

## **CHAPTER 5**

# SYSTEM ANALYSIS

### **Expected Outcomes**

- To discuss requirements determination
- To study methods in gathering requirements
- To discuss the logical modeling of processes by referring to Data Flow Diagram (DFD) as a tool to support analysis
- To know decision tables and structured language usage

# System Analysis

Analysis phase is important because every project requirements and needs must be defined accurately and completely. Based on these requirements, the project will develop a system which meets users expectation.

Systems analysis is a process of understanding in detail what a system should accomplish.

- 
-

# System Analysis

Process of analysis is large and complicated, it can be divided into two parts which are requirements determination and requirements modelling.

This process involves a huge amount of cost and effort.

# System Analysis

A system analyst must conduct eliciting requirement process.

A good analyst should has certain characteristics such as impertinence, impartiality and insight.

Impertinence- question everything and take nothing for granted

Impartiality – unbiased investigation

Insight – assume that anything is possible

# Requirement Characteristics

A well-stated requirements should have six basic characteristics:

Testable and verifiable

Justifiable, accurate and correct

Unambiguous

Consistent

Understandable and modifiable

Traceable

# Requirement Determination Methods

Gathering requirement can use various methods and techniques. Each method has its advantages and disadvantages,

Two general categories can be classified , which are Traditional Methods and Modern Methods.

# Traditional Methods

Traditional methods to gather requirements are as follows:

Interviewing

Surveys and questionnaires

Observation

Analyzing documents

Form analysis

# Modern Methods

Modern methods to gather requirements are as follows:

Joint Application Design – JAD

CASE Tools

Iterative Prototyping



# Interview

Direct Interview consists of meeting with individuals or small groups to ask questions about their roles, responsibilities and needs for the proposed system.

This method can be classified into two types:

Structured interview – prior to interview session , the interviewer prepares specific set of questions.

Unstructured interview – interviewer has general goals or subject

# Interview

There are two basic forms of questions which are:

Open-ended question – let the respondent to answer the question in any way.

Closed-ended question – the respondent has limit set of answers.

# Interview

## Advantages:

- Respondent can be motivated to answer freely
- Build an active contribution to the propose system
- Instant feedback

## Disadvantages:

- Time consuming
- Geographical limitation
- Interviewer must has good communication skill

|

# Questionnaires and Surveys

This method can be targeted to mass respondents and they can complete the questionnaire at their own time.

Advantages:

- Responses can be easily tabulated and analyze
- Less expensive

Disadvantages:

- Response rate often low
- No opportunity to clarify points

## Direct Observation

This method can be used to discover what happen and how it happens. Information obtain can be confirmed through recall that event or process.

Prior to use this method, analyst must know what to observe and set the objectives.

Unfortunately, this method is time consuming.

t

## Analyze Document

Analyst must locate and analyze all written documents in order to gain better understanding of the organization and current system.

By doing research on documents, analyst can confirm the correct requirements.

Disadvantages of this method is reliability of the document if it is not up to date.

# JAD

JAD has been developed in the late 1970s at IBM. It focus to bring together the key stakeholders of the proposed system. JAD session is more highly structured associated with the roles of participants.

It must be conducted in a facility located remotely from participants' working environment.

JAD participants consist of leader, user, manager, analyst, scribe and IS staff

# Iterative Prototyping

Iterative prototyping been used as a model for evaluation by system stakeholders.

A prototype allows the analyst to quickly transform the basic requirements into a limited working model. By allowing user to use the prototype, they can refine and add their needs.



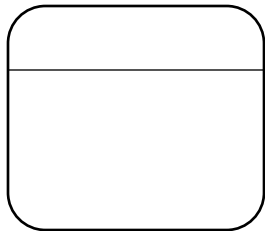
# Modeling The Process

Model the process by using graphical tool that allows the analyst to represent various processes and data flows associated with the system. It is called as the Data Flow Diagram (DFD).

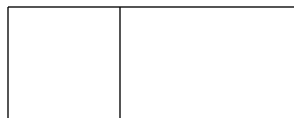
DFD is a graphical tool that depicts the sequence of processes and functions contained within a specified system boundary and the flow of data through that system.

# DFD

Two most are the Gane-Sarson and the DeMarco-Yourdon symbol sets. Below are symbols in Gane-Sarson:

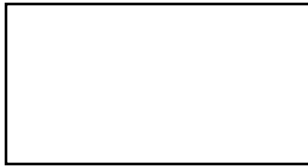


Process

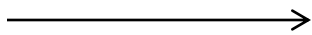


Data store

# DFD



Source or Sink



Data flow

# Components of DFD

- Process
  - An activity or function performed for a specific business reason.
  - Every process has:
    - ✓ A number
    - ✓ A name (verb)
    - ✓ One or more output data flows
    - ✓ One or more input data flows

# Components of DFD

## Data flow

- A single piece of data or a logical collection of data
- Always starts or ends at a process
- Every data flow has
  - ✓ A name (noun)
  - ✓ One or more connections to a process

# Components of DFD

## Data Store

- A collection of data that is stored in some way
- Data flowing out is retrieved from the data store
- Data flowing in updates or is added to the data store
- Every data store has
  - ✓ A number
  - ✓ A name (noun)

# Components of DFD

## External entity

- A person, organization, or system that is external to the system but interacts with it.
- External entity is the origin or destination of data (outside the system).
- Every external entity has
- A name (noun)

# DFD Rules

There are two DFD rules that apply:

- The inputs to a process are different from the outputs of that process.
  - Processes intend to transform inputs data into outputs data
- Objects on a DFD have unique names.
  - Every process has a unique name.



# Context Diagram

Context Diagram is intended to identify the system boundary with regard to its relationship to any source or sink entities that may interact with it.

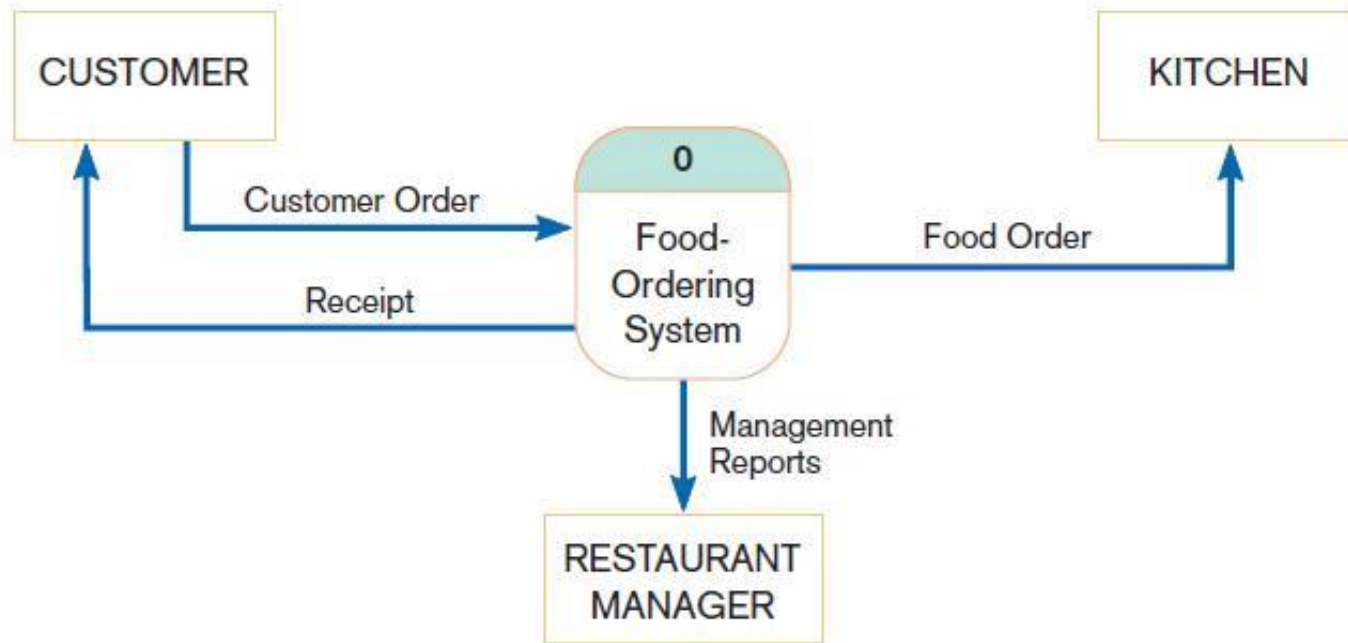
It is the initial DFD in every business process

# Context Diagram

Features of context diagram:

- Top-level view that shows the overall boundaries of the system
- Represent the results of fact-finding
- One process symbol, numbered 0 (zero) is drawn in the center
- Data flows connect the process to the entities
- No data store

# Context Diagram



**FIGURE 5.1**  
Context diagram of food-ordering system

# DFD

Level- 0 diagram is a data flow diagram that signifies a system's major processes, data flows, and data stores at a high level of detail.

- ✓ Processes are numbered as 1.0, 2.0, 3.0 etc.
- ✓ These will be decomposed into more lower-level DFDs.

# Decomposing DFD

Decomposing DFD is repetitive process to break a system into finer detail.

One process in upper level DFD is explained in greater detail.

Example: level-1 DFD results from decomposition of level-0 DFD.

The process of decompose continues until no sub process can logically be broken down any further.

## Balancing DFD

The concept of balancing define that all the input data flows to a process and all the output data flows from a process in the parent diagram should be preserved at the next level of DFD decomposition.

# Structured English

Describing a process using subset of standard English.

It describes process logic

- Use only standard sequence, selection, and iteration structures organized into nested and grouped procedures.
- simple English statements such as add, multiply, and move.
- Use indentation for readability

# Decision Table

Describing a process using decision table

It describes process logic

- Show a logical structure that describes process logic
- Every logical combination is shown initially
- Results then can be combined and simplified
- Programmers can use decision tables in developing code



# Decision Tree

- Graphical representation that shows a decision table's conditions, actions, and rules
- Logic structure is shown horizontally
- Easy to construct and understand
- Decision table is better in complex situations

# Questions

1. Discuss common mistakes made by system analyst during requirements determination?
2. What are the advantages and disadvantages of direct observation?
3. Structured English is a modified form of English language used to specify the logic of information system processes. [ T/F ]

# References

J.A Hoffer, J.F. George, and J.S. Valacich, "Modern Systems Analysis and Design", 7/E, Addison-Wesley, 2014

Kenneth E. Kendall, Julie E. Kendall., "Systems Analysis and Design", Pearson, 2014