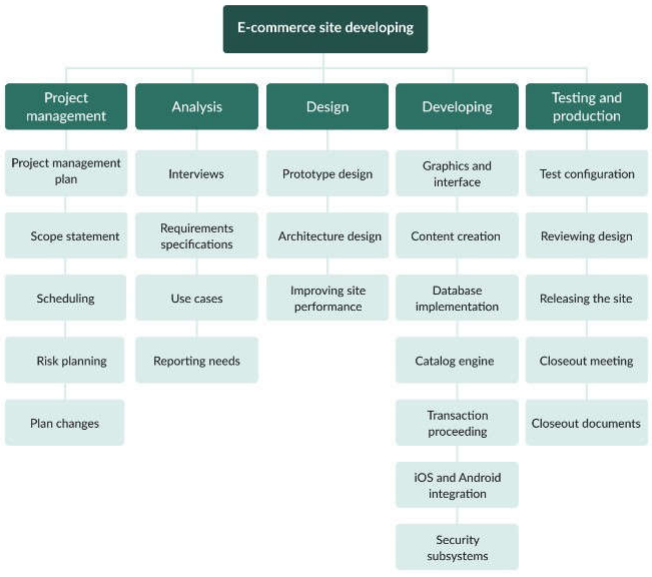
1. **Work Package: Requirements Gathering**
   - **Activity 1:** Conduct stakeholder interviews to gather payroll processing requirements.
   - **Activity 2:** Review existing payroll processes and documentation.
   - **Activity 3:** Document functional and non-functional requirements for the module.
2. **Work Package: Design and Architecture**

- **Activity 1:** Define the high-level architecture for the payroll processing module.
- **Activity 2:** Design the database schema for storing payroll data.
- **Activity 3:** Create a detailed design for payroll calculation logic.
- **Activity 4:** Design user interfaces for payroll input and reporting.

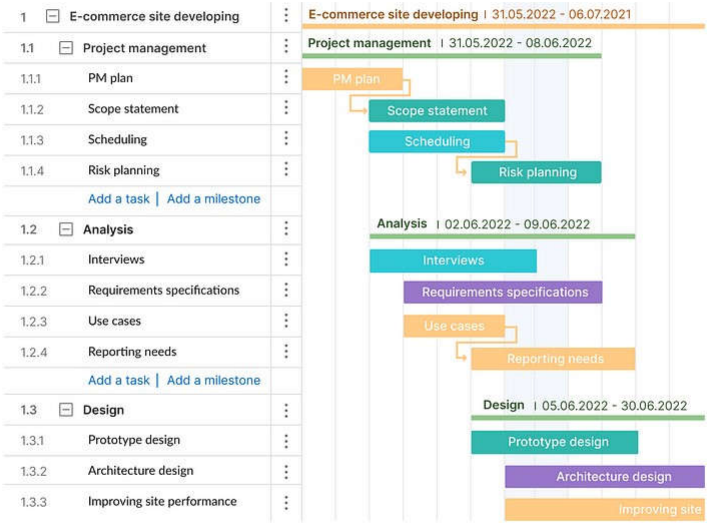
WBS & GANTT CHART



WBS & GANTT CHART



WBS & GANTT CHART



SIX METHODS FOR ESTIMATING TASK DURATION

1. Expert Judgment:
   - This method relies on the input of experienced individuals or subject matter experts. They use their knowledge and past experience to estimate how long a task will take. Expert judgment can be particularly valuable when dealing with unique or complex tasks.

2. Analogous Estimating:
   - Analogous estimating involves comparing the current task with similar tasks from past projects. By analyzing historical data and considering the similarities and differences between the tasks, you can estimate the duration of the new task. This method is often quick and relatively accurate.

Six Methods for Estimating Task Duration

3. Parametric Estimating:
   - Parametric estimating involves using mathematical models or algorithms to calculate task duration based on specific parameters. For example, you might estimate software development time based on lines of code, or construction time based on square footage. Parametric estimating is most effective when historical data and parameters are well-defined.

4. Three-Point Estimating (PERT):
   - The Program Evaluation and Review Technique (PERT) is a probabilistic method that considers three estimates for task duration: the most optimistic, most likely, and most pessimistic scenarios. These three estimates are then weighted to calculate a single, more realistic duration estimate. PERT helps account for uncertainty in task durations.

SPM-FALL-2024

Six Methods for Estimating Task Duration

5. Delphi Technique:
   - The Delphi technique involves soliciting input from a panel of experts anonymously. Each expert provides an estimate for task duration, and these estimates are compiled and shared with the group. The process is repeated until a consensus or convergence of estimates is reached. This method helps reduce bias and groupthink.

6. Bottom-Up Estimating:
   - Bottom-up estimating breaks down a project into its smallest work packages or tasks. Each task is estimated individually, and these estimates are then rolled up to determine the overall project duration. This method provides a detailed and granular view of task durations but can be time-consuming.

SPM-FALL-2024


To be continued...
SPM-FALL-2024
13-Oct-2024
Engr. Majid Kaleem
35