

Course: Software Project Management

Week 6: Work Breakdown Structure (WBS) and Scheduling

Lecturer: Yimer Amedie (MSc.)

Addis Ababa Science and Technology University, Ethiopia

Contents

- Introduction
- Overview of WBS and Schedule
- Designing WBS
- Software Project Schedule

Learning Outcomes

After completing this lesson, you will be able to:

- Explain and construct WBS
- Identify and explain common scheduling techniques and tools
- Develop a project schedule

Introduction

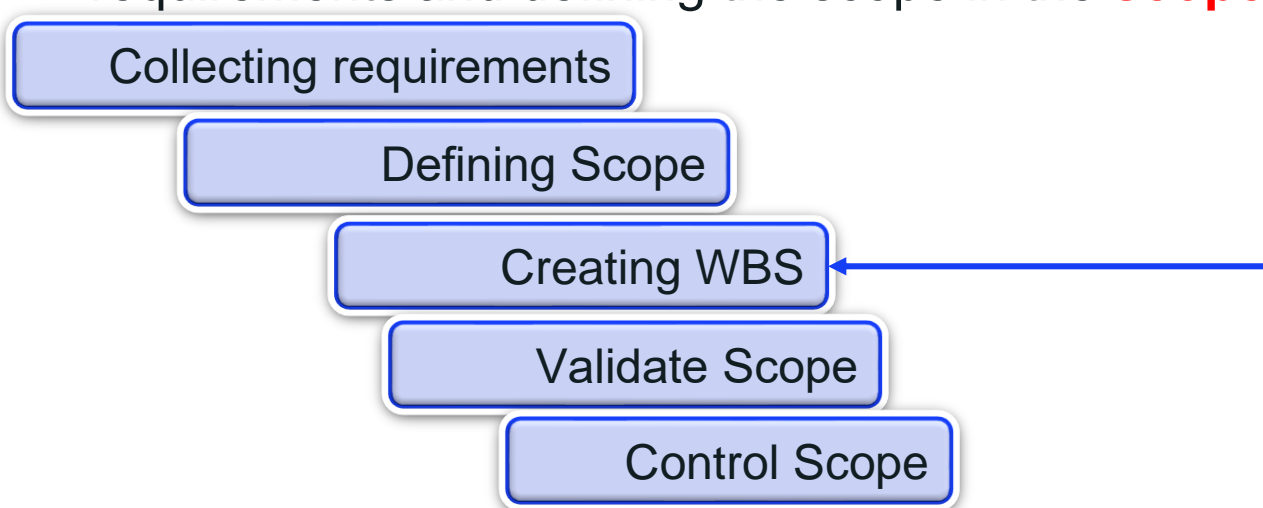
- In software project management, the five process groups or phases are the bases for its success.

Initiate → **Plan** → Execute → Monitor & Control → Close

- Planning is the foundation for successful project execution, and both
 - the WBS and scheduling are critical tools in this phase.
- Together, WBS and scheduling turn the high-level project plan into actionable steps, ensuring that the team knows
 - what to do, when to do it, and
 - how to allocate resources effectively.

Overview of WBS and Scheduling

- The project scope is the basis for lots of project management tasks including for **WBS** and **scheduling**.
- Creating the WBS is the next step after collecting and analyzing requirements and defining the scope in the **scope management**.



Overview of WBS and Scheduling

- The WBS is a **hierarchical decomposition** of the total scope of work to be carried out by the project team to
 - accomplish the project objectives and
 - create the required deliverables.
- The WBS
 - organizes and defines the total scope of the project, and
 - represents the work specified in the current project scope statement that are approved (PMI, 2013)

Overview of WBS and Scheduling

Hierarchy

- structured as a tree diagram or an indented list, where the project starts at the top & branches down into smaller tasks or work packages.

Deliverable Oriented

- It focuses on the deliverables or outcomes of the project, not just the tasks that need to be completed

Level

- can have multiple levels depending on the complexity of the project.

Work Package (lowest level of WBS)

- a group of related tasks that can be assigned to a person or a team, with specific time, cost, and resource requirements.



WBS
Features

Overview of WBS and Scheduling

Inputs, Techniques and Outputs of WBS

- Inputs
 - Scope statement, Requirement document
- Techniques
 - Decomposition, Expert judgement
- Output
 - Scope baseline, WBS, Document update

WBS Example

● Online Learning System Development



○ Planning

- Define scope
- Gather requirement
- Identify stakeholders
- Create project plan and s

○ Design

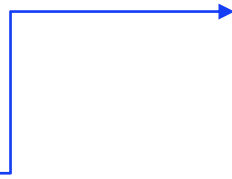
- Design architecture
- Design UI
- Design Database

○ Development

- Front-end development
- Back-end development

○ Development

- Front-end development
 - develop homepage
 - develop course catalogue
 - develop user profile
- Back-end development
 - DB Design and setup
 - Design DB schema
 - setup DB server
 - Develop API
 - Develop CRUD
 - Develop course mgmt.



- Online Learning System Development



WBS Example

- Planning

- Define scope
- Gather requirement
- Identify stakeholders
- Create project plan and schedule

- Design

- Design architecture
- Design UI
- Design Database

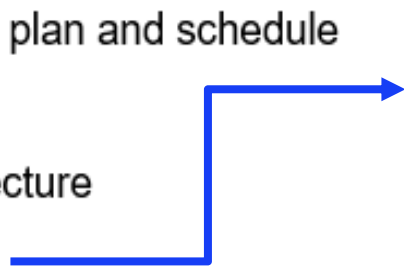
Designing login screens

Creating course overview pages

Setting up student dashboards

- Development

- Front-end development
- Back-end development

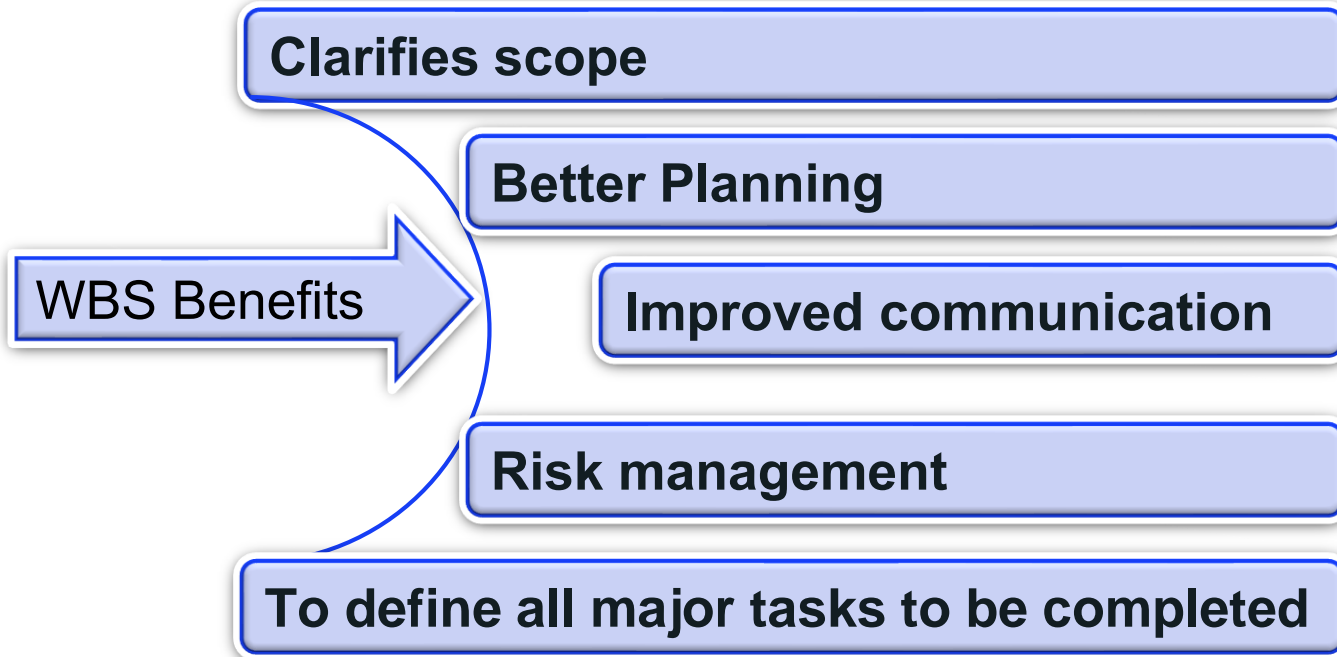


Overview of WBS and Scheduling



WBS Benefits

Overview of WBS and Scheduling



Overview of WBS and Scheduling

- WBS is used to define all major tasks or activities that need to be completed, **but NOT their sequences (order) and duration (time)**

Project Scheduling

- The process of planning and organizing the timeline for completing the project tasks
- Refinement of a plan that show dates and
 - when each activity should start and finish, by whom
 - when and how much of each resource will be required

Overview of WBS and Scheduling

- Developing and controlling the schedule is required for the successful completion of the project
- Schedule development and control is the 4th and 5th step of **time**

management.

1. Activity definition → WBS
2. Activity sequencing
3. Activity duration estimation
4. **Schedule development**
5. **Schedule control**

Refer Lecture 2 / Week 2

Overview of WBS and Scheduling

What is Software project scheduling?

- The process of planning, organizing, and managing the **timeline** of **tasks** and resources in a software development project
- Important so that it
 - Helps set clear deadlines.
 - Ensures proper resource allocation (developers, testers, tools).
 - Reduces delays and bottlenecks.
 - Increases predictability and control.
 - Keeps stakeholders informed of progress.

Overview of WBS and Scheduling

Scheduling Concepts in Software Projects

- **Task / Activity:** A specific piece of work to be done (e.g., **write code, test**).
- **Duration:** Estimated time to complete a task (**2 weeks**).
- **Dependencies:** Relationship between tasks (**"testing" depends on "coding"**).
- **Milestone:** A major checkpoint (e.g., **prototype ready, release the system**).
- **Resource Allocation:** Assigning people, tools, and money to tasks.
- **Slack/Float:** Extra time allowed for a task before it affects the deadline.
- **Critical Path:** Longest chain of dependent tasks that determines the project duration.

Overview of WBS and Scheduling

Steps in Software Project Scheduling

1. **Define Tasks:** Break the project into smaller tasks (WBS).
2. **Estimate Time:**
 - Estimate how long each task will take (use expert judgment, historical data, or estimation techniques like PERT, CPM).
3. **Identify Dependencies:**
 - Determine the order of tasks (which depend on others).
4. **Assign Resources:**
 - Assign team members, tools, or budget for each task.

Overview of WBS and Scheduling

Steps in Software Project Scheduling

5. Create a Timeline:

- Map tasks on a schedule using:
 - **Gantt Chart** (for visual timelines).
 - **PERT Chart** (for uncertainty).
 - **Critical Path Method (CPM)** (for deadline-sensitive planning).



Techniques

6. Monitor and Adjust:

- Continuously track progress and adjust the schedule if there are delays or resource changes.

Overview of WBS and Scheduling

Software Project Scheduling Techniques

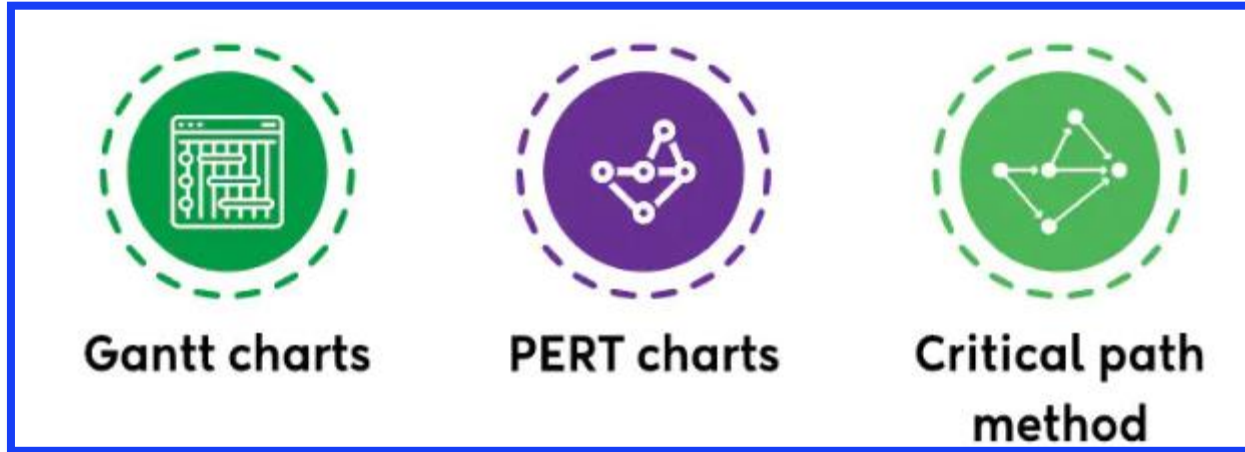


Figure 1 : Popular project scheduling techniques (Kashyap, 2024)

Overview of WBS and Scheduling

Software Project Scheduling Techniques

- **WBS:**
 - Breaks the whole project into smaller, manageable pieces
 - Advantages
 - Provides clear project structure
 - Helps identify all necessary tasks
 - Facilitates resource allocation

Overview of WBS and Scheduling

Software Project Scheduling Techniques

- **Network Diagram:**
 - A visual representation of a project's activities and the sequence in which they must be completed.
 - It shows **dependencies** between tasks and helps identify the **critical path**
 - the longest sequence of tasks that determines the shortest time to complete the project.
 - We can use [Lucidchart](#) or [Draw.io \(diagrams.net\)](#)

Overview of WBS and Scheduling

Software Project Scheduling Techniques

- **Gantt Chart**

- A horizontal bar chart that visualizes task durations and dependencies. Shows when tasks start and end.
- Highlights overlapping or dependent tasks.
- Advantages
 - Visual timeline of the project
 - Shows task relationships clearly
 - Easy to understand for stakeholders

Overview of WBS and Scheduling

Software Project Scheduling Techniques

- **Critical Path Method (CPM)**
 - Identifies the longest path of dependent tasks
 - Critical path (longest duration path)
 - Float/slack time (time tasks can be delayed)
 - Advantages:
 - Identifies most important tasks
 - Helps optimize schedule
 - Shows impact of delays

Overview of WBS and Scheduling

Software Project Scheduling Techniques

- **Program Evaluation and Review Technique (PERT)**
 - Probabilistic approach using three time estimates for each task

Expected Time = (Optimistic + 4×Most Likely + Pessimistic) / 6

- **Advantages:**
 - Accounts for **uncertainty**
 - Helps in risk assessment
 - Useful for complex projects

Overview of WBS and Scheduling

Software Project Scheduling Techniques

- **Agile Scheduling**
 - **Scrum**
 - Time-boxed sprints (typically 2-4 weeks)
 - Sprint planning meetings
 - Daily stand-ups
 - Burndown charts
 - The most effective overall choice
 - Flexibility, continuous feedback

Overview of WBS and Scheduling

Scheduling Example: [Online Learning management System]

- **WBS Structure**

1. Planning (including Requirements)
2. Design
3. Development (Frontend + Backend)
4. Integration (Combine Frontend & Backend)
5. Testing
6. Deployment

Overview of WBS and Scheduling

Scheduling Example: [Online Learning management System]

- Task breakdown with estimated duration and dependency

| Task ID | Task Name | Estimated Duration | Dependency |
|---------|-------------|--------------------|-------------------|
| 1 | Planning | 2 weeks | — |
| 2 | Design | 2 weeks | After Planning |
| 3 | Development | 6 weeks | After Design |
| 4 | Integration | 2 weeks | After Development |
| 5 | Testing | 3 weeks | After Integration |
| 6 | Deployment | 1 week | After Testing |

Overview of WBS and Scheduling

Scheduling Example: [Online Learning management System]

- Network Diagram → Text representation

[Start]



[Planning] → [Design] → [Development] → [Integration] →

[Testing] → [Deployment] → [End]

Overview of WBS and Scheduling

Scheduling Example: [Online Learning management System]

- Network Diagram → with Early Start / Late Start & Slack

| Activity | Predecessor | Duration (weeks) | Earliest Start (ES) | Earliest Finish (EF) | Latest Start (LS) | Latest Finish (LF) | Slack |
|-------------|-------------|------------------|---------------------|----------------------|-------------------|--------------------|-------|
| Planning | — | 2 | 0 | 2 | 0 | 2 | 0 |
| Design | Planning | 2 | 2 | 4 | 2 | 4 | 0 |
| Development | Design | 6 | 4 | 10 | 4 | 10 | 0 |
| Integration | Development | 2 | 10 | 12 | 10 | 12 | 0 |
| Testing | Integration | 3 | 12 | 15 | 12 | 15 | 0 |
| Deployment | Testing | 1 | 15 | 16 | 15 | 16 | 0 |

All slack = 0, meaning this is the **Critical Path**.

Overview of WBS and Scheduling

Scheduling Example: [Online Learning management System]

- Critical path Method

Critical Path Calculation: The path is:

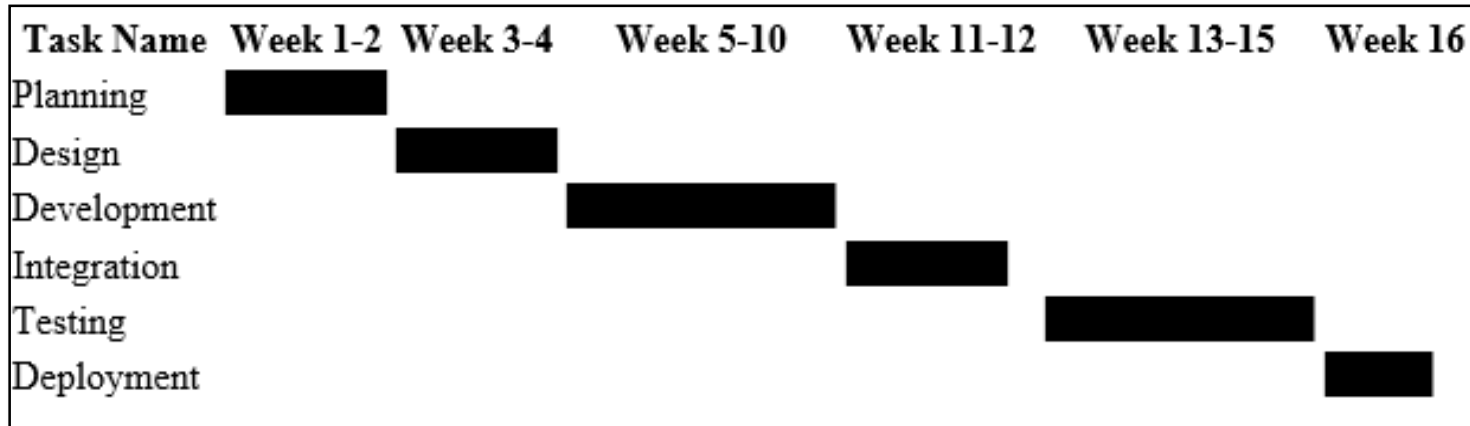
Planning → Design → Development → Integration
→ Testing → Deployment

- Total time (without buffer): $2 + 2 + 6 + 2 + 3 + 1 = 16$ weeks
- Any delay on this path will delay the whole project.

Overview of WBS and Scheduling

Scheduling Example: [Online Learning management System]

- Gantt Chart



Overview of WBS and Scheduling

Scheduling Example: [Online Learning management System]

• PERT

| Task Name | Optimistic (O) | Most Likely (M) | Pessimistic (P) | TE Calculation | Expected Time (TE) |
|-------------|----------------|-----------------|-----------------|------------------------------|--------------------|
| Planning | 1 week | 2 weeks | 4 weeks | $(1 + 4 \times 2 + 4) / 6$ | 2.16 weeks |
| Design | 1 week | 2 weeks | 3 weeks | $(1 + 4 \times 2 + 3) / 6$ | 2 weeks |
| Development | 4 weeks | 6 weeks | 8 weeks | $(4 + 4 \times 6 + 8) / 6$ | 6 weeks |
| Integration | 1 week | 2 weeks | 3 weeks | $(1 + 4 \times 2 + 3) / 6$ | 2 weeks |
| Testing | 2 weeks | 3 weeks | 5 weeks | $(2 + 4 \times 3 + 5) / 6$ | 3.16 weeks |
| Deployment | 0.5 week | 1 week | 2 weeks | $(0.5 + 4 \times 1 + 2) / 6$ | 1.08 weeks |

 **Total Estimated Project Duration (PERT): ~16-17 weeks**

Overview of WBS and Scheduling

Scheduling Example: [Online Learning management System]

- Total → ~17 weeks

Risk Buffer:

- Add ~10% extra time to the total (about 1.5-2 weeks) to cover unexpected risks:
- **New Total → ~18-19 Weeks.**

Overview of WBS and Scheduling

Scheduling Tools

- ✓ Microsoft Project
- ✓ Jira (Agile boards & sprints)
- ✓ Trello
- ✓ Asana

Commercial

- ✓ ProjectLibre
- ✓ GanttProject
- ✓ OpenProject
- ✓ Kanboard

Open Source

Overview of WBS and Scheduling

Why Project Scheduling? (Kashyap, 2024)

Provides relief from work stress

1.

Eliminates unnecessary costs

2.

Boosts project rate success

3.

Maximizes productivity

4.

Lower the risk of missing deadlines

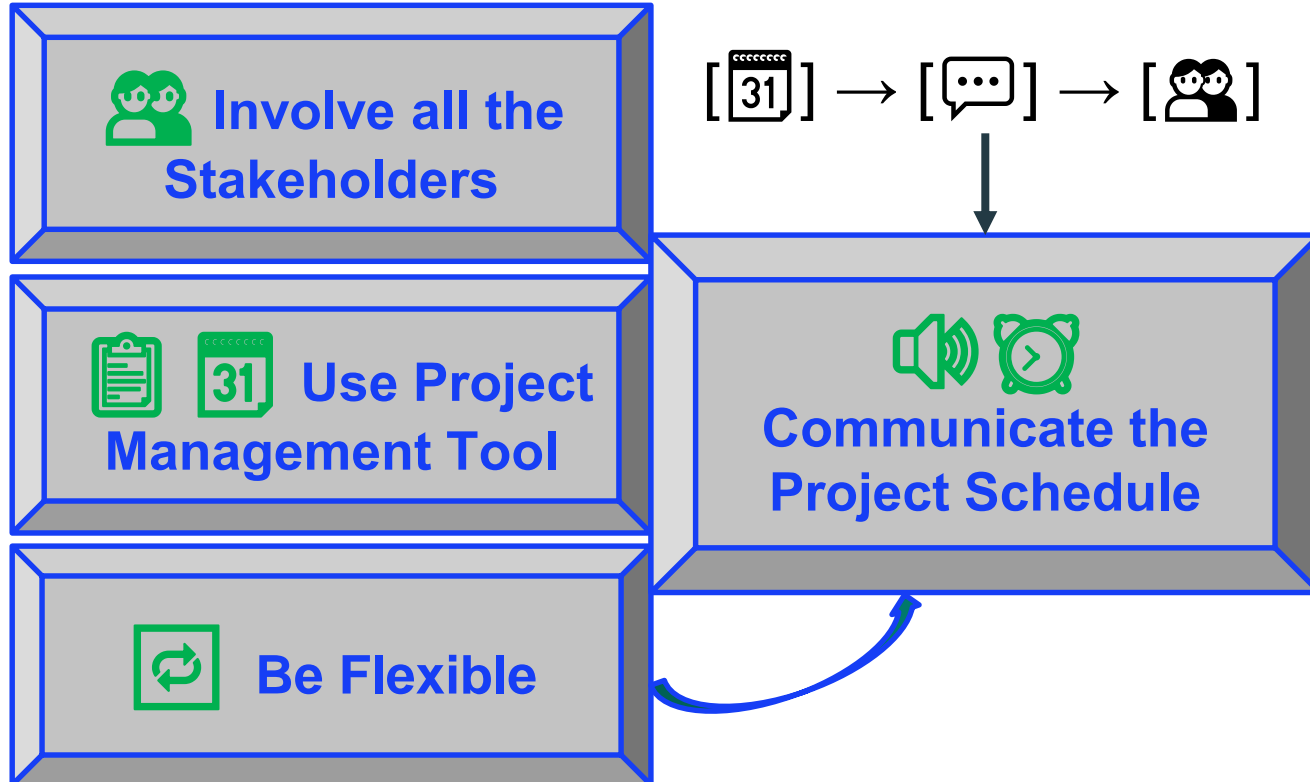
5.

Assists in better decision making

6.

Overview of WBS and Scheduling

Best Scheduling Practices



Summary

- A WBS is about breaking down the project into smaller pieces for better management, and Scheduling is about organizing these pieces over time to ensure the project is completed on schedule.
- Together, WBS and scheduling provide a clear roadmap for project execution, helping project managers control scope, resources, and time.
- The use of different scheduling tools is useful to facilitate efficient decision making, reduce confusion, and promote effective communication

Summary

- **Gantt charts** enhance project planning and communication by offering a clear picture of tasks, their durations, and dependencies.
- In contrast, **PERT charts** incorporate probabilistic techniques to handle uncertainty,
 - delivering a more refined and realistic project schedule.
- The **CPM** helps optimize resource allocation and risk management by pinpointing the longest sequence of dependent tasks,
 - Determine the shortest possible project timeline.

References

1. PMI. (2013). *A Guide to the Project Management Body of Knowledge* (5th ed.). PMI, Inc.
2. Kashyap, S. (2024, May 14). Articles: Ultimate guide on project scheduling: Know everything about it. Retrieved April 15, 2025, from ProofHub Website: <https://www.proofhub.com/articles/project-scheduling>.