

## Lecture - 1

### What are CASE Tools? (software that helps automate SDLC steps)

-> Case tools are software applications that help the system analysts, designer and developers automate different steps of the systems development Life cycle (SDLC) - Like analysis, design, documentation and coding.

এগুলোর মাধ্যমে কাজের গতি বাড়ে এবং ভুল কম হয়।

**Example of CASE Tools:** 1. Diagramming Tools, 2. Analysis Tools, 3. Central Repository, 4. Documentation Generators, 5. Code Generators.

### Benefits of Case Tools:

1. Increases Productivity and speed.
2. Reduces human error
3. Improves documentation quality
4. Makes project management easier

### eXtreme Programming:

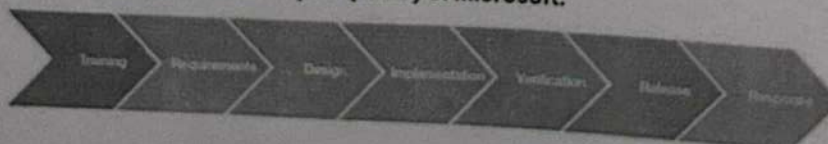
eXtreme Programming (XP) is an agile software development methodology that focuses on improving software quality and responding quickly to changing customer requirements.

এখানে ছোট ছোট ধাপে কাজ করা হয়, যেখানে কোড লেখা, টেস্ট করা এবং কাস্টমারের ফিডব্যাক একসাথে নেওয়া হয়।

### Benefits of eXtreme Programming:

1. *Short incremental Development Cycles:* work is done in small and improved continuously.  
(বড় প্রজেক্টকে ছোট ছোট অংশে ভাগ করে অল্প সময়ে তৈরি ও পরিষ্কার করা হয়)
2. *Automated Testing:* Tests are written before or during coding to ensure quality.  
(Coding & Testing একসাথে করা হয়)
3. *Pair Programming:* Two programmers work together on one computer - one writes code, the other reviews it instantly.
4. *Continuous Feedback:* Developers, managers & customers regularly review the software and give feedback. (নিয়মিত যাচাই)

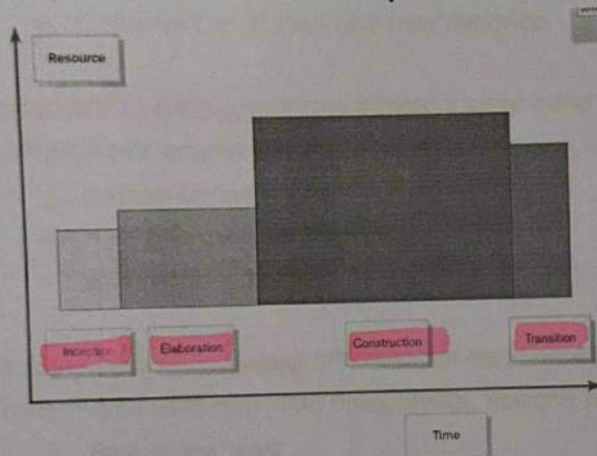
### Systems development life cycle (SDLC) of Microsoft:



### Criticisms of SDLC

1. Forcing Timed phases on intangible processes (analysis & design) is doomed to fail.  
SDLC প্রতিটি ধাপকে নির্দিষ্ট সময়ের মধ্যে শেষ করার চাপ দেয়। কিন্তু *analysis* ও *design* হচ্ছে চিন্তাশীল ও পরিবর্তনশীল ধাপ — সময়ের জোরে এগুলো করলে কাজের মান নষ্ট হয় বা প্রকল্প ব্যর্থ হয়।
2. Too much formal process and documentation slows things down  
SDLC-তে অনেক বেশি আনুষ্ঠানিক ধাপ ও ডকুমেন্টেশনের প্রয়োজন হয়, ফলে পুরো কাজের গতি ধীর হয়ে যায় এবং বাস্তবায়ন জটিল হয়।
3. Cycles are not necessarily waterfalls  
বাস্তবে সিস্টেম ডেভেলপমেন্ট একদম ধাপে ধাপে হয় না; অনেক সময় ধাপগুলো একসাথে চলে বা পুনরাবৃত্তি হয়। কিন্তু SDLC-তে এই ধরনের *flexibility* দেয় না।

### Phases of OOSAD-based development or (Rational Unified Process) :



## Lecture 2

### (Scenario based comparison question)

#### Six Sources of Software:

1. **Information technology service firms:** এই ধরনের কোম্পানিগুলো ক্লায়েন্টের প্রয়োজন অনুযায়ী **custom software** তৈরি করে দেয়। তারা সাধারণত বড় প্রজেক্টে কাজ করে। যেমন: Infosys, Accenture, Tata Consultancy Services (TCS)

- Custom-made software for each client
- Client pays for design, development & maintenance
- Long-term contracts and professional project management

Shah Jalal

## Lecture 1 & 2

### System Development Life Cycle (SDLC)

Planning -> Analysis -> Design -> Implementation -> Maintenance

**Functional Requirements:** These Define how the system should do. They describe the main features and function of the system.

Examples:

- 1) Users can log in & log out.
- 2) The system should have customer data.
- 3) The app should send a confirmation email after registration.
- 4) Students can view their exam results.

**Non-Functional Requirements:** These define how the system should perform. They describe the quality, performance and constraints of the system.

Examples:

- 1) Performance Requirements: The system should load within 3 seconds.
- 2) Security: Data must be secure.
- 3) Scalability: The system should handle 10,000 users at once.
- 4) Compatibility: The application must be compatible with windows, macOS and Linux operating systems.

### Agile Methodology Key Principle:

- 1) Adaptive rather than Predictive.
- 2) Emphasize people rather than roles.
- 3) Self adaptive process for flexible and self correcting.

### When to use methodology? (Agile)

**Changing Requirements:** Unpredictable or dynamic requirements.

**Faster delivery:** Need to deliver quickly and regularly.

**Customer Involvement:** Customers who understand the process and will get involved.

### Rational Unified Process

Rational Unified Processes are **four** phases.

**Inception:** Define the project scope, vision & feasibility. **(Planning)**

**Elaboration:** Analyze the problem domain in detail and establish the project architecture. **(Analysing phase)**

**Construction:** Build the software product based on the established architecture. **(Coding & Testing)**

**Transition:** Deliver the software to users and ensure it meets their needs. **(Deploy and**

maintenance)

### RFP (Request for proposal)

-> যখন Company নতুন software নিতে চায়, তারা vendors এর কাছে RFP পাঠায়।

### Software Re-Use:

আগের বানানো Software বা কোড নতুন প্রজেক্টে ব্যবহার করাই - ReUse.

### There is Two types of Re-use:

- 1) Object Oriented Development - একই Class/Object Project এ use.....
- 2) Component Based Development - Use small ready\_made components.

### Benefits of Re-use:

1. Time saving
2. Less Error
3. Less Cost

### 3 Steps of Re-Use:

1. **Abstraction** -> Reusable code design
2. **Storage** -> Save in Library or Repository
3. **Recontextualization** -> Understandable Code design যেনো অন্য Developer-রা বুঝতে পারে।

### Approaches of Re-Use:

Approach	Meaning
Ad-hoc	Developer নিজের ইচ্ছা মত Re-use করবে।
Facilitated	Company encourage করে Re-use এর জন্য।
Managed	Company officially Re-use system বাধ্যতামূলক রাখে।
Designed	শুরু থেকেই Re-use এর জন্য বানানো হয়।

### Why Selecting off-the-shelf-software:

- **Cost:** কম খরচে দ্রুত সফটওয়্যার সমাধান পাওয়া যায়।
- **Functionality:** প্রয়োজনীয় ফিচারগুলো আগে থেকেই প্রস্তুত থাকে।
- **Vendor Support:** সমস্যা হলে কোম্পানি থেকে সহায়তা পাওয়া যায়।
- **Vendor Viability:** দীর্ঘমেয়াদে বিক্রয়তা টিকে থাকবে কিনা তা নিশ্চিত হওয়া যায়।
- **Flexibility:** সফটওয়্যার কিছুটা নিজের প্রয়োজন অনুযায়ী পরিবর্তন করা যায়।
- **Documentation:** ব্যবহারের স্পষ্ট নির্দেশনা ও গাইড থাকে।

**2. Packaged software providers:** এরা এমন সফটওয়্যার তৈরি করে যা অনেক ইউজার একসাথে ব্যবহার করতে পারে। এগুলোকে "off-the-shelf software" বলে। যেমন: Microsoft Office, Adobe Photoshop, QuickBooks

- Ready-made, mass produced
- Same version for all customers
- Installed locally on computers

**3. Vendors of enterprise-wide solution software:** এরা বড় প্রতিষ্ঠানের জন্য **integrated system** সরবরাহ করে যা বিভিন্ন বিভাগ (HR, Finance, Inventory) একসাথে যুক্ত করে। যেমন: SAP ERP, Oracle E-Business Suite, Microsoft Dynamics

- Covers the whole organization
- Central database
- Complex installation and maintenance

**4. Cloud Computing:** সফটওয়্যার ইন্টারনেটের মাধ্যমে ব্যবহার করা হয় (Software as a Service - SaaS)। ইউজারকে ইনস্টল করতে হয় না। যেমন: Google Workspace, Salesforce, Dropbox, Zoom

- Hosted Online
- Pay-per-use or subscription
- Accessible anywhere

**5. Open-source software:** এমন সফটওয়্যার যার **source code** সবার জন্য উন্মুক্ত — যে কেউ দেখতে, পরিবর্তন করতে, বা উন্নত করতে পারে। যেমন: Linux, MySQL, Apache, Mozilla Firefox

- Free or low-cost
- Community-based development
- Transparent source code

**6. In-house Development:** প্রতিষ্ঠানটি নিজের টিম ও প্রোগ্রামার দিয়ে নিজেই সফটওয়্যার তৈরি করে। যেমন: Banks, telecom companies, or universities developing their own internal management systems.

- Developed internally
- Controlled entirely by the organization
- Best suited for company-specific needs

Comparison Table ( Based on six sources software )

Source	Customization	Cost	Time	Example	Best For
IT Service Firms	Very high	High	Long	Infosys	Unique business needs
Packaged Software	Low	Low	Short	MS Office	General users
Enterprise Vendors	Medium-high	Very high	Long	SAP ERP	Large Organization
Cloud Computing	Medium	Medium	Short	Salesforce	Flexible online access
Open source	High	Free/Low	Medium	Linux	Tech-savvy users
In-house Dev.	Very high	High	Long	Bank system	Specialized internal needs

Ranking of these 6 sources of software: যেটা বেস্কী ভালো

1. Cloud computing
2. Open-source software
3. Packaged software
4. IT service firms
5. In-house development
6. Vendor of Enterprise-wide solution software

### Outsourcing

Outsourcing means hiring another company or outside experts to perform services or build systems that your organization would otherwise do itself.

**Outsourcing** হচ্ছে এমন একটি প্রক্রিয়া যেখানে কোনো প্রতিষ্ঠান নিজের কাজ নিজে না করে, বাইরের কোম্পানি বা বিশেষজ্ঞদের দিয়ে সেই কাজ করিয়ে নেয়।

যেমন – কোনো ব্যাংক নিজে সফটওয়্যার না বানিয়ে সেটা অন্য IT কোম্পানিকে দিয়ে তৈরি করায়।

### Advantages of Outsource:

Advantage	Explanation
Cost saving	কম খরচে উচ্চ মানের কাজ
Faster delivery	দ্রুত কাজ শেষ
Access to experts	দক্ষ ও অভিজ্ঞ জনবল ব্যবহার
Focus on core business	মূল ব্যবসায় Focus দেওয়া যায়
Advanced technology	নতুন Tools ব্যবহারের সুযোগ

### Lecture 3



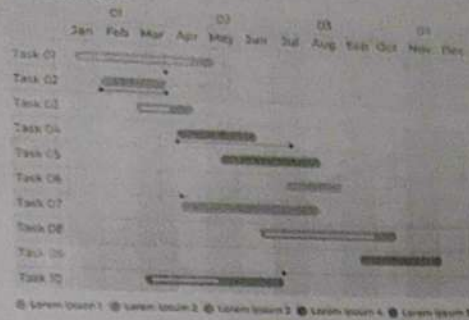
### Gantt Charts vs Network Diagrams

Criteria	Gantt chart	Network Diagram
Definition	A bar chart that shows the start and end dates of tasks along a timeline.	A diagram that shows relationships and sequence between project activities using nodes (circles) and arrows.
Representation	Uses horizontal bars to represent the time duration of each task.	Uses circles (nodes) for activities and arrows for their dependencies (sequence).
Focus	Focuses on time scheduling — when tasks start and finish.	Focuses on logical order and dependency — which task depends on which.
Purpose	To track project progress over time.	To analyze the workflow and sequence of tasks.
Complexity	Easy to read but becomes cluttered for large projects.	Better for complex projects to show inter-task relationships clearly.
Example	A Gantt chart may show Task A: 1–5 days, Task B: 6–10 days.	A Network Diagram shows Task A → Task B → Task C (dependency flow).

## Gantt Chart:

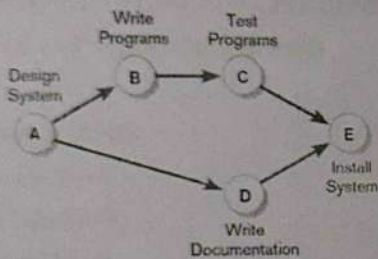
টাইমলাইন ভিত্তিক চার্ট — project এর প্রতিটি কাজ কত দিন লাগবে এবং কখন শুরু ও শেষ হবে তা দেখায়।

Gantt Chart

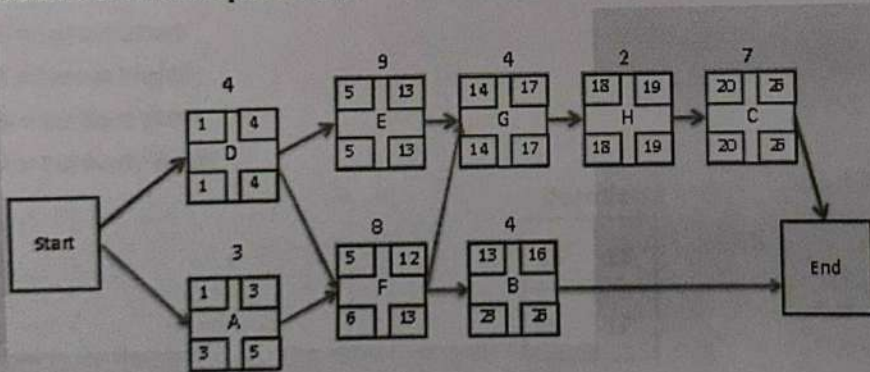


## Network Diagram:

কাজগুলোর relationship ও order দেখায় — কোন কাজ শেষ না হলে পরেরটা শুরু হবে না।



## Backward Forward pass Math

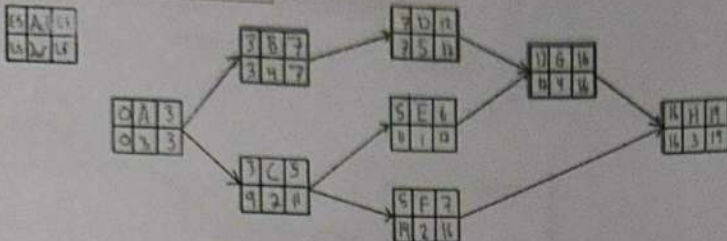
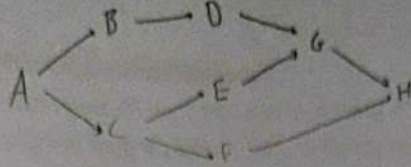


$$EF = ES + \text{duration} - 1 \text{ (For forward pass)}$$

$$LS = LF - \text{duration} + 1 \text{ (For backward pass)}$$

\*Collected from University slide.

Activity	Predecessor	Duration (days)
A	-	3
B	A	4
C	A	2
D	B	5
E	C	1
F	C	2
G	D, E	4
H	F, G	3



\*Collected from: <https://youtu.be/4oDLMs11Exs?si=N40-mepzrGh26UuA>

Here,

A → B → D → G → H = Critical Path

C → E → F = Non Critical Path

**Slack = LS - ES**

$$C = (11 - 5) = 6$$

$$E = (12 - 6) = 6$$

$$F = (16 - 7) = 9$$

**LS = Latest Start**

**LF = Latest Finish**

**ES = Earliest Start**

**EF = Earliest Finish**

**Duration**

ES	EF
LS	LF

**Critical path : EF = LF**

- > **Backward pass** এর সময় ছোটো সংখ্যা নিবো
- > **Forward Pass** এর সময় বড় সংখ্যা নিবো

**ES/EF (Forward pass এর মাধ্যমে বের হবে)**

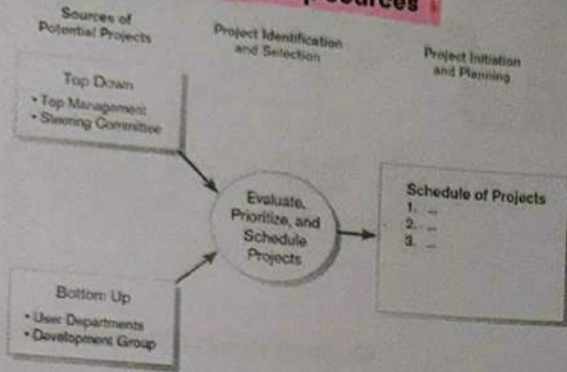
**LS/LF (Backward pass এর মাধ্যমে বের হবে)**

## Lecture 4

### Identifying and Selecting IS Development Projects

Information System (IS) Development Project মানে হলো এমন প্রজেক্ট যা একটি প্রতিষ্ঠানের তথ্য ব্যবস্থাপনাকে উন্নত করতে তৈরি করা হয় — যেমন নতুন সফটওয়্যার বানানো, ডেটাবেস তৈরি, বা সিস্টেম আপগ্রেড করা।

### Top Down and Bottom-Up Sources



Type	Description	Example
Top-Down planning	Top-down source are projects identified by top management or by a diverse steering committee. (প্রজেক্টের আইডিয়া আসে Top Management বা Steering Committee থেকে)	CEO বলে, "একটা নতুন ERP সিস্টেম দরকার।"
Bottom-Up planning	Bottom-up source ideas come from manager, department or development group workers.	HR টিম বলে, "একটা attendance tracking অ্যাপ দরকার।"

### Selecting IS Development Projects (প্রজেক্ট নির্বাচন)

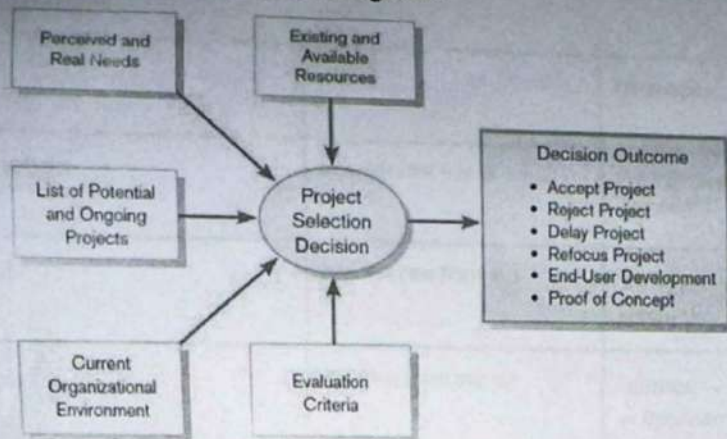
অর্থ:

বিভিন্ন প্রজেক্ট আইডিয়ার মধ্যে থেকে এমন প্রজেক্ট নির্বাচন করা যা প্রতিষ্ঠানকে বেশি লাভ দেবে এবং কৌশলের (strategy) সাথে মিলবে।

বেছে নেওয়ার মানদণ্ড:

- কৌশলের সাথে মিল আছে কি না (Strategic fit)
- খরচ ও লাভ (Cost-benefit ratio)
- ঝুঁকি (Risk)
- সময় (Time)
- সম্পদের প্রাপ্যতা (Resources)

### Project Selection Decision Diagram:



### Weighted Multicriteria Analysis

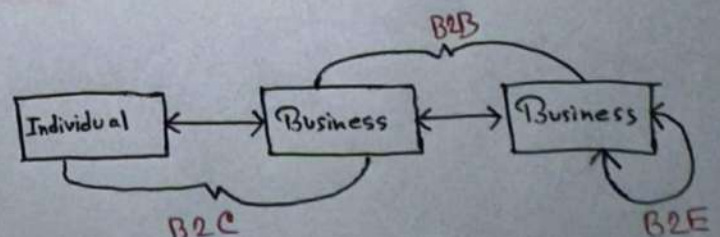
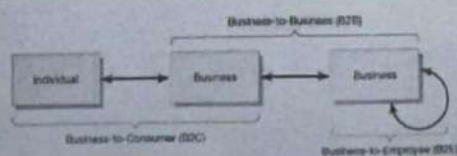
$$\text{Score} = \sum (\text{Weight} \times \text{Rating})$$

Criteria	Weight	Alternative A		Alternative B		Alternative C	
		Rating	Score	Rating	Score	Rating	Score
Requirements							
Real-time data entry	18	5	90	5	90	5	90
Automatic reorder	18	1	18	5	90	5	90
Real-time data query	14	1	14	5	70	5	70
	50		122		250		250
Constraints							
Developer costs	15	4	60	5	75	3	45
Hardware costs	15	4	60	4	60	3	45
Operating costs	15	5	75	1	15	5	75
Ease of training	5	5	25	3	15	3	15
	50		220		165		180
Total	100		342		415		430

Highest total score → selected project.

### Electronic Commerce (E-Commerce)

Type	Meaning	Example
B2C	Business → Customer	Amazon, Daraz
B2B	Business → Business	Dell ↔ Intel
B2E	Business → Employees	HR portal, payroll system



### Types of Planning Matrices:

Matrix Type	সম্পর্ক	Example	Exam এ যেসকল Scenario based Question আসতে পারে
Location to Function	কোন স্থানে কোন কাজ হয়	ঢাকা ব্রাঞ্চে "Sales" হয়, চট্টগ্রামে "Support"	প্রশ্নে যদি "Which branch performs which activity?" থাকে → Location-to-Function
Location to Unit	কোন স্থানে কোন বিভাগ আছে	হেড অফিসে HR, ফ্যাক্টরিতে Production	যদি "Which department is at which site?" বলা থাকে → Location-to-Unit
Unit to Function	কোন বিভাগ কোন কাজ করে	Finance → Budgeting, HR → Recruitment	যদি "Which department performs which function?" → Unit-to-Function
Function to Objective	কোন কাজ কোন লক্ষ্য পূরণে সহায়তা করে	Sales → Increase revenue	যদি "Which function supports which business goal?" → Function-to-Objective
Function to Process	কোন কাজের জন্য কোন প্রক্রিয়া লাগে	Sales → Order Entry, Invoice Process	যদি "Which processes are part of this function?" → Function-to-Process
Function to Data entity	কোন ফাংশন কোন ডেটা ব্যবহার করে	Payroll uses Employee Data	যদি "Which data entity does this function use?" → Function-to-Data Entity
Process to Data entity	কোন প্রক্রিয়া কোন ডেটা ব্যবহার করে	Order Process → Order Table, Customer Table	যদি "Which data are needed for this process?" → Process-to-Data Entity
Process to Information system	কোন প্রক্রিয়া কোন সফটওয়্যার বা সিস্টেমে চলে	Order Entry → Sales System	যদি "Which system supports this process?" → Process-to-Information System
Data entity to information system	কোন ডেটা কোন সিস্টেমে ব্যবহৃত হয়	Customer Table → CRM System	যদি "Which system uses this data entity?" → Data Entity-to-Information System
Information system to objective	কোন সিস্টেম কোন লক্ষ্য অর্জনে সাহায্য করে	ERP System → Increase efficiency	যদি "Which information system helps achieve this objective?" → Information System-to-Objective

Scenario question-এ সবসময় keyword খুঁজো (যেমন process, system, data, function, objective, unit, location)।

**Tips :** দুইটি keyword-কে match করলেই সঠিক matrix-type ধরতে পারবে

## Lecture 5

### Assessing Project Feasibility

#### Definition:

Feasibility assessment means determining whether a proposed information system project is possible, practical, and worthwhile for the organization.

Feasibility মানে হলো — কোনো প্রস্তাবিত সিস্টেম বাস্তবে তৈরি করা সম্ভব কি না, প্রতিষ্ঠানটির জন্য লাভজনক কি না, এবং তা প্রযুক্তিগতভাবে বাস্তবসম্মত কি না, তা মূল্যায়ন করা।

#### Purpose:

To evaluate if the project should continue, be modified, or be rejected before investing large resources.

#### Main types of Feasibility

Type	Explanation	Bangla
Economic Feasibility	Checks cost vs. benefit (is it financially worthwhile?).	খরচ ও লাভ বিশ্লেষণ করা হয়।
Technical Feasibility	Can we build it with existing technology & skills?	প্রয়োজনীয় প্রযুক্তি ও দক্ষতা আছে কি না দেখা হয়।
Operational Feasibility	Will it work well and be accepted by users?	সিস্টেমটি বাস্তবে চলবে ও ইউজাররা গ্রহণ করবে কি না।
Scheduling Feasibility	Can it be finished on time?	নির্দিষ্ট সময়সীমায় শেষ করা সম্ভব কি না।
Legal & Contractual Feasibility	Any legal or policy restrictions?	কোনো আইনি বা চুক্তিগত সমস্যা আছে কি না।
Political Feasibility	Will management and employees support it?	কর্তৃপক্ষ ও কর্মচারীরা সমর্থন করবে কি না।

- **Response Time:** সহায়তা পেতে বা সমস্যা সমাধানে দ্রুত প্রতিক্রিয়া পাওয়া যায়।
- **Ease of installation:** সহজে ও দ্রুত সিস্টেমে ইনস্টল করা যায়।

### Lecture 3

#### Core concepts:

1. **Project:** নির্দিষ্ট লক্ষ্য পূরণের জন্য শুরু ও শেষসহ পরিকল্পিত কাজ।

2. **Project Management:** প্রকল্প শুরু, পরিকল্পনা, বাস্তবায়ন ও সমাপ্তির নিয়ন্ত্রিত প্রক্রিয়া।

- > **Controlled Process** • **Initiating** → Relationship with customer / Project management environment
- > **Planning** → Describe Project scope / Risk Analysis
- > **Executing** → Execute baseline Project Plan / Manage changes / Maintain Project workbook
- > **Closing** → Close / conduct reviews / close contract

3. **Project Manager:** নেতৃত্ব, প্রযুক্তি, দ্বন্দ্ব সমাধান ও স্টেকহোল্ডার ব্যবস্থাপনায় দক্ষ ব্যক্তি।

4. **Deliverable:** SDLC-এর প্রতিটি ধাপ শেষে তৈরি হওয়া চূড়ান্ত Software.

5. **System Service Request (SSR):** সিস্টেম উন্নয়নের অনুরোধ বা প্রস্তাব জানানোর মানসম্মত ফর্ম।

6. **Feasibility Study:** Economic + Operational viability check (প্রস্তাবিত সিস্টেমটি অর্থনৈতিক ও কার্যক্রমগতভাবে বাস্তবসম্মত কিনা তা যাচাইয়ের বিশ্লেষণ)

#### Project Manager এর কাজ -

- Customer and Management Expectations (গ্রাহক ও ব্যবস্থাপনার প্রত্যাশা):  
গ্রাহক ও কর্তৃপক্ষের চাহিদা ও প্রত্যাশা বুঝে তা পূরণ করা।
- Documentation and Communication (লিখিত ও যোগাযোগ):  
প্রকল্প সম্পর্কিত তথ্য সঠিকভাবে লিখিত করা এবং দলের মধ্যে কার্যকর যোগাযোগ বজায় রাখা।
- Time and Resource Constraints (সময় ও সম্পদ সীমাবদ্ধতা):  
নির্দিষ্ট সময় ও সীমিত সম্পদ ব্যবহার করে কাজ সম্পন্ন করা।
- Methodologies and Tools (পদ্ধতি ও সরঞ্জাম):  
সঠিক প্রজেক্ট ম্যানেজমেন্ট টুলস ও কৌশল (যেমন Agile, Scrum, Gantt Chart) ব্যবহার করা।
- Managing People (মানুষ ব্যবস্থাপনা):  
টিমের সদস্যদের কাজ সমন্বয় করা, অনুপ্রাণিত করা এবং দ্বন্দ্ব সমাধান করা।
- Contractors and Vendors (ঠিকাদার ও সরবরাহকারী ব্যবস্থাপনা):  
বাহ্যিক পার্টনার বা সরবরাহকারীদের সঙ্গে সমন্বয় রাখা।
- Organizational Change and Complexity (প্রতিষ্ঠানের পরিবর্তন ও জটিলতা):  
সংগঠনের পরিবর্তন বা জটিল পরিস্থিতির সঙ্গে খাপ খাওয়ানো।

- **Systems Development Life Cycle** (সিস্টেম ডেভেলপমেন্ট লাইফ সাইকেল):  
প্রকল্পের বিভিন্ন ধাপ (যেমন পরিকল্পনা, বিশ্লেষণ, ডিজাইন, বাস্তবায়ন ও রক্ষণাবেক্ষণ) পরিচালনা করা।
- **Technological Change** (প্রযুক্তিসম্পন্ন পরিবর্তন):  
নতুন প্রযুক্তির সঙ্গে প্রকল্পকে মানিয়ে নেওয়া ও প্রয়োগ করা।

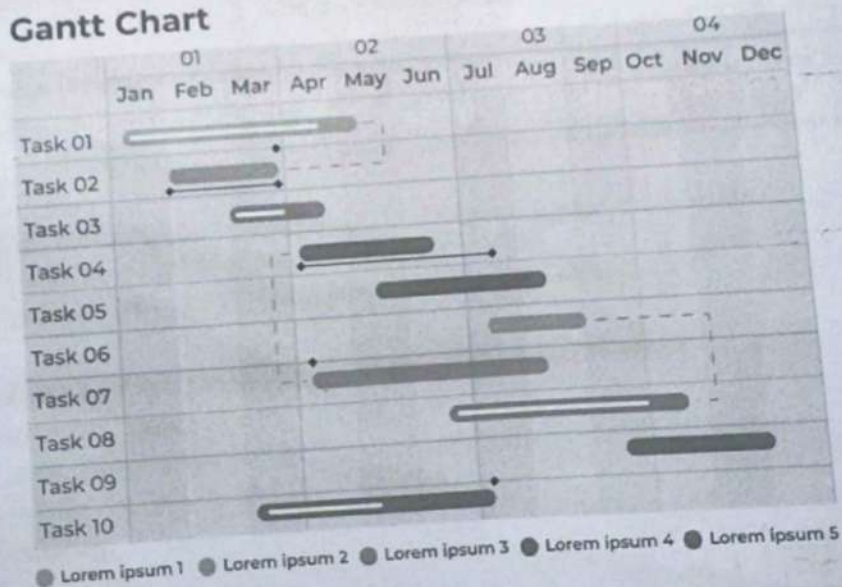
#### Phases of Project Management Process

- 1. Initiation** - প্রকল্প শুরু ও অনুমোদনের ধাপ
- 2. Planning** - প্রকল্পের সবকিছু কিতাবে সম্পন্ন হবে তার পরিকল্পনা তৈরি
- 3. Execution** - পরিকল্পনাকে বাস্তব কাজের মাধ্যমে কার্যকর করা
- 4. Closedown** - Closedown ধাপে Project Completion (প্রকল্পের কাজ সম্পন্ন), Evaluation (মূল্যায়ন যাচাই), Team Release (টিম মুক্তি), Documentation (রিপোর্ট সংরক্ষণ) ও Celebration and Acknowledgment (সফলতায় মূল্যায়ন) করা হয়
  - > **Natural:** successful completion or unnatural termination (time/money).
  - > **Actions:** close project, post-project review (প্রকল্প-পরবর্তী পর্যালোচনা), close customer contract (গ্রাহকের সঙ্গে চুক্তি সমাপ্ত করা).

#### Gantt Chart

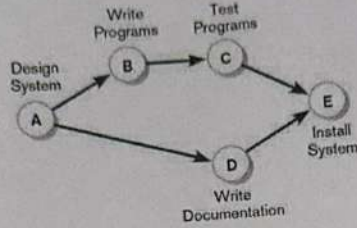
"কখন" কোল কাজ হবে (সময়সূচি তিরিক চার্ট)।

#### Gantt Chart



## Network Diagram

"কোন কাজের পর কোন কাজ হবে" (dependency ভিত্তিক চার্ট)।



Step	Work	Explanation
A	Design System	প্রকল্পের প্রথম ধাপ, সিস্টেম ডিজাইন করা।
B	Write programs	সিস্টেম ডিজাইন শেষ হলে প্রোগ্রাম লেখা শুরু হয়।
C	Test Programs	প্রোগ্রাম লেখা শেষ হলে তা পরীক্ষা করা হয়।
D	Write Documentation	একই সময় বা পরবর্তীতে ডকুমেন্টেশন লেখা হয়।
E	Install System	টেস্ট ও ডকুমেন্টেশন শেষ হলে সিস্টেম ইনস্টল করা হয়।

ভীরচিত্রগুলো দেখাচ্ছে,

- $A \rightarrow B \rightarrow C \rightarrow E$  একটি মূল পথ (main workflow)।
- অন্যদিকে  $A \rightarrow D \rightarrow E$  একটি বিকল্প বা সমান্তরাল পথ (parallel task)।

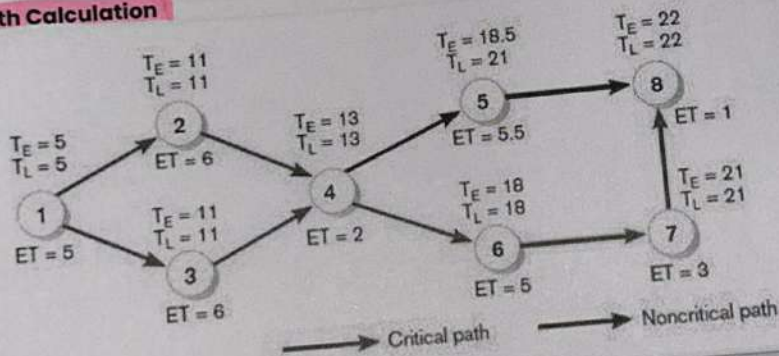
**PERT Calculations Depend upon the following three time estimates**

Time Estimate 3 Types

1. Optimistic (Least)
2. Pessimistic (Greatest)
3. Realistic

ACTIVITY	TIME ESTIMATE (in weeks)			EXPECTED TIME (ET) $\frac{o + 4r + p}{6}$
	o	r	p	
1. Requirements Collection	1	5	9	5
2. Screen Design	5	6	7	6
3. Report Design	3	6	9	6
4. Database Design	1	2	3	2
5. User Documentation	2	6	7	5.5
6. Programming	4	5	6	5
7. Testing	1	3	5	3
8. Installation	1	1	1	1

### Path Calculation



ACTIVITY	$T_E$	$T_L$	SLACK $T_L - T_E$	ON CRITICAL PATH
1	5	5	0	✓
2	11	11	0	✓
3	11	11	0	✓
4	13	13	0	✓
5	18.5	21	2.5	✓
6	18	18	0	✓
7	21	21	0	✓
8	22	22	0	✓

**Slack** → বাড়তি সময়

উদাহরণ: উপরের ফিগারে আমরা দেখতে পাচ্ছি (Activity or Node 5 →  $T_L = 21$ ,  $T_E = 18.5$ )

তাহলে  $Slack = (21 - 18.5) = 2.5$  Days

মানে, Activity 5 Project এর সময়সীমার মধ্যে 2.5 দিন দেড়ি করলেও প্রজেক্ট ঠিক সময়ে শেষ করা যাবে।

**Critical Path** → যে কাজের Slack = 0, তারা Project এর Critical Path এ আছে কারন তাদের কোনো দেড়ি হলে প্রজেক্টও দেড়ি হবে।

**TE (Forward pass** করলে **TE** পাওয়া যায়)

**TL (Backward pass** করলে **TL** পাওয়া যায়)

Symbol	Meaning	Bangla Explanation
ET	Elapsed Time	এই কাজ করতে যত সময় লাগে
TE (When Forward Pass)	Earliest Time	যত দ্রুত কাজটা শেষ করা যায়
TL (When Backward Pass)	Latest Time	যত দেড়িতে শেষ করলেও প্রজেক্ট Ready হতে দেড়ি হবে না
SLACK	$TL - TE$	কাজে দেড়ি করার যতটুকু সুযোগ আছে
Critical Path	$Slack = 0$	এই কাজ দেড়ি করলে প্রজেক্ট হইতে ও দেড়ি হবে
Non Critical or Flexible path	$Slack > 0$	এই কাজ দেড়ি করার সুযোগ আছে।