

## Introduction:

System Analysis and design course intends to help students understand its importance in developing systems that meet the users' requirements, conform to the operational standards of an organization, and therefore achieve the goal of the organization.

System analysis and design examines carefully what a system has to do and how to do them.

## Course outline:

### Chapter I: Fundamentals of Information System

- Introduction to Information System
- Components of Information System
- Elements of a system
- Roles of Information System in an organization
- Types of Information Systems

## Course outline (con.):

### Chapter II: Software metrics and quality assurance

- Use of software metrics
- Lines of Code (LOC)
- Algorithmic Cost Model—COCOMO
- Quality Management
- Quality Assurance team
- Quality Review

## Course outline (con.):

### Chapter III: System Analysis methods

- What is System Analysis?
- Structured System Analysis method
- System prototype method
- Models of System Development
- System Development Life Cycle (SDLC)
- Structured System Development Cycle
- Prototyping
- Types of prototyping tools
- Advantages and disadvantages prototyping method

## Course outline (con.):

### Chapter IV: System development methods

- Structured Analysis, Design, and Programming
- Object-Oriented Analysis and Design
- Joined Application Development
- Rapid Application Development

### Chapter V: Information System Planning

- Introduction
- Reasons for new information system
- Factors affecting information system
- Project scheduling methods
- Procedure for planning a project

## Course outline (con.):

### Chapter VI: Process Analysis and Modeling

- Introduction
- Data Flow Diagram (DFD)
- Data Flow symbols
- What is a good DFD?
- Decision tree & Decision table
- Physical & Logical level diagrams
- English Structured and Algorithm chart
- Data Modeling

## Course outline (con.):

### Chapter VII: User Interface Design

- Introduction
- Good interface
- Interactive interface
- Web design

## Assignment:

During this course, students are required to develop the following systems:

1. Games (students' choices to develop any games they like)
2. Website visitors trend analysis (Chart)
3. Online Forum
4. Sound manipulator
5. Admin system
6. Web page editor

The assignment will do in group. Each group has from 5 to 7 students. A different assignment topic is assigned to each group. Each group's presentation will be held on a date and time specified by the lecturer of the class.

## Chapter I: Fundamentals of Information System

### Objectives:

- Defining Information System
- Defining the need of Information System in improving business performance
- Defining the elements of a System
- Defining the types of Information System



## Chapter I: Fundamentals of Information System

### 1. Introduction to Information System

Information System is referred to a set of organized components working together to collect, store, process, and generate outputs ( data and information) to achieve the goal of the organization. Therefore, Information System is a system that provides information to the organization so than it can improve its business performance to stay alive in competitive world. Below are examples of Information Systems that commonly used in a business organization:

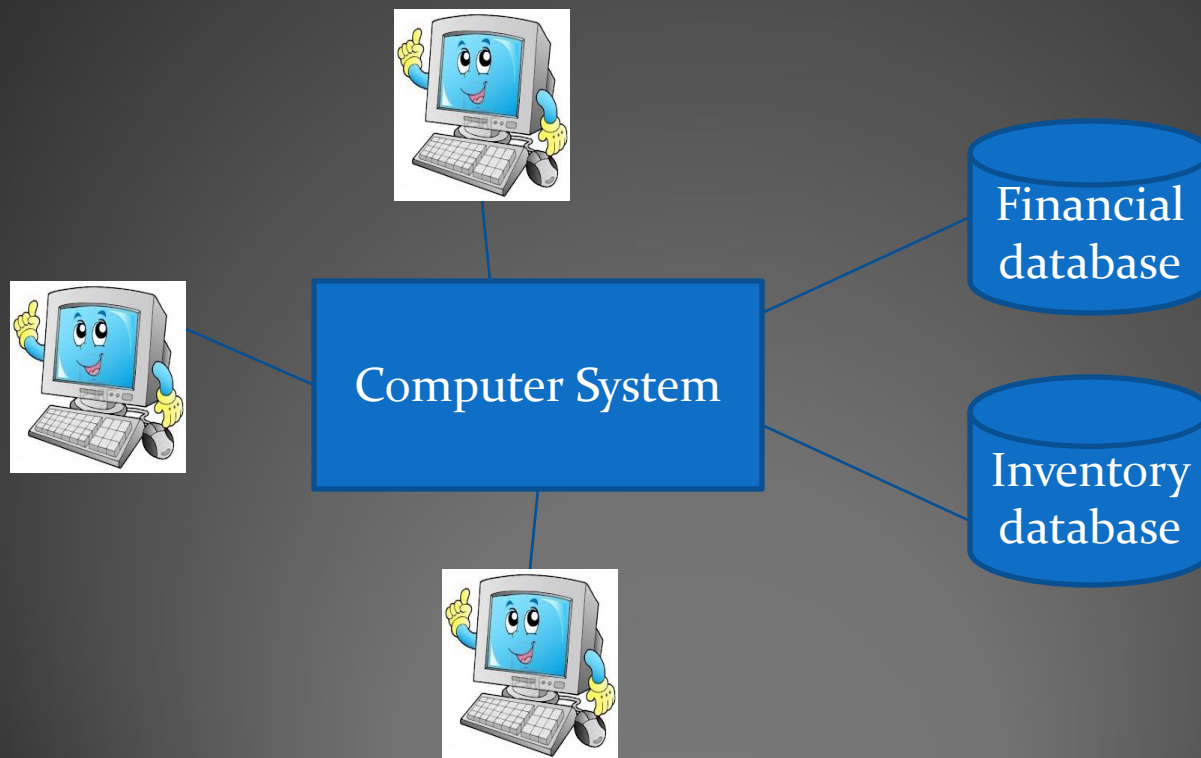
*-Payroll System*

*-Inventory System*

*-Admin System*

## Chapter I: Fundamentals of Information System

### 1. Introduction to Information System



## Chapter I: Fundamentals of Information System

### 1. Introduction to Information System

#### =>Data vs. Information

**Data** are raw facts that represent something. For example, name of employee, salary, position, etc.

Data are transformed to information by adding extra values to the raw facts.

#### Data:

Chan Thorn  
1000\$  
Sale manager

#### Information:

Employee name:  
Chan Thorn  
Salary:  
1000\$  
Position:  
Sale manager

## Chapter I: Fundamentals of Information System

### 1. Introduction to Information System

=>Information Technology vs. Information System

Technologies are used to create information systems.

#### Information Technology:

- Hardware
- Software
- Database
- Programming
- Network
- Other s related

#### Information Systems:

- Payroll system
- Inventory system
- Admin system
- Hotel reservation system
- Accounting system

## Chapter I: Fundamentals of Information System

### 2. Components of Information System

#### 2.1 People

- People produce information
- People use information in making decisions regard to their work activities
- E.g. A bank teller may use the computer to check the account balance before the withdrawal is allowed.

#### 2.2 Process

- Processes determine what is to be done with the input data in to the system.
- E.g. A process to define how a withdrawal is made for a customer of the bank

## Chapter I: Fundamentals of Information System

### 2. Components of Information System

#### 2.3 Equipment

-Equipments are used to store data, compute, and transmit it.

-E.g. computers, disk drives, input devices, communication devices, etc.

## Chapter I: Fundamentals of Information System

### 3. Elements of a system

#### 3.1 Input & Output

-A system feeds on input to produce output. It is much like a business brings in human, financial, and material resources to produce goods and services.

-When the output is not met the goal, the feedback will be performed to adjust the system.

#### 3.2 Processor (s)

-A processor transforms input to output.

-When the output changes, the process changes

## Chapter I: Fundamentals of Information System

### 3. Elements of a system

#### 3.3 Control

- Making decisions
- Controlling the pattern of activities
- In an organization, management is the decision making body. They guide and control the activities in the organization.

#### 3.4 Feedback

- Feedback measures the output whether it meets the goal of system.
- It measures the output against the performance standards.



## Chapter I: Fundamentals of Information System

### 3. Elements of a system

#### 3.4 Feedback

- Feedback can be positive and negative.
- Positive feedback reinforce the input , process, and control.
- Negative feedback calls for changes in input , process, and control.
- The system analysts may be informed about the performance of the new installed system that needs enhancements.

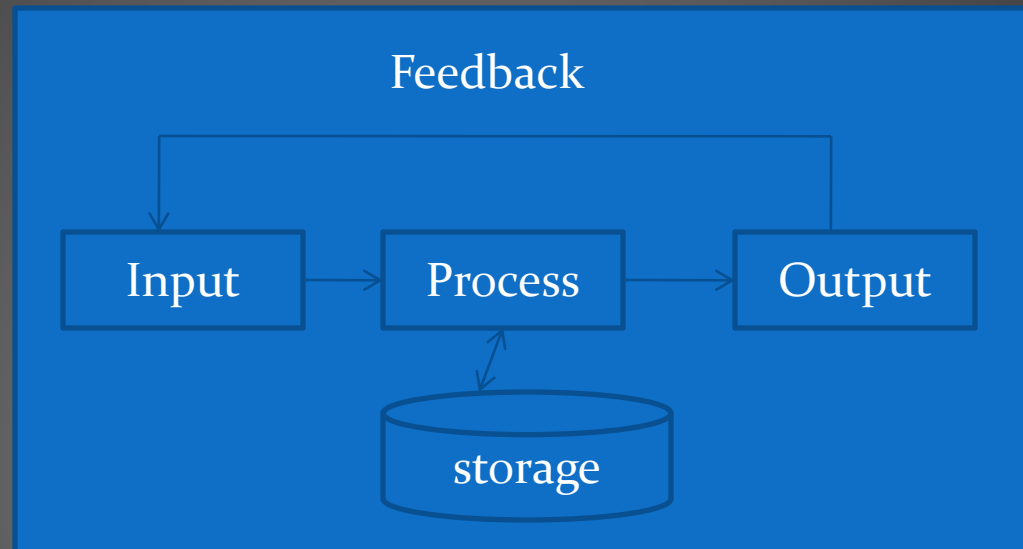
## Chapter I: Fundamentals of Information System

### 3. Elements of a system

#### 3.5 Environment

**System environments** are things outside the system boundary. They can effect behavior of the system.

#### Environment



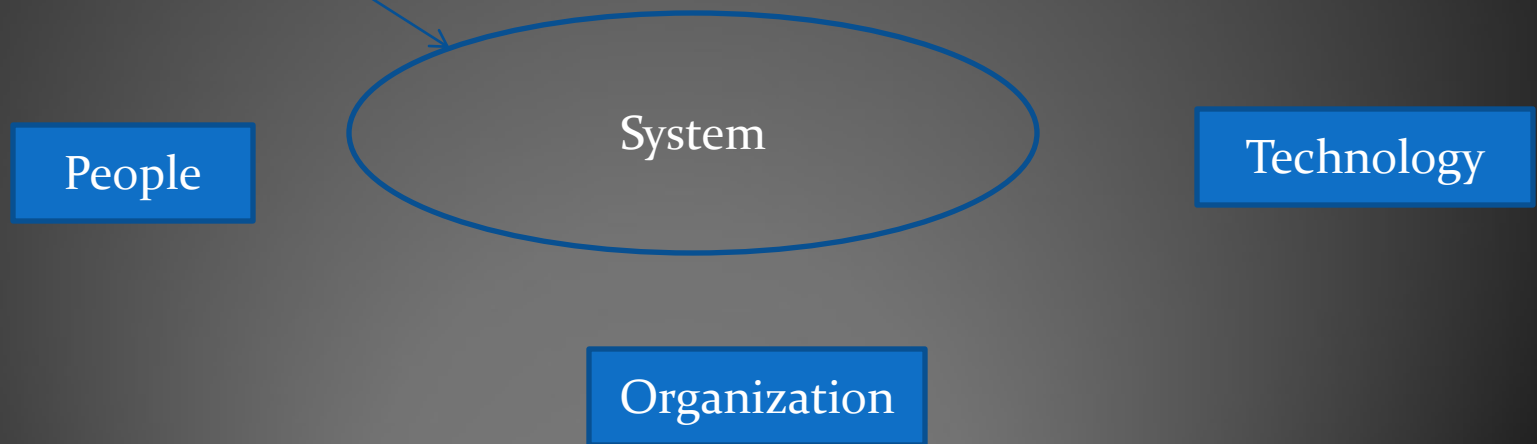
## Chapter I: Fundamentals of Information System

### 3. Elements of a system

#### 3.5 Environment

**System environments** are things outside the system boundary. They can effect behavior of the system.

Environment boundary



## Chapter I: Fundamentals of Information System

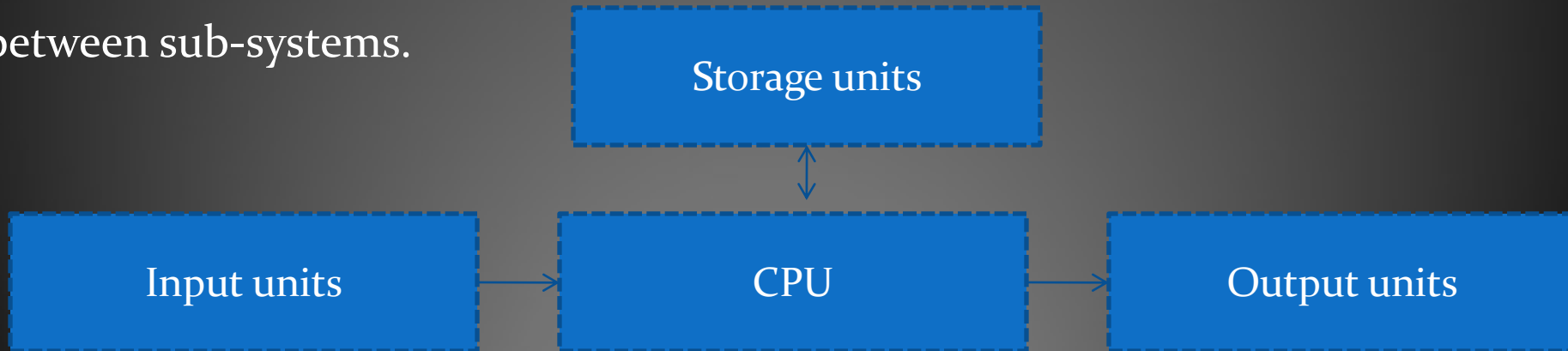
### 3. Elements of a system

#### 3.6 Boundary & Interfaces

-Boundary defines what are in the system and what are not in the system.

-E.g. Input units, CPU, and output units are confined in the computer system boundary.

--Interfaces are about interconnections and interactions between sub-systems.



## Chapter I: Fundamentals of Information System

### 4. Types of systems

#### 4.1 Physical system

- A physical system contains tangible components.
- A computer contains keyboard, cables, hard disks, monitor, etc.
- A school has buildings, teachers, students, textbooks, etc.

#### 4.2 Abstract system

- It is a conceptual model that defines the components and their relationships in the system.
- It can represent the real system.

## Chapter I: Fundamentals of Information System

### 4. Types of systems

#### 4.3 Open system

- Interacts freely with system environment by getting input and produce the output.
- System environment effects the system.
- Information Systems and originations are Open System.

#### 4.4 Characteristics of Open System

- Input from outside: system environment is the source of input for the system.
- Process, output, & cycle: An open system produces output and operates in cycles.
- Specialization: system components and functions tend to be specialized and differentiated.
- Goal: a system has may interrelated sub-systems that work

## Chapter I: Fundamentals of Information System

### 4. Types of systems

together to achieve the goal of the system.

#### -4.4 Closed system

- Doesn't interact with environment.
- Environment can't effect the system.
- Closed systems are rare in business world.

## Chapter I: Fundamentals of Information System

### 5. Roles of Information in business

- Gain competitive advantage: by innovative use of IS, allow the management of a business firm to gain strategic advantage over competitors.
- Support decision making: right information at the right time can be used to make strategic plan to push organization forward.
- Support business processes: business processes and operations can be improved with the use of IS.



# Quiz

## Chapter II: Software metrics and quality assurance

### 1. The use of software metrics

- A metric is a quantitative measure the degree to which a software, or process met the quality objective.
- By gathering quantitative information, we can identify the root causes, inefficiencies, and opportunity to improve the quality of a system or process.
- It can provide lessons to learn. Then we can build a better system in the future.
- It can be a useful source of information to predict qualities of systems or processes.
- The cost and schedule of future projects can be estimated.
- E.g. number of errors per thousand lines of code

## Chapter II: Software metrics and quality assurance

### 2. Lines of Code (LOC)

- LOC is used to count number lines in a program to determine its size.
- Each line has an end of line character.
- Lines of code can vary from one coding style to another or from one programming language to another.
- Logical (LLOC) vs. physical (LINES) lines of code
- LLOC doesn't count blank and comment lines.
- LINES does include blank and comment lines.
- CCCC is a free tool to for lines of C or C++ code counting.

## Chapter II: Software metrics and quality assurance

### 2. Algorithmic Cost Model-COCOMO II

-COCOMO II is a model used to estimate the cost and schedule of a program or software.

-It uses Effort Equation to estimate Person-months required in project development.

$$\text{-Effort} = 2.94 * \text{EAF} * (\text{KSLOC})^E$$

Where EAF is the Effort Adjustment Factor derived from Cost Drivers, KSLOC is the logical lines of code divided by 1000, and E is the Effort multiplier derived from Scale Drivers.

-**Cost Drivers** are factors that determine the effort required to complete the project.

-E.g. required software reliability, database size, software complexity, execute time constraint, analysis, and design ability, etc.

## Chapter II: Software metrics and quality assurance

### 2. Algorithmic Cost Model-COCOMO II

-**Scale Drivers** are factors that affect project duration and cost :  
precedentedness, development flexibility, risk resolution, team cohesion, process maturity, etc.

-**Precedentedness**: experience in working with related software, understanding of project objectives, need for innovative data processing procedures, algorithms, etc.

## Chapter II: Software metrics and quality assurance

### 2. Algorithmic Cost Model-COCOMO II

- Development flexibility:** premium on early completion, need for conformance with external application interfaces, etc.
- Risk resolution:** risk management plan, tool support available for resolving risks, number of risk items, etc.
- Team cohesion:** difficulties in reconciling different stakeholders' objectives, difficulties, stakeholders' lack of experience in working in group, etc.
- Process maturity:** percentage of compliance with Key Process Area (KPA)
- 18 Key Process Areas:** 1. Requirements Management, 2. Software Project Planning, 3. Software Project Tracking and Oversight, 4. Software Subcontract Management,

## Chapter II: Software metrics and quality assurance

### 2. Algorithmic Cost Model-COCOMO II

5. Software Quality Assurance, 6. Software Configuration Management,
7. Organization Process Focus, 8. Organization Process Definition,
9. Training Program, 10. Integrated Software Management
11. Software Product Engineering, 12. Intergroup Coordination, 13. Peer reviews,
14. Quantitative Process Management , 15. Software Quality Management,
16. Defect Prevention,
17. Technology Change Management , 18. Process Change Management

-**Schedule equation:**  $\text{Duration} = 3.67 * (\text{Effort})^{\text{SE}}$

Where Effort is the result of Effort Equation and SE is the schedule equation exponent derived from scale factor.

## Chapter II: Software metrics and quality assurance

### 3. Quality Assurance

- The goal of quality management is to institute and monitor quality assurance program.
- Quality Assurance plays as an important component in software development process.
- The quality assurance program includes :
  - +validation of the system against requirements;
  - +checking errors in design doc and system;
  - +checking for qualitative features such as portability and flexibility; and
  - +checking for usability



## Chapter II: Software metrics and quality assurance

### 3. Quality Assurance

#### 3.1. Quality Assurance team

- A quality assurance team is responsible for carrying out software reviews.
- They are skillful people.
- Exiting environment that fosters work responsive is a good sign for good software.
- The team consists of 3 to 7 people. Different roles are assigned team member.
- The roles should include a producer, a reader, an inspector, and a moderator.

## Chapter II: Software metrics and quality assurance

### 3. Quality Assurance

#### 3.1. Quality Assurance team

- The producer is the person who produced the Software product.
- The reader is the person who guide the inspector through product.
- The inspector (s) evaluates the product.
- There is also a moderator who control the review process.

## Chapter II: Software metrics and quality assurance

### 3.2 . Quality Reviews

- Each quality assurance program is made up of reviews.
- Reviews are to ensure that the task outputs match the requirements done early in system analysis.
- In a quality review, there are two important activities: **inspection**, and **walkthrough**.

#### 3.2 .1. Inspection

- The inspection is done to ensure quality of product. The inspection activities should include the following steps:
  - +Step 1: the producers of product will explain to the inspectors about their Products.

## Chapter II: Software metrics and quality assurance

### 3.2 . Quality Reviews

#### 3.2 .1. Inspection

+Step 2: the inspectors prepare documents for inspection.

+Step 3: a meeting between reader, moderator, and inspectors to go through the work is conducted.

+Step 4: Producers correct any deficiencies in the product.

+Step 5: the final step in the inspection is to make sure that the deficiencies are corrected by the producer.

## Chapter II: Software metrics and quality assurance

### 3.2 . Quality Reviews

#### 3.2 .2. Walkthrough

- Walkthrough is commonly used to find errors in the system.
- It checks the correctness of models produced by system analysis and design.
- Walkthrough has a specific checking procedure, team structure, and requires document to be produced during and after walkthrough.

## Chapter II: Software metrics and quality assurance

### 3.2 . Quality Reviews

#### 3.2 .2. Walkthrough

- The team must check that the models:
  - +meet system objectives
  - +is a correct presentation of the system
  - +is easy to understand
  - +has no omission or inaccuracies things

## Chapter II: Software metrics and quality assurance

### 3.2 . Quality Reviews

#### 3.2 .2. Walkthrough

-The walkthrough procedure will involve following DFD, describing data stores, or going through the logic of each process. Omissions and inaccuracies are recorded in Action list.

## Chapter II: Software metrics and quality assurance

### 3.2 . Quality Reviews

#### 3.2 .2. Walkthrough

- Two outcomes are possible from the walkthrough .
- One is no errors found in the models and hence they are accepted.
- The other outcome is where the errors detected the action list will be produced and the models are amended before submitting to another walkthrough.



## Chapter II: Software metrics and quality assurance

### 3.2 . Quality Reviews

#### 3.2 .2. Walkthrough

System Description:  
System Producer:

Walkthrough Action list  
Walkthrough date:  
Walkthrough Leader:

ISSUE	Responsible member	Amendment completed