

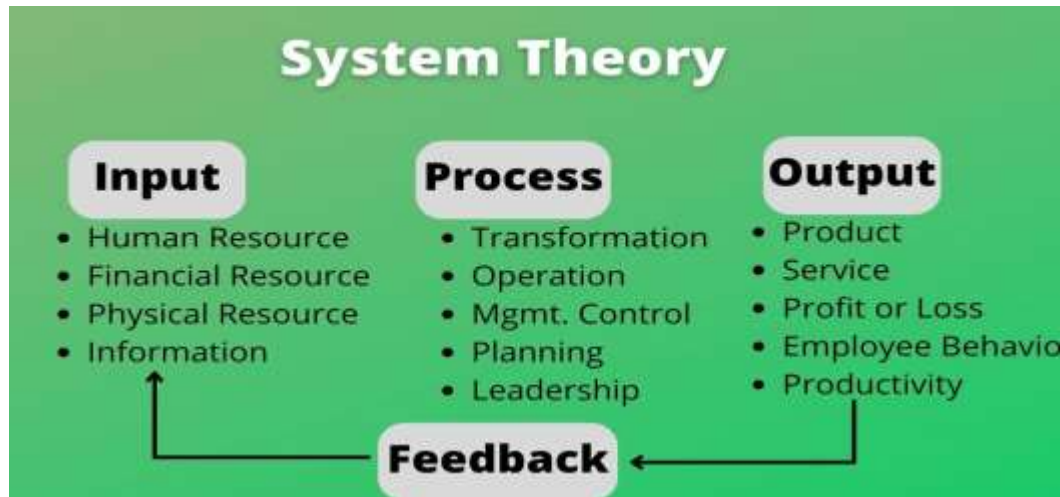
unit :1- System Analysis & Design,software Engineering & concept of Quality Assurance

- Definitions
- System Analyst
- SDLC
- Fact-finding technique
- Tools for documentation procedure
- Data flow analysis tool
- Software engineering
- Introduction to QA
- Quality control
- difference between QA & QC
- QA activity

Definition:

- **What is a System?**

- The word System is derived from Greek word Systema, which means an organized relationship between any set of components to achieve some common cause or objective.
- A system is “an orderly grouping of interdependent components linked together according to a plan to achieve a specific goal.”
- For example:
 - traffic management system,
 - payroll system,
 - automatic library system,
 - human resources information system.



Elements of a system Following are considered as the elements of a system in terms of Information systems:

1. Inputs and outputs
2. 2. Processor
3. 3. Control
4. 4. Environment
5. 5. Feedback
6. 6. Boundaries and interface

- Inputs and Outputs- Inputs are the elements that enter the system for processing and output is the result of processing.
- Processor- It is the element that involves the actual transformation of input into .
- Control- The control element guides the system.
- Feedback- Output is compared against performance standards.
- Environment- It is the super system within which an organization operates.
- Boundaries and Interface- A system should be defined by its limit.

Types of systems Systems are classified in different ways:

1. Physical or abstract systems.
2. Open or closed systems.
3. 'Man-made' information systems.
4. Formal information systems.
5. Informal information systems.
6. Computer-based information systems.
7. Real-time system.

Characteristics of a System:

- 1) organization (order)
- 2) interaction
- 3) interdependence
- 4) integration
- 5) a central
- 6) A central objective.

What is sub-System?

- A group of related components that are part of a larger system.
- A subsystem is a single, predefined operating environment through which the system coordinates the work flow and resource use. The system can contain several subsystems, all operating independently of each other. Subsystems manage resources. All jobs, with the exception of system jobs, run within subsystems.
- For example, a disk subsystem is a part of a computer system. A bus is a part of the computer.

What do you mean by business system?

- A business system is a documented procedure that outlines how to do something in your organization to achieve your business goals.
- A business system is a group of interdependent elements or tasks that meet a business objective. You should document every step for each business process or activity. Business activities consistently produce results. Strategizing how you complete these tasks can help streamline daily operations.
- Examples of business systems:
 - Customer Relationship Management (CRM) Systems,
 - Enterprise Resource Planning (ERP) Systems,
 - Quality Management Systems (QMS),
 - Inventory Management Systems.

What do you mean by information system?

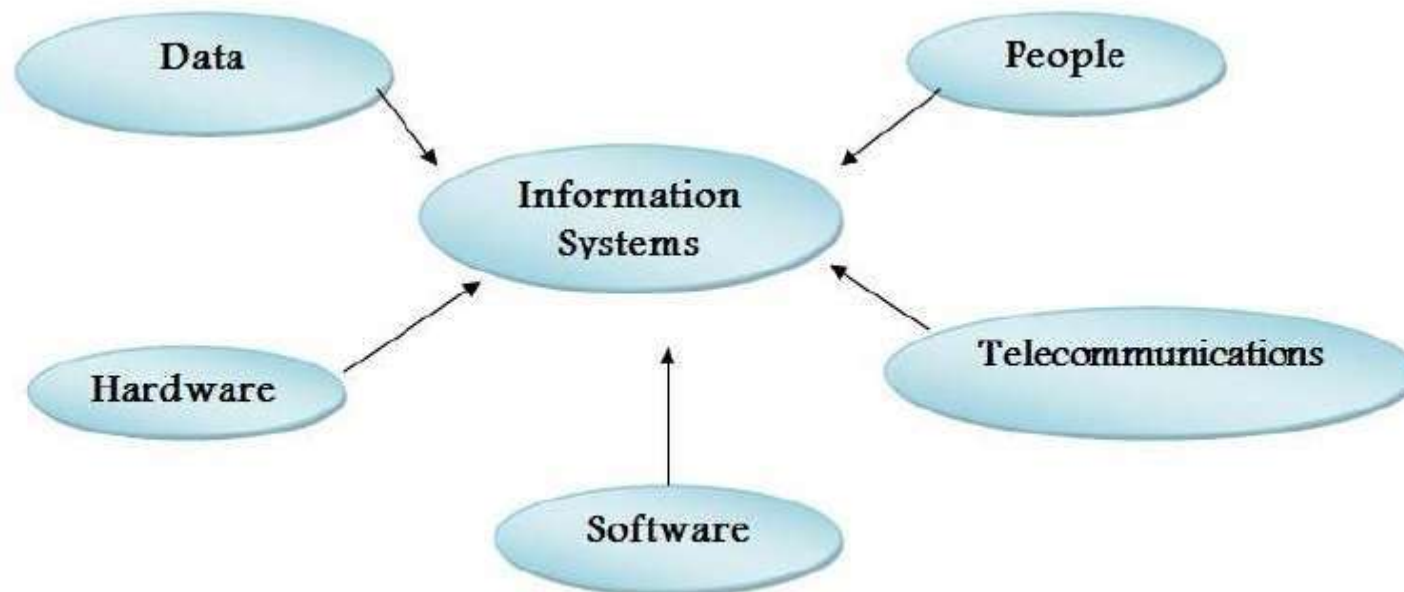
- An information system is a combination of software, hardware, and telecommunication networks to collect useful data, especially in an organisation.
- Many businesses use information technology to complete and manage their operations, interact with their consumers, and stay ahead of their competition. Some companies today are completely built on information technology, like eBay, Amazon, Alibaba, and Google.

The basic components of computer based information system are:

- Hardware- these are the devices like the monitor, processor, printer and keyboard, all of which work together to accept, process, show data and information.
- Software- is the programs that allow the hardware to process the data.
- Databases- are the gathering of associated files or tables containing related data.
- Networks- are a connecting system that allows diverse computers to distribute resources.
- Procedures- are the commands for combining the components above to process information and produce the preferred output.

Information System can be divided into 4 major categories, which are:

1. Transaction Processing System (TPS)
2. Management Information System (MIS)
3. Decision Support System (DSS)
4. Executive Support System (ESS)



System Analyst

- An individual in charge of designing, modifying, or analysing various systems to ensure compatibility and user effectiveness. System analysts may work independently, but are often part of a larger information technology unit that keeps technical aspects of a company running smoothly.
- System analysts do not focus on hard-coding, but more on evaluating the code and making suggestions.
- System analyst is a person who conducts the system study & identifies the activities & objectives, formulate a set of rules to achieve the objects.

His main activities are

- Problem identification, which is a crucial part.
- Data gathering
- develop a solution for the problem
- Co-ordinate to develop all possible solutions.
- Develop plans to meet management objectives
- System Designing
- Co-ordinate the testing procedures.

He should have following Qualities.

1. Knowledge of people
2. Knowledge of Business functions
3. Knowledge of data processing principles
4. Ability to communicate
5. Flexibility
6. Analytical Mind
7. Well educated with sharp mind

Roles / Responsibilities

Role of System Analyst differs from organization to organization.

Most common responsibilities of System Analyst are following

1) System analysis

It includes system's study in order to get facts about business activity. It is about getting information and determining requirements. Here the responsibility includes only requirement determination, not the design of the system.

2) System analysis and design:

Here apart from the analysis work, Analyst is also responsible for the designing of the new system/application.

3) Systems analysis, design, and programming:

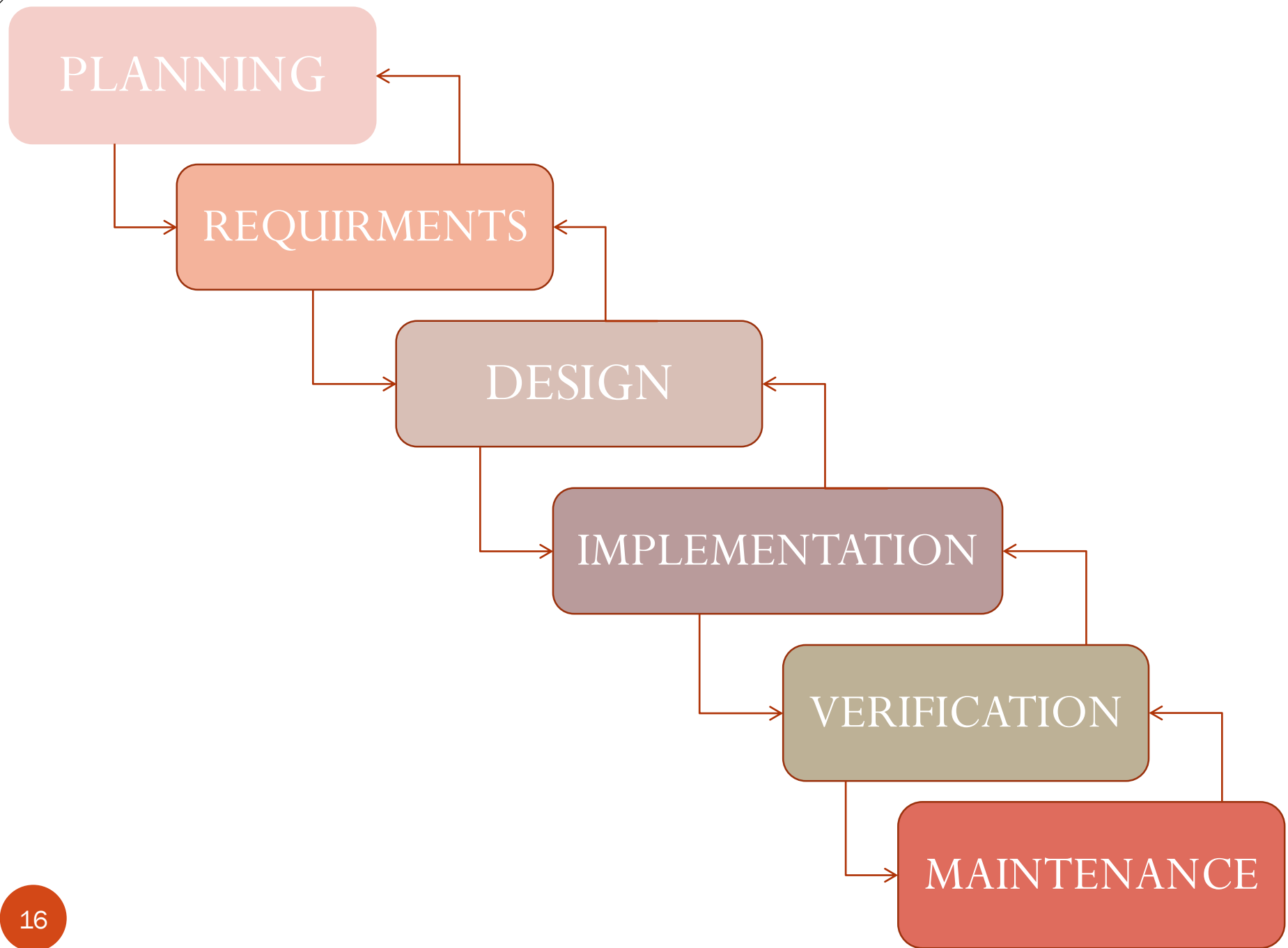
➤ Here Analyst is also required to perform as a programmer, where he actually writes the code to implement the design of the proposed application.

➤ Due to the various responsibilities that a system analyst requires to handle, he has to be multifaceted person with varied skills required at various stages of the life cycle. In addition to the technical know-how of the information system development a system analyst should also have the following knowledge.

- **Business knowledge:** As the analyst might have to develop any kind of a business system, he should be familiar with the general functioning of all kind of businesses.
- **Interpersonal skills:** Such skills are required at variou the users and extracting the requirements out of them
- **Problem solving skills:** A system analyst should have enough problem solving skills for defining the alternate solutions to the system and also for the development process.

SDLC: Software Development Life Cycle

- Software Development Life Cycle is the process used in a project to develop software product.
- It describes how the development activities will be performed and how development phases follow each other.
- The project manager decides the SDLC to use in a project during project planning phase. Because every project is unique, the project manager will usually need to tailor a standard SDLC to fit the nature and needs of the project.
- Phase of SDLC:



What are the phases of a software project?

- The main phases are:
- Planning: In this phase the requirements are examined from high level so the resources of the organization are allocated accordingly. Example of resources are software tools, hardware, and team with needed skills.
- Requirements (analysis): In this phase the problem of the domain is studied and analysed and a Software Requirements Specification (SRS) is developed to describe the expected behaviour of the system.
- Design: The purpose of software design is described how the technology will be used to provide solution for the problem specified during the requirements analysis phase. The design phase is usually carried in two phases:

- Architecture and high level design: In this phase technologies are specified and the main software elements.
- Detailed design: at which the algorithms are specified along with the necessary details to transform the design into code in the next phase.
- Implementation (coding): In this phase the design is transformed into software.
- Verification (testing): The purpose of software verification is to ensure that the software is being correctly built as specified by specifications.
- Operation and Maintenance: In this phase the software product is deployed and support is provided to the customer to correct issues or to add enhancements.

Fact Finding Techniques

- To study any system the analyst needs to do collect facts and all relevant information.
- The facts when expressed in quantitative form are termed as data. The success of any project is depended upon the accuracy of available data.
- Accurate information can be collected with help of certain methods/ techniques. These specific methods for finding information of the system are termed as fact finding techniques. **Interview, Questionnaire, Record View and Observations** are the different fact finding techniques used by the analyst. The analyst may use more than one technique for investigation.

INTERVIEW

- This method is used to collect the information from groups or individuals.
- Analyst selects the people who are related with the system for the interview. In this method the analyst sits face to face with the people and records their responses. The interviewer must plan in advance the type of questions he/ she is going to ask and should be ready to answer any type of question. He should also choose a suitable place and time which will be comfortable for the respondent.
- The information collected is quite accurate and reliable as the interviewer can clear and cross check the doubts there itself. This method also helps gap the areas of misunderstandings and help to discuss about the future problems. Structured and unstructured are the two sub categories of Interview.

Structured interview is more formal interview where fixed questions are asked and specific information is collected whereas unstructured interview is more or less like a casual conversation where in-depth areas topics are covered and other information apart from the topic may also be obtained.

Questionnaire

- It is the technique used to extract information from number of people. This method can be adopted and use only by a skilful analyst.
- The Questionnaire consists of series of questions framed together in logical manner The questions are simple, clear and to the point. This method is very useful for attaining information from people who are concerned with the usage of the system and who are living in different countries.

- The questionnaire can be mailed or send to people by post. This is the cheapest source of fact finding.

Record View

- The information related to the system is published in the sources like newspapers, magazines, journals, documented. This record review helps the analyst to get valuable information about the system and the organization.

Observation

- Unlike the other fact finding techniques, in this method the analyst himself visits the organization and observes and understand the flow of documents, working of the existing system, the users of the system etc.
- For this method to be adopted it takes an analyst to perform this job as he knows which points should be noticed and highlighted. In analyst may observe the unwanted things as well and simply cause delay in the development of the new system.

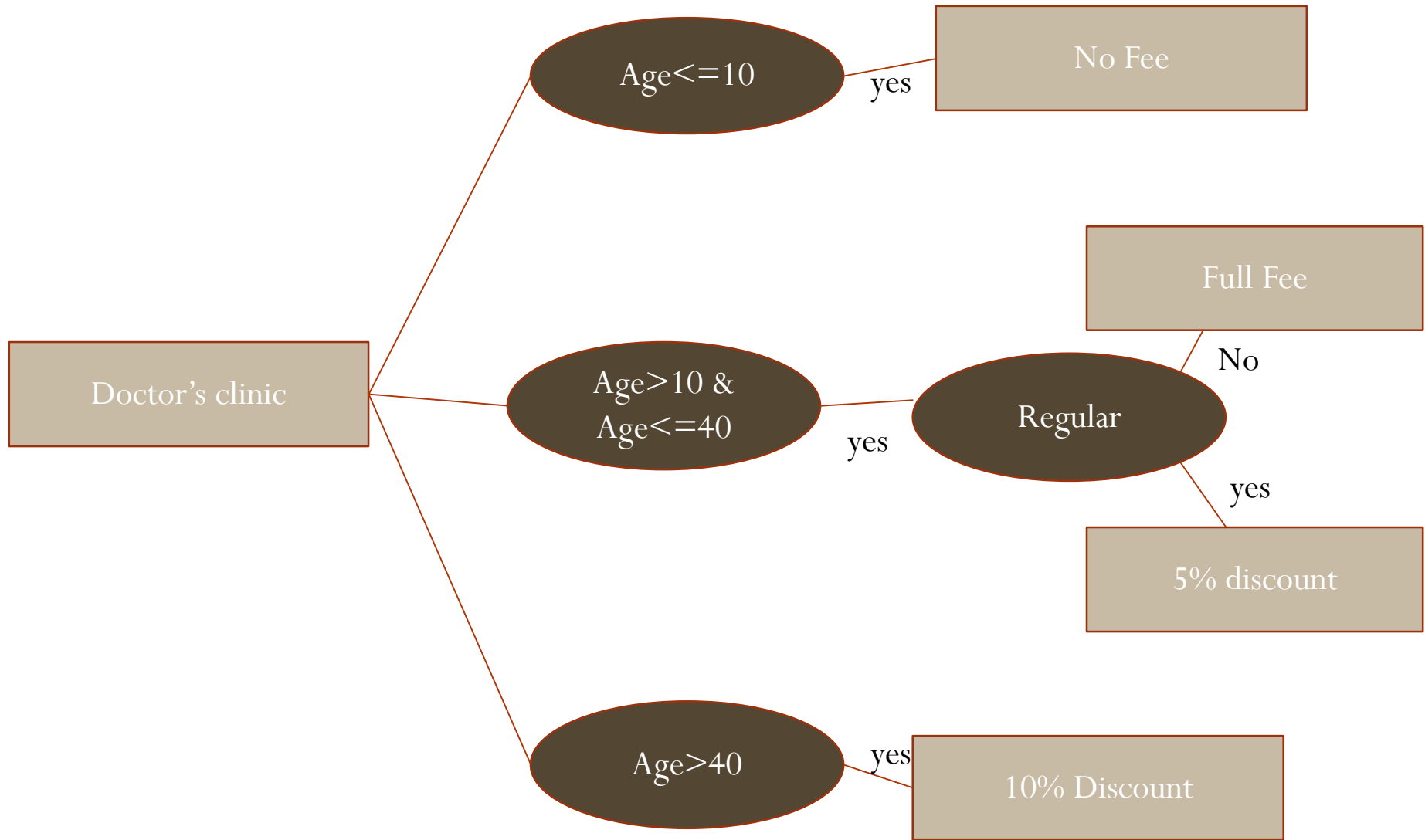
Documentation

- A document is a form of information . A document can be put into an electronic form and stored in a computer as one or more files.
- Often a single document becomes a single file. An entire document or individual parts may be treated as individual data items. As files or data, a document may be part of a database .
- The organized collection of records that describe the structure, purpose, operation, maintenance, and data requirements for a computer program, operating system, or hardware device is known as documentation.
- **Tools used for Documentation procedure and decisions:**
 1. **Decision Tree**
 2. **Decision tables**

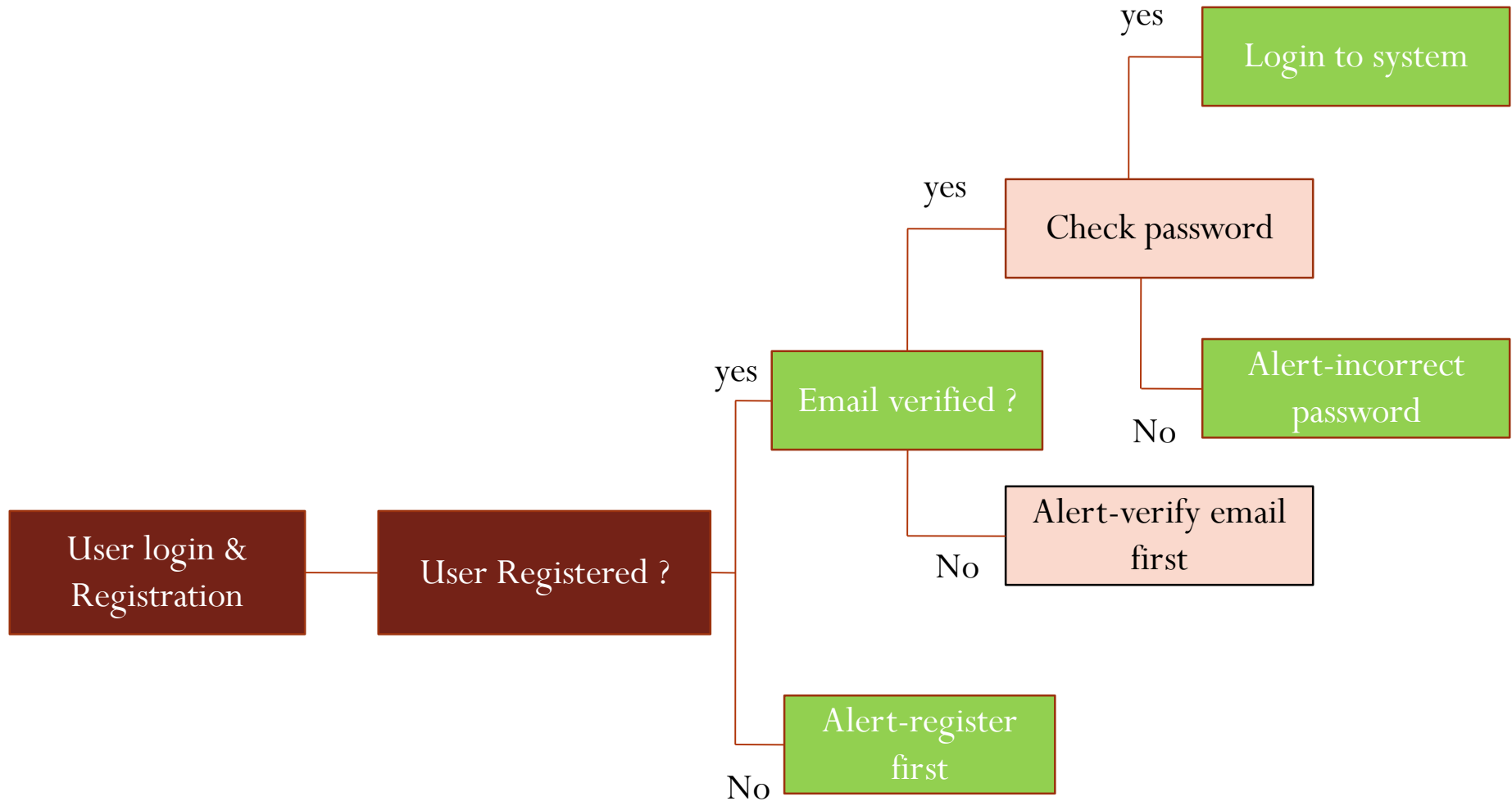
1) Decision Tree

- A schematic tree-shaped diagram used to determine a course of action or show a statistical probability.
- Each branch of the decision tree represents a possible decision or occurrence. The tree structure shows how one choice leads to the next, and the use of branches indicates that each option is mutually exclusive.
- It helps in processing logic involved in decision making and corresponding actions taken.
- It is a diagram that show conditions and their alternative actions within horizontal tree framework.
- It helps to consider the sequence of decision and indentifies the accurate decision that must be made.
- Decision trees are commonly used in operations research, specifically in decision analysis, to help identify a strategy most likely to reach a goal.

Doctor's clinic



User Login-Registration System



2) Decision tables

- Decision tables are used to model complicated programming logic. They can make it easy to see that all possible combinations of conditions have been considered; when conditions are missed, it is easy to see this. The tables are composed of 4 parts: conditions, actions, condition alternatives (each column is a rule), and actions for the rules.
- A decision table is a table composed of rows and columns, separated into four separate quadrants.

conditions	Conditions Alternative
Actions	Action Entries

- The upper left quadrant contains the conditions. The upper right quadrant contains the condition rules for alternatives. The lower left quadrant contains the actions to be taken and the lower right quadrant contains the action rules.

User login & Registration System

condition	C-1	C-2	C-3	C-4
Username	0	0	1	1
Password	0	1	0	1
Action				
Output	error	error	error	Login

Doctor's clinic

condition	C-1	C-2	C-3	C-4
Age≤8	yes	no	no	no
Age>9 & age >40(regular)	no	yes	no	no
(new)	no	no	yes	no
Age>40	no	no	no	yes
Action				
Fees	No fee	5% discount	Full fees	10% discount

DATA FLOW ANALYSIS TOOLS

- Following are the tools for Data flow Analysis
- 1) Data Flow Diagram
- 2) Data Dictionary

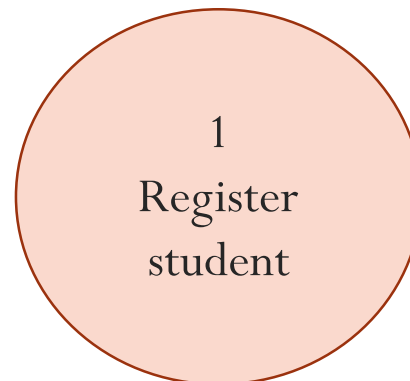
DATA FLOW DIAGRAM(DFD):

- A Data Flow Diagram (DFD) is a diagrammatic representation of the information flows within a system, showing:
 - how information enters and leaves the system,
 - what changes the information,
 - Where information's stored.
- A DFD shows the flow of data from data sources and data stores to processes and from processes to data stores and data sinks.
- DFDs are used for modelling and analyzing the flow of data in data processing systems, and are usually accompanied by a data dictionary, an entity relationship model, and a number of process descriptions.

Components of DFD are:

1) Processes

- Processes are parts of the system that receive inputs and generate outputs.
- Processes are drawn as circles on a data flow diagram, which are labeled by a ``process number" and a ``process name." The process name is a short phrase that describes what the process is supposed to do. The process number is unique - that is, every process shown on a system's data flow diagram(s) has a different one.



2) Data Flows

- Data Flows represent information transferred between processes, stores, and external users or systems.
- Data flows are represented using arrows on data flow diagrams, which are labeled by the name of the data that can be transmitted along them .

ID Number



3) Data Stores

- Data stores are data storage areas that are internal to the system that is being modeled.
- They often correspond to some part of the entity-relationship diagram for the same system. In the data flow diagrams presented in this course, a data store will be drawn as a rectangle with the right hand side missing, labeled by the name of the data storages area it represents.



Student

- Some other references, including Point, use a different notation, in which the left hand side of the rectangle is missing, too, so that the name for the data store appears between two horizontal lines. Thus, the ``Students" data store shown above would be drawn as follows.



Student

4) Terminators(External Entity)

- Terminators represent users of the system - people or other systems that are external to the system being modeled, whom the system being modeled communicates with, by receiving input or returning output.
- A terminators is shown as a rectangle on a data flow diagram that is labeled by the name (or role) of the system user that it represents.



DFD rules and tips

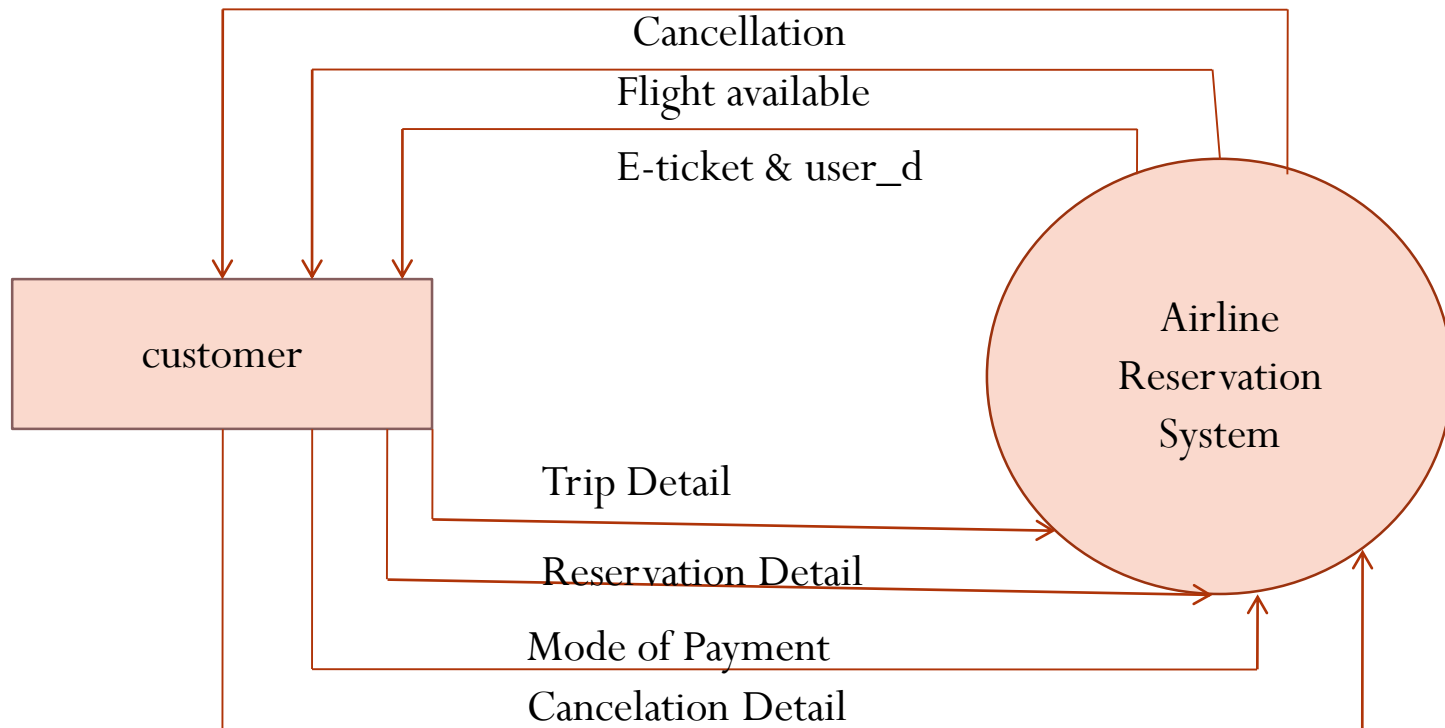
- Any data flow leaving a process must be based on data that are input to the process
- All data flow are named, the name reflects the data flowing between process, data stores , sources , or links.
- Consider only those data which are needed in process.
- Maintain consistencies between processes that are not identified in the higher level diagram are introduced at the lower level.
- However, within the process new flow and data stores are identified.
- Follow meaningful leveling conventions: leveling refers to the handling of local files data stores and data flows that are relevant only to the inside of a process are concealed until that process is exploded into greter detail.

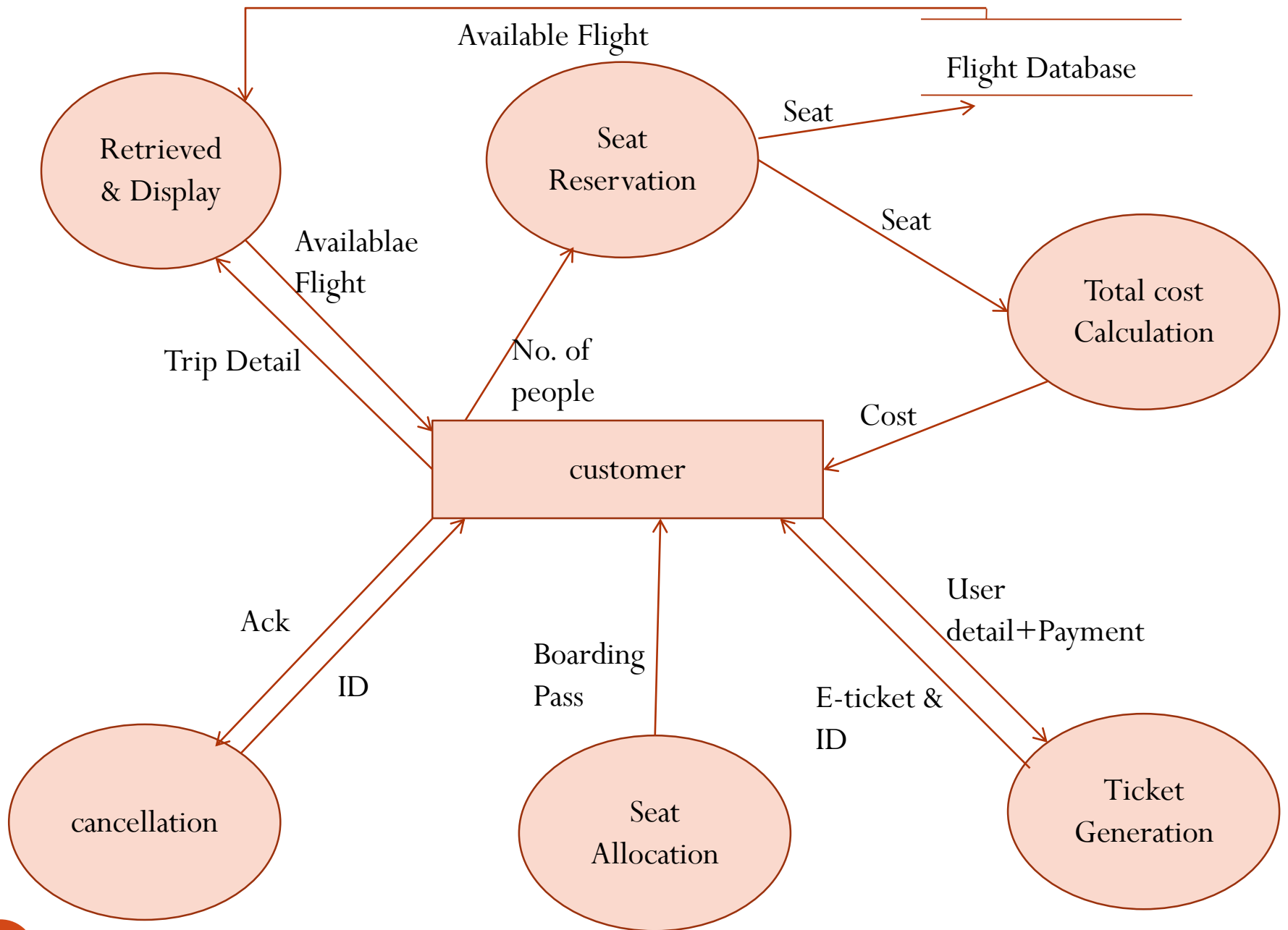
- Assign meaningful labels: the descriptions assigned to data flows and processed should tell the reader what is going on all data flows should be named to reflect their content accurately.
- Data flow naming: the name assigned to data flows should reflect the data of interest to the analyst not the document on which they reside.
- Process naming: all process should be assigned named that tell the reader something specific about the nature of the activities in the process.

Context Diagrams/0 level/bird view:

- A context diagram is a top level (also known as Level 0) data flow diagram. It only contains one process node (process 0) that generalizes the function of the entire system in relationship to external entities.

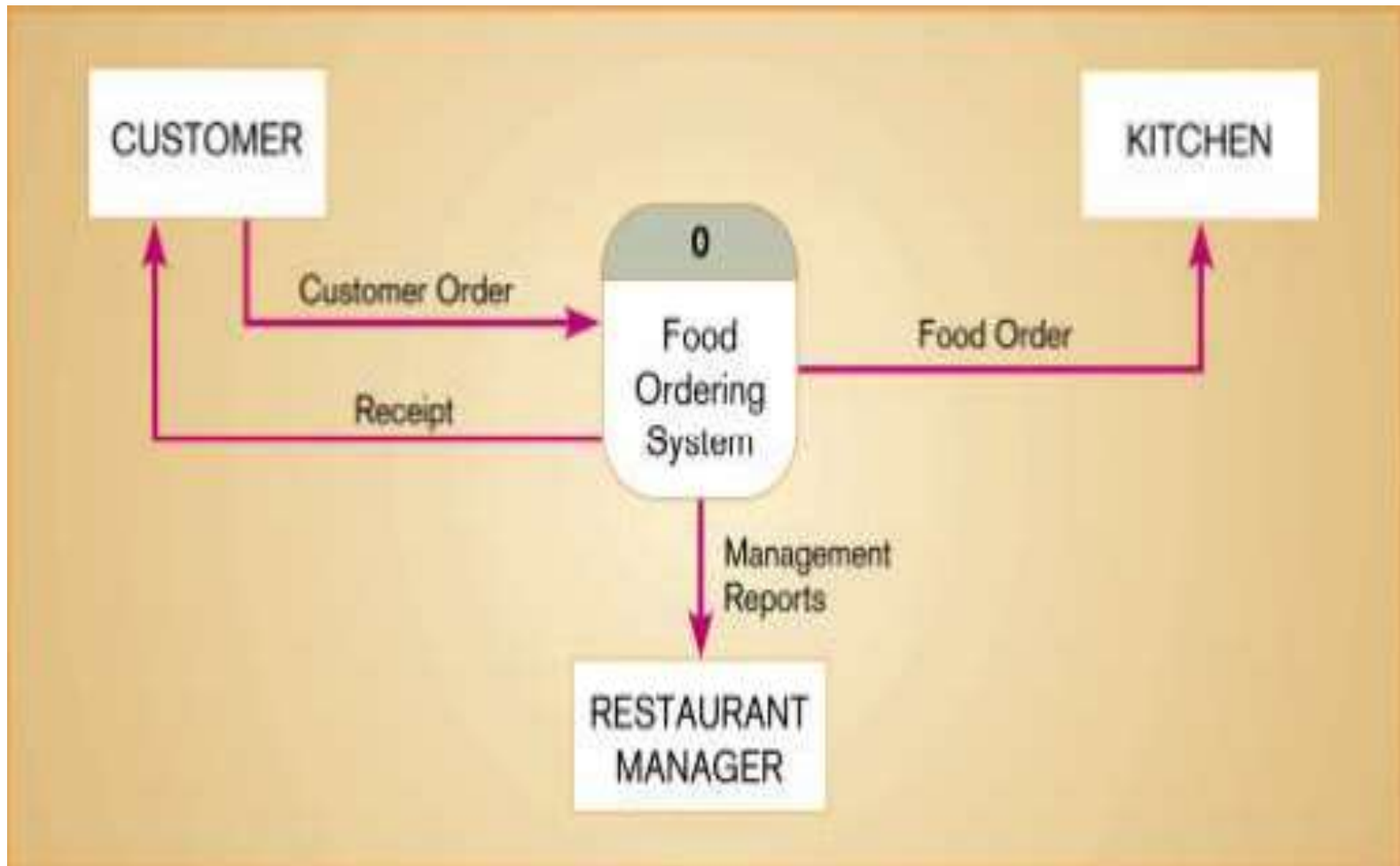
❑ Flight reservation system:



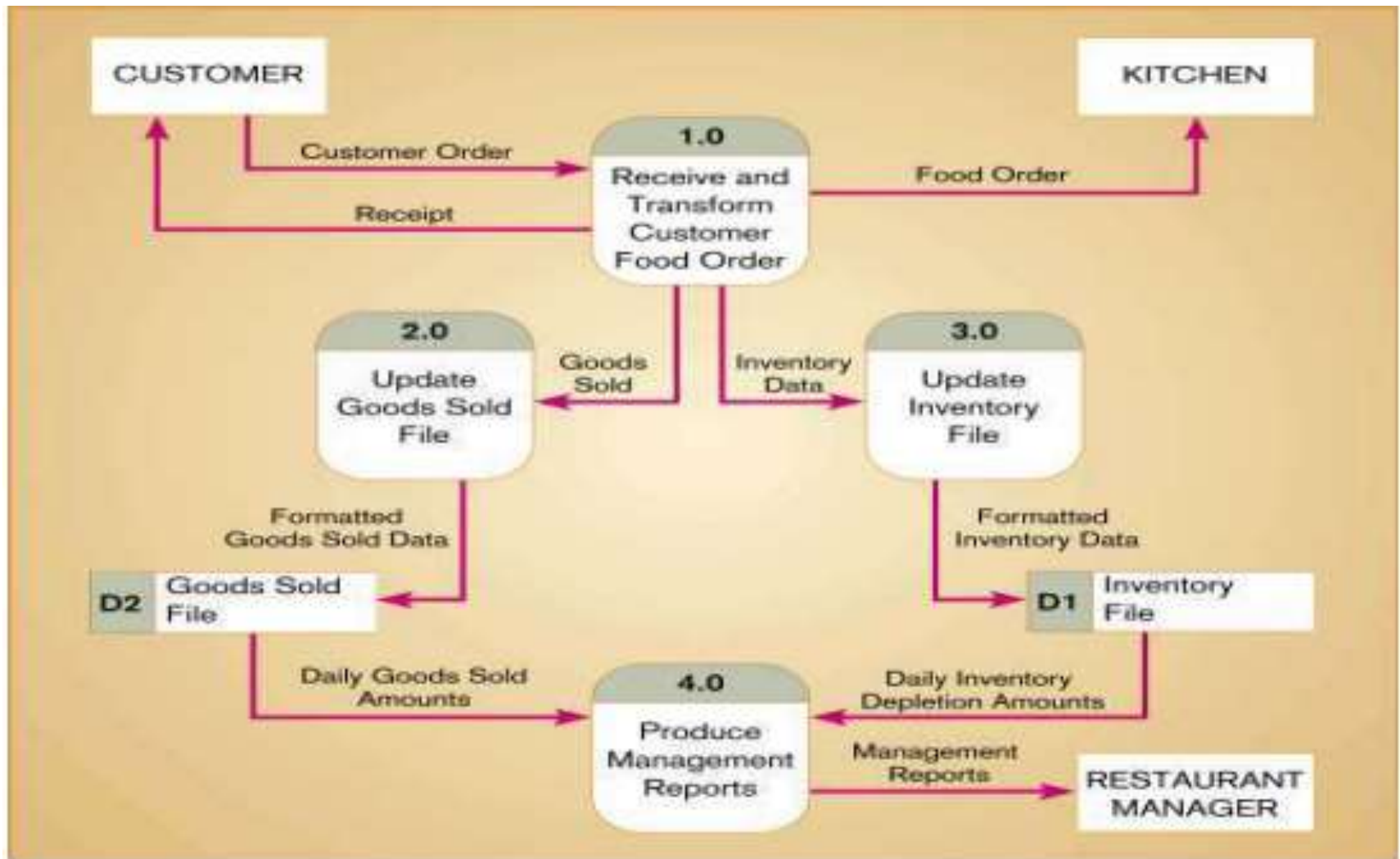


1st level DFD

Food ordering system



0-level DFD



1st level DFD

DATA DICTIONARY

- A data dictionary, or metadata repository, is a "centralized repository of information about data such as meaning, relationships to other data, origin, usage, and format." [1] The term may have one of several closely related meanings pertaining to databases and database management systems (DBMS):
- a document describing a database or collection of databases
- an integral component of a DBMS that is required to determine its structure
- a piece of middleware that extends or supplants the native data dictionary of a DBMS

The term data dictionary and data repository are used to indicate a more general software utility than a catalogue. a data dictionary is a data structure that stores metadata i.e., (structured) data about data.

Active Data Dictionary

- If the structure of the database or its specifications change at any point of time, it should be reflected in the data dictionary. This is the responsibility of the database management system in which the data dictionary resides.
- So, the data dictionary is automatically updated by the database management system when any changes are made in the database. This is known as an active data dictionary as it is self updating.

Passive Data Dictionary

- This is not as useful or easy to handle as an active data dictionary. A passive data dictionary is maintained separately to the database whose contents are stored in the dictionary. That means that if the database is modified the database dictionary is not automatically updated as in the case of Active Data Dictionary.
- So, the passive data dictionary has to be manually updated to match the database. This needs careful handling or else the database and data dictionary are out of sync.

Client table

Client_id	Client_name	password	Contact_no	email
1	ram	1234@	2222333366	ram@gmail.com
2	seeta	4567&	5544339911	seeta@gmail.com

Data Dictionary

Field name	Data Type	Field length	constraints	Description
Client_id	number	10	Primary key	Client id, auto generated
Client_name	varchar	20	Not null	Name of client
password	varchar	30	Not null	Login password
Contact_no	number	10	Not null	Contact of client
email	varchar	40	Not null	Client email

Software Engineering

- Software engineering is concerned with all aspects of software production from the early stages of system specification through to maintaining the system after it has gone into use.
- Software engineering is the branch of computer science that creates practical, cost-effective solutions to computing and information processing problems, preferentially by applying scientific knowledge, developing software systems in the service of mankind.
- Software engineering is about teams. The problems to solve are so complex or large, that a single developer cannot solve them anymore. Software engineering is also about communication.
- Teams do not consist only of developers, but also of testers, architects, system engineers, customer, project managers, etc.

- Software projects can be so large that we have to do careful planning. Implementation is no longer just writing code, but it is also following guidelines, writing documentation and also writing unit tests. But unit tests alone are not enough.
- The different pieces have to fit together. And we have to be able to spot problematic areas using metrics. They tell us if our code follows certain standards. Once we are finished coding, that does not mean that we are finished with the project: for large projects maintaining software can keep many people busy for a long time. Since there are so many factors influencing the success or failure of a project, we also need to learn a little about project management and its pitfalls, but especially what makes projects successful.
- And last but not least, a good software engineer, like any engineer, needs tools, and you need to know about them.

What is Quality Assurance?

- **Software Quality:** Software Quality is defined as the conformance to explicitly state functional and performance requirements, explicitly documented development standards, and inherent characteristics that are expected of all professionally developed software.
- **Quality Assurance:** Quality Assurance (QA) is a common practice to ensure that the end product of any Software Development Lifecycle (SDLC) conforms to the overall and scope-agreed expectations.
- Quality Assurance focuses on how the engineering and management activity will be done?
- As anyone is interested in the quality of the final product, it should be assured that we are building the right product.
- It can be assured only when we do inspection & review of intermediate products, if there are any bugs, then it is debugged. This quality can be enhanced.

What is quality Control (QC)?

- quality Control (QC) is the process through which you achieve, or improve, product quality. The QC process can also include the activities used to eliminate waste processes in the SDLC.
- QC functions involve a range of testing activities used to detect and resolve technical issues. These are followed together with the development processes over the course of the SDLC.
- Quality Assurance (QA) refers to the meta process that ensures continuous and consistent improvement and maintenance of processes that enables a QC job.

Quality Assurance (QA)

Quality Assurance (QA) is the set of actions including facilitation, training, measurement, and analysis needed to provide adequate confidence that processes are established and continuously improved to produce products or services that conform to specifications and are fit for use.

QA is an activity that establishes and calculates the processes that produce the product. If there is no process, there is no role for QA.

QA helps establish process.

QA sets up a measurement program to evaluate processes.

QA identifies weakness in processes and improves them.

Quality Assurance is a managerial tool.

Verification is an example of QA.

Quality Control (QC)

Quality Control (QC) is described as the processes and methods used to compare product quality to requirements and applicable standards, and the actions are taken when a non conformance is detected

QC is an activity that demonstrates whether or not the product produced met standards.

QC relates to a particular product or service.

QC verified whether particular attributes exist, or do not exist, in a explicit product or service

QC identifies defects for the primary goals of correcting errors.

Quality Control is a corrective tool.

Validation is an example of QC.

What are the activities of Quality Analysis?

- Quality Assurance activity works on the quality audit.
- The define the process is one of the activities of quality assurance.
- Tool identification and selection.
- Quality Assurance activity involves training of Quality Standards and processes.
- All the activities are concerned for QA of any product, not for Software.

In the case of software,

QA will act as SQA (Software quality assurance)