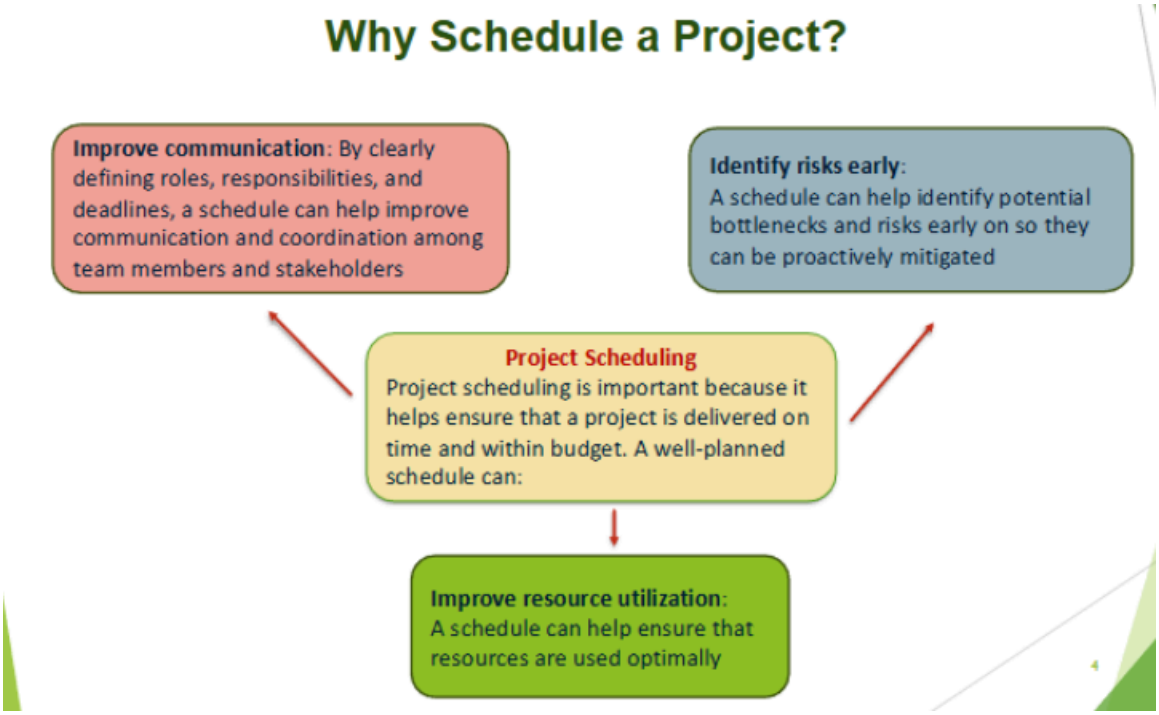
Lecture(6-10)



Work Breakdown Structure (WBS)

A **Work Breakdown Structure (WBS)** is a **hierarchical breakdown of the work, products, or outcomes** a project will deliver.

It acts as the project's **shopping list**, breaking the **main project product** into smaller parts.

A **work package** is the **lowest-level task** in the WBS.

Steps to Work Breakdown Structure (WBS)

1. Identify the **main deliverables** of the project.
2. Break deliverables into **smaller sub-deliverables**.
3. Continue breaking down until tasks are **manageable**.
4. Organise tasks into a **hierarchical structure**.
5. Assign **resources** and **estimate time** for each task.
6. **Review and refine** the WBS to ensure completeness.
7. Use the WBS as a **reference throughout the project**.

Gantt Charts

- **Definition:** Graphical representation of project tasks vs. time.
- **Structure:** Tasks → horizontal bars (Gantt bars).
- **Information Shown:** Start/end dates, dependencies, deadlines, progress, task owner.
- **Pros:** Visual, simple, tracks progress, helps resource planning.
- **Cons:** Hard to read for large schedules, less flexible for changes.

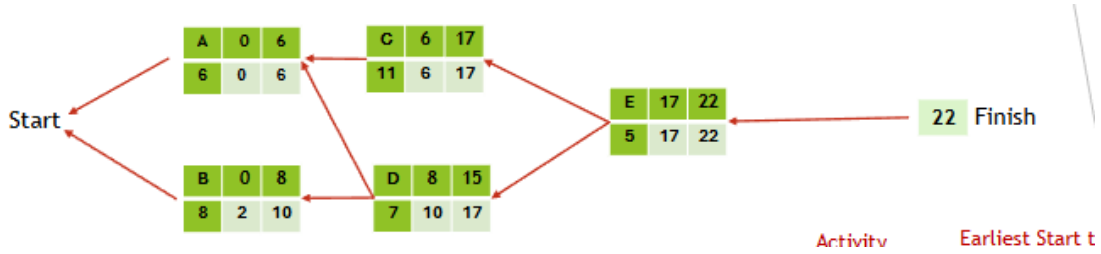
Critical Path Method (CPM)

- **Purpose:** Determine minimum project completion time; identify critical tasks.
- **Critical Path:** Longest sequence of dependent tasks; zero slack.
- **Float/Slack:** Time a task can be delayed without affecting project deadlines.

CPM Calculation

1. **Forward Pass:** Calculate ES (Earliest Start) & EF (Earliest Finish).
   - EF = ES + t
   - ES = latest EF of all predecessors
2. **Backward Pass:** Calculate LS (Latest Start) & LF (Latest Finish).
   - LS = LF - t
   - LF = earliest LS of all successors
3. **Slack/Float:**
   - Slack = LS - ES = LF - EF
4. **Critical Activities:** Tasks with zero slack.

Activity	A	B	C	D	E
Immediate predecessor	___	___	A	A/B	C/D
Expected in time weeks	6	8	11	7	5



Project Risk Management

Project Constraints and Risk

- **Risk Definition:** The chance of something happening (positive or negative), potential harm, loss, or unexpected outcomes.
- **Project Risk:** Any uncertainty that may positively or negatively impact achieving project objectives.
- **Types of Risk:**
  1. **Negative Risk (Threats):** Potential problems that can impede project success.
     - Management approach: Minimize potential negative risks, similar to insurance or investment.
  2. **Positive Risk (Opportunities):** Risks that lead to beneficial outcomes.
     - Management approach: Maximize potential positive risks.

Risk Management Concept

- **Definition:** Combination of art (judgment, perspective) and science (numerical approach) to identify, analyze, and respond to risks.
- **Benefits:**
  - Improves project success.

- Helps in selecting good projects.
- Supports defining project scope.
- Aids in developing realistic estimates for planning.

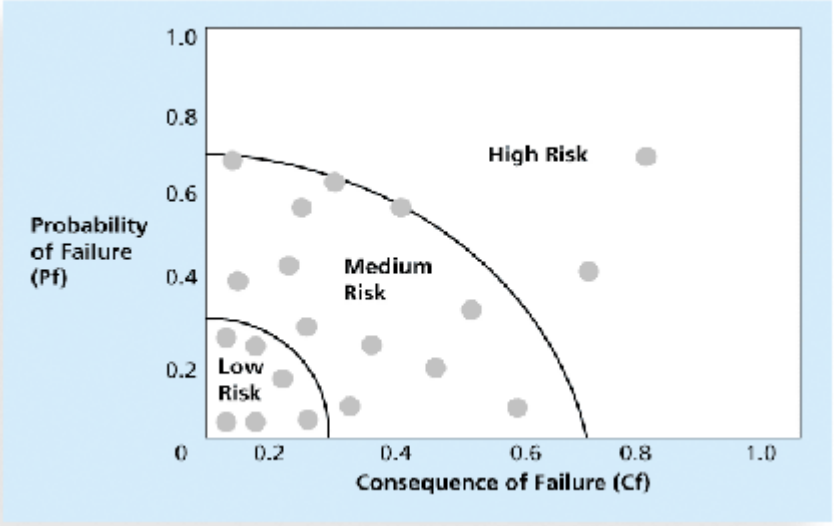
Risk Categorization (PESTLE Analysis)

- **Purpose:** Categorize external environmental factors affecting a project.

Category	Description / Examples
Political	Competition policy, government spending, tax policies, industry regulations
Economical	Interest rates, exchange rates, consumer spending, income trends
Social	Demographic changes, consumer tastes, lifestyle changes
Technological	Disruptive tech, new production methods, mobile adoption
Legal	Employment law, health & safety, environmental legislation, minimum/living wage
Ethical/Environmental	Ethical sourcing in supply chain, tax practices, pollution, carbon emissions (sustainability)

Risk Analysis and Documentation

1. **Probability/Impact Analysis:**
   - Uses a matrix to evaluate risks based on probability of occurrence and impact if it happens.
   - Risks are rated High, Medium, or Low for probability and impact.
   - Overall risk level (Severity) = Probability × Impact.
   - Risk Factors quantify overall risk based on likelihood and consequences.



2. **Risk Register:**
    - **Definition:** Primary document from the risk identification process.
    - **Purpose:** Records potential risks, mitigation actions, responsible owners, and status updates.

    Risk events refer to specific, uncertain events that may occur to thedetriment or enhancement of the project

Project Cost Management

Importance of Cost Management

Effective cost management throughout a project's lifecycle is critical because it:

1. Provides the project sponsor with a true picture of progress.
2. Reduces financial and operational risk.
3. Identifies areas of inefficiency.
4. Supplies valuable data for future planning and decision-making.

Keeping Track of Costs

1. **Labor Costs:**
    - Assign financial value to hours recorded on timesheets.
    - Calculated from the worker's salary, broken down to an hourly rate.
2. **Overhead Costs:**

- Include heating, lighting, office space, equipment, and other indirect costs.

3. **Timesheets:**
   - Mechanism for allocating labor and overhead costs to the correct client or cost center.
4. **Expenditure Control:**
   - Project managers must understand reasons for any unforeseen costs before authorizing payments.

Earned Value Management (EVM)

- **Definition:** Integrates project scope, cost, and schedule to assess performance.
- **Purpose:** Measures project accomplishments against money spent and time consumed.

Key EVM Metrics

Acronym	Full Name	Definition	Also Known As
PV	Planned Value	Authorized budget assigned to scheduled work	BCWS (Budgeted Cost of Work Scheduled)
EV	Earned Value	Work performed expressed in budget terms	BCWP (Budgeted Cost of Work Performed)
AC	Actual Cost	Realized cost for work performed during a period	ACWP (Actual Cost of Work Performed)

Variance Calculations (Performance Measurement)

Metric	Formula	Interpretation
Cost Variance (CV)	CV = EV - AC	CV > 0: Under budget; CV < 0: Over budget; CV = 0: On budget
Schedule Variance (SV)	SV = EV - PV	SV > 0: Ahead of schedule; SV < 0: Behind schedule; SV = 0: On schedule

Metric	Formula	Interpretation
Cost Variance %	CV% = (CV / EV) × 100	Relative measure of budget performance
Schedule Variance %	SV% = (SV / PV) × 100	Relative measure of schedule performance

Software Development Methodologies (SDM)

Software Methodology

- **Definition:** A framework to structure, plan, and control the development of an information system.

Software Development Lifecycle (SDLC) Stages

1. **Feasibility Study:** Evaluate if the project can be completed on time, budget, and with available staff; includes ‘buy vs build’ decisions.
2. **Requirements Analysis:** Complete specification including business rules, security, use cases, and sample designs; customer approval required.
3. **Design:** Map requirements to logical and physical designs; reviewed by developers, architects, and customers.
4. **Implementation:** Programmers write code, reuse existing code where possible; unit testing and demos begin.
5. **Testing:** Ensure system works properly via multiple layers (Unit, Integration, Load, UAT).
6. **Deploy:** Release software to live production; phased rollout for complex systems.
7. **Operate/Maintenance:** Support operations including patches, fixes, and maintenance.

Software Development Methodologies

1. **Waterfall (Linear-Sequential):**
    - Phases completed sequentially; no overlap.
    - Simple, easy to understand; suited for engineering design.
2. **Iterative (Incremental):**

- Starts with a subset of requirements; enhanced iteratively until complete.
- Repeated cycles, adding design changes and new features.
- Suited for large systems or when future changes are expected.

3. **Agile:**
   - Focuses on incremental, iterative development, quick delivery, and collaboration.
   - Short-term cycles; frequent working software delivery.
   - **Values:** People & interactions over processes; customer satisfaction via continuous delivery.
   - **Scrum:** Agile framework promoting self-organizing teams, continuous learning, and improvement.

Types of Testing

Test Type	Definition	Purpose	Example
Unit Test	Test individual components	Validate each unit works as intended	Fee payment function
Integration Test	Test integrated modules	Expose defects between modules	POS system integrated with Warehouse system
Load Test	Examine system under normal/high load	Ensure system handles high user demand	Online ticket sales system for a concert
User Acceptance Test (UAT)	Final stage before go-live	Ensure software works in real-world scenarios	Students testing management system modules