Course Code:PG CA S2-05(DSE)

Two Year Post Graduate Programme

System Analysis and Design

Second Semester
Discipline Specific Elective Course

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY

Hiranya Chandra Bhuyan School of Science and Technology

KRISHNA KANTA HANDIQUI STATE OPEN UNIVERSITY

Subject Experts

Prof. Anjana Kakoti Mahanta, Gauhati University
Prof. Hiren Kumar Deva Sarma, Gauhati University
Dr. Lipi B Mahanta, IASST, Guwahati
Dr. Sanjib Kr. Kalita, Gauhati University

Course Co-ordinator :

Dr. Sruti Sruba Bharali, KKHSOU

SLM Preparation Team

UNITS	CONTRIBUTORS
1, 11, 15	**Dr. Utpal Barman**, GIMT, Guwahati
2	**Dr. Tapashi Kashyap Das**, KKHSOU
3,4,5	**Mr. Saurabh Sutradhar**, Royal Global University
6,7, 8	**Dr. Gautam Chakraborty**, NERIM
9, 10, 12	**Mrs. Shrabani Medhi**, GIMT, Guwahati
13, 14	**Mr. Adarsh Pradhan**, GIMT, Guwahati

Editor

Content: Dr. Ridip Dev Choudhury, KKHSOU

Structure, Format & Graphics: Dr. Ridip Dev Choudhury, KKHSOU

This Self Learning Material (SLM) of the Krishna Kanta Handiqui State Open University is made available under a Creative Commons Attribution-Non Commercial-Share Alike 4.0 License (international): http://creativecommons.org/licenses/by-nc-sa/4.0/

Printed and published by Registrar on behalf of the Krishna Kanta Handiqui State Open University

Headquarters: Patgaon, Rani Gate, Guwahati-781017

City Centre: NH 37, Khanapara, Guwahati-781022; Web:www.kkhsou.in

The University acknowledges with thanks the financial support provided by the **Distance Education Bureau, UGC** for the preparation of this study material.

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY
System Analysis and Design

DETAILED SYLLABUS

		Pages
UNIT 1 :	**System Concept** System definition, Characteristics of a system: Organization, Interaction, Interdependence, integration, Elements of a system: inputs and outputs, Processors, control, feedback, environment, boundaries and interface, Types of systems: Physical or abstract system, open or closed system.	**7-14**
UNIT 2 :	**System Development Life Cycle** Introduction, Recognition of need, Feasibility study, Analysis, design, implementation, post implementation and maintenance, consideration for candidate system, prototyping	**15-31**
UNIT 3 :	**Role of System Analyst** Introduction, Interpersonal skill, academic and professional qualification, multifaceted role of the analyst, analyst/user interface, conflict resolution, place of analyst in the MIS organization	**32-44**
UNIT 4 :	**System Planning and Initial Investigation** Introduction, bases for planning in system analysis, dimensions of planning, strategic MIS planning, Managerial and operational MIS planning, initial investigation, needs identification, determining the user's information requirements, background analysis, fact finding, review of written documents, on site observations.	**45-54**
UNIT 5 :	**Information Gathering** Introduction, types of information needed, information about the organization, information about user staff, information about work flow, origin of information, review of literature, procedures and forms, interviews and questionnaires	**55-61**
UNIT 6 :	**Tools of Structured Analysis** Introduction, Data flow diagram (DFD), DFD symbols, constructing a DFD, data dictionary, structured English, decision tree, decision table	**62-86**
UNIT 7 :	**Feasibility Study** Introduction, system performance definition, statements of	**87-101**

constraints, identification of specific system objectives, feasibility consideration: economic feasibility, technical feasibility, behavioral feasibility, steps in feasibility analysis, feasibility report

UNIT 8 : Cost/Benefit Analysis **102-113**

Introduction, data analysis, cost and benefit categories, procedure for cost/benefit determination

UNIT 9 : The Process and Stages of System Design **114-133**

Introduction, The process of design : Logical and physical design, design methodologies, structured design, functional decomposition, Forms driven methodology, HIPO and IPO chart

UNIT 10 : Input/Output and Form Design **134-156**

Introduction, Input design, input data, source data, input media and devices, online data entry, formatted form, screen design, output design, form design, requirements of form design, types of form, box design

UNIT 11 : Database Design **157-175**

Introduction, objective of database, key terms in database, Logical and physical views of data, schemas and subschemas, Types of relationships, Types of data structure : hierarchical, network, relational, entity and attributes, role of dataset administrators, Normalization

UNIT 12 : System Testing **176-196**

Introduction, Need for system testing, nature of test data, test plan, activity network for system testing, types of system test

UNIT 13 : Quality Assurance **197-220**

Introduction, quality factor specification, levels of quality assurance, software requirement specification, software design specification

UNIT 14 : Implementation **221-231**

Introduction, conversion, activity network for conversion, file conversion, creating test files, training aids

UNIT 15 : Project Scheduling **232-242**

Introduction, reason for system failure, project management, Planning tools: Gantt charts, program evaluation and review techniques (PERT), project management software.

COURSE INTRODUCTION

Welcome to the world of System Analysis and Design! This course is all about understanding how systems work, planning them effectively, and making them better. We'll break down complex concepts into simple language, so you can learn step by step.

In this course, you'll explore the following key areas:

Unit 1: System Concept
We'll start by defining what a system is and its characteristics. You'll learn how systems are organized, how they interact, and why they are essential in today's world.

Unit 2: System Development Life Cycle
Discover the journey a system goes through, from recognizing the need for it to maintaining it after it's built. We'll also talk about prototyping, a creative way to design systems.

Unit 3: Role of System Analyst
Explore the role of a system analyst and how they help bridge the gap between technology and people. Learn about the skills and qualifications needed for this important job.

Unit 4: System Planning and Initial Investigation
Understand the planning phase, where we figure out why a system is needed and what it should do. We'll also delve into initial investigations and identifying user requirements.

Unit 5: Information Gathering
Discover how to collect the right information to build an effective system. We'll cover various methods, including interviews and questionnaires.

Unit 6: Tools of Structured Analysis
Learn about essential tools like Data Flow Diagrams (DFD) and Decision Trees that help visualize and design systems.

Unit 7: Feasibility Study
Find out how to determine if a system idea is feasible by looking at its economic, technical, and behavioral aspects.

Unit 8: Cost/Benefit Analysis
Explore how to analyze the costs and benefits of a system to make informed decisions.

Unit 9: The Process and Stages of System Design
Dive into the design process, including logical and physical design, methodologies, and more.

Unit 10: Input/Output and Form Design
Understand how to design inputs, outputs, and forms for your system effectively.

Unit 11: Database Design
Discover the importance of organizing data in a structured way and learn about different types of databases.

Unit 12: System Testing
Explore the critical phase of testing a system to ensure it works as expected.

Unit 13: Quality Assurance
Learn about maintaining the quality of the system through proper documentation.

Unit 14: Implementation
Explore how to turn your system into reality, including data conversion and training.

Unit 15: Project Scheduling
Understand the importance of planning and managing a project using tools like Gantt charts and PERT.

By the end of this course, you'll have a solid understanding of how to analyze, design, and build systems that can make a real impact in various fields. So, let's embark on this learning journey together and become skilled System Analysts and Designers!

UNIT 1: SYSTEM CONCEPT

UNIT STRUCTURE

1.1 Learning Objectives

1.2 Introduction

1.3 System Definition

1.4 Characteristics of System

1.5 Elements of System

1.6 Types of System

1.7 Let Us Sum Up

1.8 Further Reading

1.9 Answer to Check Your Progress

1.10 Model Questions

1.1 LEARNING OBJECTIVES

After going through this unit, you will be able to –

- define a system
- understand the characteristics of a system
- describe the elements of a system
- describe different types of a system

1.2 INTRODUCTION

This is the first unit of the block for the course “System Analysis and Design”. This unit covers topics like system definition, characteristics of system, elements and types of system in detail. The next unit will explore the topics related to system development life cycle.

1.3 SYSTEM DEFINITION

A system is a group of related modules or parts or elements that work together. Interacting, interrelated and interdependent elements are combinedly known as system. In computer science, a system is a set of components such as input, output, process, memory and communicating

devices. A computer system means a digital processing device. It is a collection of technologies to serve the users based on the set of requirements. For example, in a computing system, social networking applications like Instagram, Netflix are considered as a system. In real life, hotels, homes, buildings etc. are considered as a system. All systems have some components that works together to serve the user to fulfill the requirements.The followings are the components of a computer system:

i) Input
ii) Processing
iii) Output
iv) Memory
v) Communication devices

The above-mentioned components of a computer system have their purposes, and they are used to serve the requirements of a user. The input components such as keyboard, mouse, etc. are used to put the input to the computer system. Like, processing components process the input parameters to produce the output and display the same in the output devices, printer, monitor, etc. Communicating devices send the information from one device to another or the other components of the computer system. The term design denotes the process of selecting the parameters to design the computer system, the interaction process, etc.To design a system, the followings parameters are required to satisfy the requirements of a system

i) Architecture
ii) Product design
iii) Interfaces
iv) Data

1.4 CHARACTERISTICS OF SYSTEM

A system is a combination of different components. A complete system must have the following characteristics.

a) **Organization**: It defines the structural representation of the components of a system to achieve a goal based on the user

requirements. It is the arrangement of components of a system for objectives.

b) **Interaction**: It defines the interactions between the components of a system. For example, in the computer system, the input, data, and process complements are interacted among themselves to produce the outputs.

c) **Interdependence**: It defines the dependencies of one component to other components in a system. It also shows the linkage between the components.

d) **Integration**: It explains how different subsystems are tied together to form a large-scale system. Those subsystems or components are integrated to achieve the objectives of the system.

e) **Central Objectives**: Though a system consists of many subsystems, and components, it always produces a unit and central objectives.

1.5 ELEMENTS OF SYSTEM

As mentioned above, a system has different components and elements. The followings are the elements of a system.

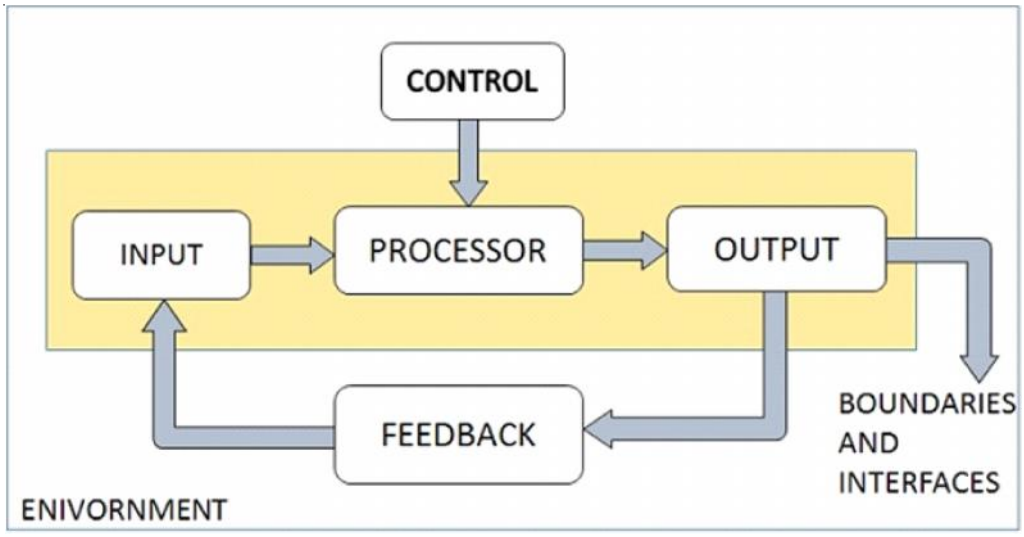


Figure 1.1: Components of System

i) Components: A component is one of the important elements of a system. It can be also considered as a subsystem. Because each component may be also a system. Different components form a system.

ii) Interrelated components: A system may have different components, that may have interrelation among them.

iii) Boundary: The boundary defines the outer line of a system that differentiates a system from the environment. It shows the inner and outerlines of a system.

iv) Purpose: It defines the final aim of a system, that must have some objectives. Every system should have a purpose, aim, and objectives.

v) Environment: Apart from a system, everything external that interacts with the system, is known as the environment.

vi) Interfaces: It is the point area where a system touches the environment or a subsystem meets another subsystem.

vii) Constraints: A limit to what a system can accomplish.

viii) Input: The information or data or fact that enters into the system for processing, is known as input.

ix) Output: The main objective of a system is to get an output that is helpful for its user. Output is the outcome of processing.



CHECK YOUR PROGRESS

Q 1: What do you mean by a system?

Q 2: What are the system components?

Q 3: Define two characteristics of a system.

Q 4: System Study involves identifying current deficiencies and establishing new goals (State whether True or False)

Q 5: System Study involves the study of an existing system (State whether True or False)

1.6 TYPES OF SYSTEM

The followings are the types of a system:

i) **Physical and Abstract Systems**: The physical systems are perceptible units. One can touch and feel them. A physical system

may be static or dynamic. For example, in the computer system, the keyboard, mouse, monitor, etc. are the physical parts of the computer, that are static. But a software or a program is a dynamic system, that produces different outputs based on the user requirements and user's needs. An abstract system is a non-physical unit or conceptual thing, that has to formulate, to form a real system.

ii) **Open and Closed Systems**: A system that interacts with its environment, known as open system. It receives inputs and delivers outputs to the outside of the system. It is a process that exchanges information with its environment. For example, information systems. A system that does not interact with its environment, is known as a closed system. It is isolated from environmental parameters. A completely closed system is rare in reality.

iii) **Adaptive and Non-Adaptive System**: Adaptive systems are adaptive in nature. To improve the performance and survive, the system adopts the changes with the environment. For example, a human being can adopt new things to survive and for a better life. A non-adaptive system does not respond to the environment. For example, machines.

iv) **Permanent or Temporary System**: A permanent system is built for the long term and it persists for long time objectives. For example, business policies. A temporary system is made for a short goal, and then it is demolished. It is only used for short-term objectives or requirements. For example, a DJ system is set up for a program and it is dissembled after the program.

v) **Natural and Manufactured System**: Nature creation and its system is known as the natural system. For example, the Solar system. The man-made system or machine-based system is manufactured. For example, Rockets, buses, etc.

vi) **Deterministic or Probabilistic System**: Deterministic systems determine the objective based on the interaction between the system components in a predictable manner. For example, different steps

are combined to create an algorithm. A probabilistic system shows uncertain comportment of the components of the system. The precise output of the system is not known to anyone. For example,the probability of happening of ludo dice.

vii) **Social System**: It is made up of people for sharing information, thoughts, etc. For example, social clubs. In Human Social Machine System, both machines and humans are involved to perform a particular task. For example, computer programming, machine system. All the tasks are performed by the machine. For example, an autonomous robot.



CHECK YOUR PROGRESS

Q 6: What is a closed system?

Q 7: State two properties of an open system.

Q 8: Difference between adaptive and nonadaptive systems?

Q 9: What is a physical system?

Q 10: The Robotics system is a social system (True or False).

Q 11: Rocket is a natural system (True and False)

Q 12: The machine is an adaptive system (True or False)



1.7 LET US SUM UP

- A system is a group of related modules or parts or elements that work together. Interacting, interrelated, or interdependent elements are together known as systems.
- A computer system means a digital processing device. It is a collection of technologies, to serves the users, based on the set of requirements.
- All the types of the system have some components that work together to serve the user to fulfill the requirements.
- In a computer system, the input, data, and process are interacting among themselves to produce the outputs.

- In a Computer system, the keyboard, mouse, monitor, etc. are the physical parts of the computer center, that are static.
- A system that interacts with its environment, knowns open system.
- A system that does not interact with its environment, is known as a closed system.
- The machine is a non-adaptive system.
- A DJ system is set up for a program and it is dissembled after the program. So, it is a temporary system.
- Rockets, buses, etc. are the manufactured system.
- The probability of happening of ludo dice is the probabilistic system.
- An autonomous robot is a social system.



1.8 FURTHER READING

1. https://www.tutorialspoint.com/system_analysis_and_design/system_analysis_and_design_overview.htm.



1.9 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q No 1: A system is a group of related modules or parts or elements that work together. Interacting, interrelated, or interdependent elements are togetherly known as systems.

Ans. to Q No 2: The followings are the components of a computer system.

a. Input
b. Processing
c. Output
d. Memory
e. Communication devices

Ans. to Q No 3: A complete system must have the following characteristics.

a. Organization.
b. Interaction

Ans. to Q No 4: True

Ans. to Q No 5: True

Ans. to Q No 6: A system that does not interact with its environment, is known as a closed system.

Ans. to Q No 7:

a. An Open system interacts with its environment.

b. The open system shares information with the environment.

Ans. to Q No 8: Adaptive systems are adaptive. To improve the performance and survive, the system adopts the changes with the environment. For example, a human being can adopt new things to survive and for a better life. A non-Adaptive System is a system that does not respond to the environment. For example, machines.

Ans. to Q No 9: The physical systems are perceptible units. One can touch and feel them. A physical system may be static or dynamic. For example, in the Computer system, the keyboard, mouse, monitor, etc. are the physical parts of the computer center, that are static

Ans. to Q No 10: True

Ans. to Q No 11: False

Ans. to Q No 12: False



1.10 MODEL QUESTIONS

Q 1: What is a system?

Q 2: What are the characteristics of a system?

Q 3: Why do we need to design a system?

Q 4: What is a large-scale system?

Q 5: What are the components of asystem?

Q 6: What are the types of systems?

Q 7: What is the difference between open and closed system?

Q 8: What is the difference between adaptive and non-adaptive systems?

Q 9: Explain about autonomous robot system.

*** ***** ***

UNIT 2 : SYSTEM DEVELOPMENT LIFE CYCLE

UNIT STRUCTURE

2.1 Learning Objectives
2.2 Introduction
2.3 The System Development Life Cycle (SDLC)
2.4 Phases of SDLC
    2.4.1 Recognition of Need
    2.4.2 Feasibility Study
    2.4.3 Requirement Analysis
    2.4.4 Design
    2.4.5 Development
    2.4.6 System Testing
    2.4.7 Implementation
    2.4.8 Maintenance
2.5 System Documentation
2.6 Let Us Sum Up
2.7 Further Reading
2.8 Answers to Check Your Progress
2.9 Model Questions

2.1 LEARNING OBJECTIVES

After going through this unit, you will be able to –

- learn various phases of System Development Life Cycle
- learn how a system can be developed following various phases
- identify different task in each phases of system development
- describe system documentation

2.2 INTRODUCTION

Previous unit is about system and its characteristics. Various elements associated with a system are descibed in the unit. We have also been familiarized with the types of system like physical, abstract open, and closed system. are also discussed.

This unit intends to acquaint you with system development and the various phases associated with the life cycle of the system development.



Automation
The act of implementing the control of equipment with advanced technology

2.3 THE SYSTEM DEVELOPMENT LIFE CYCLE (SDLC)

In this age of computer, automation is preferred everywhere rathar than manual system. For this, some standard and structural procedure or methodology is to be introduced in the organization so that the transition from manual to automated system becomes easy. The concept of *System Development Life Cycle*, in short SDLC has come into existence as a result of the transition from the manual system to automation.

Just as all living beings follow a life cycle, systems also follow a life cycle. The System Development Life Cycle is the set of activities that analysts, designers and users carry out to develop and implement an information system. The system analyst gives a system development project meaning and direction. A system development project begins with a decision to analyse user requirements. The project is completed once a usuable software product is produced and implemented and released. The SDLC is a systematic process for developing software that ensures the quality and correctness of the software product. This process tries to produce a high-quality software product so that it meet with custermers expectation.

The process of software development that starts with the requirement analysis, design, development of software and implementation, and continues through the system testing, evaluation and maintenance of the system, is called the ***System Development Life Cycle****.*

2.4 PHASES OF SDLC

The entire System Development Life Cycle can be divided into different phases of development such as:

- Recognition of Need
- Feasibility Study

- Requirement Analysis
- Design
- Development
- System Testing
- Implementation
- Maintenance

In this section, we will be learning about the various stages that make the system's life cycle.

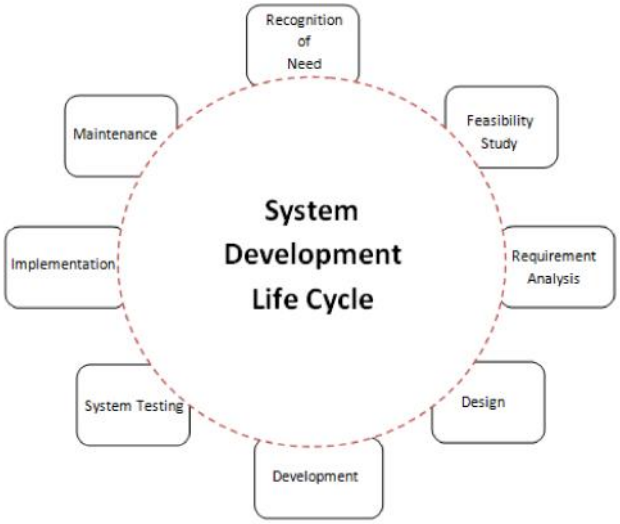


Figure 2.1: Various phases of System Development Life Cycle

2.4.1 Recognition of Need

The recognition of need is basically a survey or assessment of the existing system at the initial stage. This survey is used to determine whether a new alternative system is required or not. Recognition of need is sometime referred to as *preliminary investigation*. The main aim of recognition of need is to identify the problem. The idea of developing a new system arises when some problem appears. One must know what the problem is before going to solve it though it is difficult to identify the real problem. Without preliminary survey and clear understanding of the problem in the system, any further work done will lead to wastage of effort at a later stage. First, the need for the new or the enhanced system is

established. Only after the recognition of need for the proposed system is accomplished then only further analysis is possible. This may lead to a replacement of the existing system by developing a new computer based system. That means, if there is an issue for solving a particular task in the existing system, then it is to determine an alternative solution to that problem by developing a new system.

Detailed investigation is not actually perfomed at this stage. It is the role of a system analyst to obtain more crucial information by interviewing and meeting concerned people who are already working in that situation. This will clarify how the problem was identified, how often it occurs, how it affects the overall functioning etc. A proper understanding of the problem is necessary for the system analyst which enables to separate symptoms from the real causes. It also helps the analyst to consider it as a systems problem rather than other problem and thereby increasing the chances of success in proposing solutions. It will be more clear if we explain this with an example.

Let us assume, in an organization, all leave applications are processed manually. Now, this organization is recruiting many new people time to time. As the number of employees in the organization is increasing, so manual processing of leave application has also become very difficult. And the management is considering the option of automating the leave processing system. If this is the case, then the system analyst would need to investigate the existing system, find the limitations present, and finally evaluate whether automating the system would help the organization.

Once the need for new or improved system is recognised, all possible alternate solutions are chalked out. All these systems are known as ***candidate systems***. From all the candidate systems, the best alternative solution is selected, which is termed as the ***proposed system***. The proposed system is evaluated for its feasibility.

2.4.2 Feasibility Study

Feasibility study is a study, usually carried out by software engineers to determine whether conditions are right to implement a particular project. Feasibility study is an important stage in any system development process. Feasibility for a system means whether it is practical and beneficial to build that system. Feasibility is evaluated from the developer and customer's point of view. The developer sees whether they have the required technology or manpower to build the new system. It is also necessary to check that developing a new system really going to benefit the customer. Does the customer have the required money to build that type of a system? All these issues are covered in the feasibility study of the system.

The feasibility of the system is evaluated on the three main issues: ***technical feasibility***, ***economical feasibility***, and ***operational feasibility***. Another issue in this regard is the ***legal feasibility*** of the project.

Technical Feasibility :

Under technical feasibility, it is necessary to know whether the correct technology exists to support the software to be developed. In technical feasibility, availibility of both hardware and software resources are assessed so that the new product can be developed. Technical skills along with capabilities of technical team for the development of the new product is also analysed at this feasibilty study.

Economic Feasibility :

Under economic feasibility, costs and benefits are determined. Initially a detailed analysis is carried out for determining the overall cost in design and development of the new software product. It is also determined whether the product will be benificial for the organisation or not.

Legal Feasibility :

In this study, the feasibility is assessed in legal point of view. Privacy issues, data protection, authorization, copyright issues, social media laws and ethics etc. come under legal feasibilty study. All these issues and barriers are checked prior to development and implementation of the product.

There is one more issue of feasibility which is known as **operational feasibility** where it is observed that how a system can be maintained after being built.

Feasibility study helps software engineers whether to go ahead with proposed project if the project seems practically feasible or to stop the project if it is not feasible to develop. The outcome of feasibility study is usually a formal document or a report which contains the detail nature and scope of the proposed solution.

2.4.3 Requirement Analysis

The requirement analysis begins once the feasibility study has been completed and the proposed project is found to be finantially and technically feasible. In this phase, the emphasis is on identifying what is needed from the system and not how the system will achieve its goals. The main purpose of requirement analysis phase is to formulate the system's reruirements in a document format. This can be accomplished by analysing the expectations and requirements of the clients who will ultimately use the product.

In this phase, the relationships of the various system components among themselves and with environment are studied and understood. This task is complicated because there are often at least two parties involved in software development - *client* and *developer*.

Usually the developer initially does not understand the client's expectation from the product to be developed, and the client often does not understand the issues involved in software systems. This may lead to a communication gap, which has to be adequately

bridged during the analysis phase. Therefore, a few members of the developer team usually visit the client or customer site. The engineers who gather and analyze client requirements and prepare the requirement specification document are often known as **system analyst**. The system analyst communicate with different categories of persons to collect information about the process and their views for changing the existing process. This requires data collection from a variety of sources for which interviews, questionnaires, forms, study of existing document, records *etc. are necessary. Data flow diagrams(DFD), on-site observations* are also used in this phase.

After being analyzing the user requirements, the system analyst tries to sort out the anomalies, inconsistencies, and incompleteness. He then prepares the *Software Requirements Specification* (SRS) in a document format which is the final output of the requirement analysis phase.

The SRS document is reviewed by a internal project team to ensure that it is easily understanble, complete, consistent, and possess no ambiguity. The document is then given to the client for their approval after review process. After being approved by the client or customer, the proposed project goes for design and development. The SRS document sometime serves as a contract document between customer and the development organization.

Effort is required for writing the SRS document because the this document is needed by different people for various purposes. For example, users, customers, and marketing personnel may ask the SRS document to ensure the system to be developed described in the SRS will meet their needs. Again software developers may also refer to SRS document to make sure that the software that they are developing is exacly same what is required by the client. Sometime the engineers who will maintain the software may also require the SRS document to understand the functionality of the system.

2.4.4 Design

The purpose of the design phase is to plan a solution of the problem specified in the SRS document. It can be accomplished by transforming the customer requirements mentioned in the SRS document into a form that is implementable using a programming language. During the design phase different modules required to implement the design are identified, relationship among the identified modules are also determined. Data structures and algorithms for individual modules are also designed in this phase.



Module
Module is a small part of the software that is responsible for performing any individual task. Sometimes, the term sub-program is also used to refer to the term module.

The design of a system is perhaps the most critical factor affecting the quality of the software, and has a major impact on the later phases, particularly testing and maintenance. The design activities can be grouped into two categories: ***system design*** and ***detailed design***.

In ***system design***, the focus is on identifying the modules, whereas during detailed design the focus is on designing the logic for each of the modules.

The first step in system design is to determine how the output is to be produced and what will be the format of this output. The second step is to design the input data as well as the database to meet the requirements of the proposed output. Thus, system design includes the design of user interface, system interface, network, databases, as well as the major modules in the system and their specifications. The output of this phase is the *design document*. This document is similar to a blue print or a plan for the solution, and it is used later during the implementation, testing and maintenance phase.

During ***detailed design*** further details of the data structures and algorithmic design of each of the modules is specified. The internal logic of each of the modules are decided. In this phase, the SRS document is transformed into logical structure which contains the specifications that can be implemented in a progrmming

language.Thus, the operational phases are handled through program construction and testing, including a list of the programs needed to meet the system's objectives. The final report prior to the implementation phase is also known as *design document.* It includes flowcharts, layouts of record and reports separately, with a workable plan for implementing the candidate system. Information regarding manpower, hardware facililities, and their estimated cost are also included in this report.

A good software design should possess some characteristics like understandibility, correctness, efficiency, and maintainability. A design which is easily understandable is also easy to develop, implement, maintain, and change if required. Software design becomes clear when the design is modular, names of various design components are consistent and meaningful, modules are in a hierarchy. ***Modularity*** and ***decomposition*** should be properly maintained in a good software design. Decomposition of a problem into modules facilitates the design. The complexity of the design reduces if different modules perform different task independent of each other.

In the figure 2.2, a large program P is decomposed into set of three different modules M_1, M_2 and M_3. Specific tasks are performed by each of these three modules. A proper decomposition of the problem into module is very necessary for a good software design. There are two important terms associated with module decomposion : cohesion and coupling.

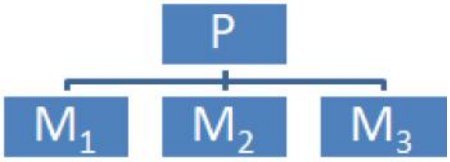


Figure: 2.2: A modular design

Cohesion is a measure of the functional strength of a module, on the other hand **coupling** is a measure of the degree of interdependence or interaction between the two modules. A module

is said to functionally independent of other modules if it possess *high cohesion* and *low coupling.* Funcionally independent means that the module has minimum or no interaction with other modules. A design is good when the modules are functionally independent of each other. It also helps in isolating errors, increase the scope for reuse, and makes easy to understand.



CHECK YOUR PROGRESS

Q 1: Choose the correct answer :

(i) The first step of the SDLC is

(a) System design (b) Preliminary investigation

(c) Development of software (d) None of these

(ii) The detailed study of the present system is referred to as

(a) System planning (b) Feasibility study

(c) System analysis (d) None of these

(iii) The person responsible for the analysis of the system is often called

(a) client (b) programmer

(c) user (d) analyst

(iv) Interviews, questionnaires, forms are required in .....of SDLC.

(a) System Design phase (b) Analysis phase

(c) Preliminary investigation phase (d) None of these

Q 2: Fill in the blanks:

(i) High cohesion and low coupling indicated modules are funcionally.....

(ii) Two important characteristics of a proper module decomposion are _____ and _____.

(iii) In _____ phase, SRS document is transformed into logical structure which contains the specifications that can be implemented in a progrmming language.

2.4.5 Development

Once the system design is completed, the system is actually programmed. The goal of this phase is to translate the design of the system into code in a selected programming language. A well written code reduces the testing and maintenance effort. Since the testing and maintenance cost of software are much higher than the coding cost, the goal of coding should be to reduce the testing and maintenance effort. Hence, during coding the focus should be on developing programs that are easy to write. The analyst should decide whether to buy a commercial software or to develop new customized programs with the help of programmers. This decision depends on the cost of the software and the cost of programming such software. Documentation of the programs while programming the software is also necessary to provide explaination of the procedure so that other programmers can work on the programs if required .

In the system development life cycle, after the completion of the development phase the product becomes ready to be delivered to the client.

2.4.6 System Testing

With the help of system testing, quality of the product is tested by applying various quality control measure after software development. It is the process to ensure that the developed system perform the intended tasks which are already mentioned in the SRS document. Its basic function is to detect errors in the software if any.

System testing is usually carried out in a planned manner. During system testing, the system is used as an experiment to ensure that the software does not fail, i.e., it will run according to its specification. The system is tested with some test data and the results are examined for their validity. For testing to be successful, proper selection of test cases is essential. Some of the users may

be permitted to operate on the system so that the analyst can ascertain that the system can work in the specified environment.

There are three different categories in testing activities, which are ***α -*** *testing,* ***β-****testing* and *acceptance testing*.

Generally, the ***α-testing*** (alpha-testing) is performed by the development team or testers who are usually internal employees of the organization. It is performed before releasing the product to the real user of the software.

β-testing (beta-testing) is performed to access the product by some clients in their environment who are not part of the organization. We can say that beta testing is performed at end-user level. The main focus of beta testing is to access the quality of the product. This test helps in collecting feedback from the users on the product. It also ensures that the product is ready for real time users after fixing the defects if any. This also helps in enhancing the product.

After the product delivery, ***acceptane testing*** is performed to determine whether to accept or reject the delivered product. It is a method of software testing where a system is tested for acceptability.

Testing phase continues until the software becomes stable, error-free and working according to the requirements.

2.4.7 Implementation

In the implementation phase, the organization configures and enables the system security features, tests the functionality of these features, installs or implements the system, and obtains a formal authorization to operate the system. This phase consists of some activities like, installation of required hardware, programs, collection of data, organization of users to interact with and run the system etc.

Necessary training for the users to use the system is also provided in implementation phase. When the new system is linked

to terminals or remote machines/sites, then the test of the network along with the system is also included under implementation.

2.4.8 Maintenance

Once the system is implemented, it should be evaluated. In this phase, systems and products are in place and operating. The organization should continuously monitor performance of the system to ensure that it is consistent with pre-established user and security requirements and that the needed system modifications are incorporated.Thus, *evaluation* is the process of veryfying the capability of a system, after it is put in operation, to see whether it is meeting the objectives or not.

Maintenance includes all the activities after the installation of the software that is performed to keep the system operational. As we have mentioned earlier, software often has design faults. For large systems, removing all the faults before delivery of the software is extremely difficult and faults will be discovered long after the system is installed. As these faults are detected, they have to be removed. Maintenance includes fixing of errors as well as removing of errors.

Maintenance also needed due to a change in the environment or the requirements of the system. Activities like enhancement, adaptation are also included in maintainance phase. Adding new functions or additional capabilities come under *enhancement* and customizing the software to run in the new environment is termed as *adaptation*.

2.5 SYSTEM DOCUMENTATION

System documentation is the collection of documents that describes the requirements, capabilities, limitations, design, operation, and maintenance of a system. It is basically a written document that is often accompanied with a software product. Documentation are very important

because they represent the formal information flow of the present system. It also provided the performance criteria of a system and an explaination of how the system can be used. The documentation needs to be kept up-to-date throughout the project. For instance, a user manual can be a good example of documentation. Usually, a user manual is used to refer for installing, using, and providing maintenance to the software product or application.

The objective of system documentation is to keep the system running in optimum conditions. Some other uses of documentations are:

- To ensure that the system can be maintained after completion even though programmers involved in the original design or programming of the system may not be available.
- To help new programmer for making necessary corrections, adaptation of new features etc.
- To ensure what data should be inputed and how and what information will be available from the system.

There are different types of documentation. Some of them with their usefulness are mentioned here:

- ***Design documentation*** describes the overall system design and includes system flowcharts, all input/output formats, file description etc.
- ***Program documentation*** consists of programming specification like programming logic, control flow, graphics used etc.
- ***Training documentation*** includes manual for user training.
- ***Operation documentation*** describes how to handle the software.
- ***User reference documentation*** provide quick and clear answer just like dictionary.



CHECK YOUR PROGRESS

Q 3: Fill in the blanks:

(a) Customizing the software to run in the new environment is termed as .....

(b) ..... is the process of verifying the capability of a system after it is put in operation.

(c) ..... needs to be kept up-to-date throughout the project

Q 4: State True or False:

(a) Maintenance is the process of incorporating changes in the existing system.

(b) Selection of hardware is an implementation activity.

(c) User interaction is not necessary in analysis phase of SDLC.

(d) Design documentation includes programming secifications.



2.6 LET US SUM UP

The unit discusses the complete development process of a software product with the various stages of System Development Life Cycle(SDLC). The various phases are: *Recognition of need, Feasibility Study, Requirement Analysis, System Design, Development, Testing, Implementation, and Maintenance.* The activities discussed in different phases of SDLC are summarized in the following points:

- Recognition of need is basically a preliminary assessment of the existing system to determine whether an alternative system can solve the problem.
- A feasibility study, as the name implies is a study to determine the necessity for developing a new system or product. It is a process of ensuring a product to be developed is technically. economically as well as legally justifiable.
- In requirement analysis phase, users expectation from the new software product being built are identified. The main purpose of requirement analysis phase is to formulate the system's reruirements in a document format.
- Design phase involves the process of tranforming user requirements into some suitable form which is useful for the programmer or developer to develop the new system.

- During design phase, different modules, interactions among them, algorithms, data structures are identified.
- Based on the design document, programmer build the software product in the development phase. The goal of this phase is to translate the design of the system into code in some programming language.
- Once the developers build the software product, then the testing team tests the functionality of the entire newly developed system. It ensures that the newly developed product works according to the customer requirements.
- After testing process is over the product is released to the customer for implementation. Users are provided training so that they can operate and handle the software product in the implementation phase. Documentation part is also provided for better understanding of the product.
- In the maintenance phase, the developed product is taken care of. As per change of environment and technology, the software product is updated and maintained time to time.

At the end of this system documentation is described. With the help of documentation, maintenance of software becomes easier. All the aspects of software can be understood if proper software documentation is done.



2.7 FURTHER READING

1. Awad Elias M. (1996). System Analysis and Design.
2. Mall, R. (2018). Fundamentals of software engineering. PHI Learning Pvt. Ltd..
3. Kendall, K. E., & Kendall, J. E. (2011). Systems analysis and design (Vol. 2013). Upper Saddle River, NJ: Pearson Prentice Hall.



2.8 ANSWERS TO CHECK YOUR PROGRESS

Ans to Q No 1:

(i) (b) Preliminary investigation (ii) (c) System analysis

(iii) (d) analyst (iv) (b) Analysis phase

Ans to Q No 2: (i) independent (ii) high cohesion, low coupling

(iii) detailed design

Ans to Q No 3: (a) adaptation (b) evaluation (c) documentation

Ans to Q No 4: (a) True (b)True (c) False (d) False



2.9 MODEL QUESTIONS

Q 1: What is SDLC? What are phases associated with SDLC?

Q 2: Define the terms cohesion and coupling in the context of software design.

Q 3: When is a system design said to be good design?

Q 4: What do you mean by feasibility study? In preliminary investigation three types of feasibilities are usually studied. Name them.

Q 5: Why is maintenance of a system necessary?

Q 6: Discuss briefly the various phases of SDLC.

Q 7: What is documentation? Explain the uses of documentation.

Q 8: Write short notes on any one of the following:

(a) System Design

(b) Recognition of need

(c) Analysis

(d) System Testing

*** ***** ***

UNIT 3: ROLE OF SYSTEM ANALYST

UNIT STRUCTURE

3.1 Learning Objectives

3.2 Introduction

3.3 Role of System Analyst

3.4 Interpersonal Skill

3.5 Academic and Professional Qualification

3.6 Multifaceted Role of System Analyst

3.7 User Interface Analyst

3.8 Conflict Resolution

3.9 Place of Analyst in The MIS Organization

3.10 Let Us Sum Up

3.11 Further Reading

3.12 Answers to Check Your Progress

3.13 Model Questions

3.1 LEARNING OBJECTIVES

After going through this unit, you will be able to –

- define the role and responsibility of a system analyst
- understand interpersonal skill and able to apply it
- learn about professional and academic qualification needed for system analyst
- define conflict and apply conflict resolution strategy
- compare and contrast system analyst and MIS analyst

3.2 INTRODUCTION

In the previous unit we learned about the system development life cycle. In this unit, we will learn about the role of system analyst. The unit covers topics like interpersonal skill, academic and professional qualification, user interface analyst and conflict resolution in detail. The next unit explores the topics related to system planning and initial investigation.

Before diving into the role of System Analyst let us recall what a system is.A system is an association of interconnected components for attaining a specific goal. So there should be someone who is responsible for analysing, designing and implementing a full system with its interconnected components to fulfil organizational goals and this person is called a system analyst.A system analyst is a person who conducts a methodical study and evaluation of an activity such as business to identify its desired objectives in order to determine procedures by which these objectives can be gained.In simple words,a system analyst manages the systems of an organization to achieve the desired goal.

3.3 ROLE OF SYSTEM ANALYST

From the discussion above, we can clearly state that the primary role of a system analyst is to analyse the requirements of an organization and the problems associated with those requirements in order to determine what procedure and information technology can be combined together to bring improvements in the organization. A system analyst helps system users and other stakeholders to define their requirements for new or enhanced information services.The analyst must be able to work with people of all disciplines and be experienced in working with information technology. The analyst plays many roles,but three primary roles of the systems analyst can be summarized as consultant, supporting expert, and agent of change.

(a) System Analyst as Consultant

The system analyst frequently acts as a system consultant within an organization and consults its work force and its businesses to curate appropriate information systems for users working in the organization through systematic design approach.

(b) System Analyst as Supporting Expert

In this role, the analyst manages computer hardware and software and suggest their efficient uses in the business of the organization.In this role, system analyst is merely a supporting entity in the entire project, he/she is not a part of project management for the system.

(c) Systems Analyst as Agent of Change

The main role of a system analyst in an organization is that he is the agent of change. An agent of change can be defined as a person who serves as a catalyst for change, develops a plan for change and works with others in facilitating those changes. As an analyst, the main task is to develop a new system or enhance an existing system; but in both cases, a system analyst brings changes to the organization.

An organization requires system analysts as business managers have only knowledge of business problems but they do possess the detailed technical knowledge to determine information-based solutions that are possible for their business problems. A system analyst bridges the gap between the business problem and business system through his/her different roles.

Let's consider one example of the types of organizational problems you could face as a systems analyst. Suppose you work as a system analyst in the information department of a university. The current ERP (Enterprise Resource Planning) System does have the capability to mail student marks sheets. Now it is your duty to devise a plan to enable this capability to the existing system.

3.4 INTERPERSONAL SKILL

From the foregoing descriptions of the roles of a system analyst, it is clear that a system analyst must interact with all stakeholders of a system.So, one of the important skills for a system analyst is Interpersonal Skill. It helps working with end users as well as with other analysts and programmers. As a systems analyst, one plays a major role as a liaison among users, programmers and other systems professionals.

The interpersonal skills required by a system analyst are:

- **Communication**: The analyst needs to be a very good communicator to understand and communicate to the user group as well as to the technical specialists. Sometimes the users may not be able to communicate their needs fully to the analyst, but the analyst must be able to understand their needs from incomplete communication of the users.

- **Foresightedness and vision**: The analyst must have foresight and vision, so that they can factor in the future requirement of the users even if they have not factored that in the design. The analyst must also have vision with regard to the technological changes. He/she must be able to predict where the business needs and technological capabilities/constraints will be in the future. They should also clearly communicate that the design holds good not only for the short term but also the long term.
- **Adaptability and flexibility skills**: The analyst may be new to the environment of the particular business but he/she has to be quick on the uptake and adapt fast to the culture and environment of the organization. Some flexibility in the understanding of problems is also required along with the flexibility to come up with alternative solutions.
- **Selling**: The analyst needs to have flair to sell their ideas and solutions to the users. Sometimes this may be difficult as the users and clients might not know what solution will serve them best. The analyst needs to employ his selling skills to convince the users on the suitability of a solution.
- **Patience and rationality**: The analyst need to be patient and rational so that he/she does not rush to a solution. If they make haste then they might miss critical information about the problem/opportunity and end up promoting a wrong solution for the users. Rationality is also a virtue for the system analyst, as this will help them in analysing the problem/opportunity with a clear mind without prejudice.
- **Sound temperament**: The analyst needs to remain calm in the face of adverse situations. Most of the time the critical data that the analyst seeks is hard to come by and may be late in coming. The analyst will have to put up with all this and be calm in such situations. Thus, the temperament that he/she exhibits will help him/her in devising an appropriate solution for the client.

- **Management skills**: These skills are an absolute necessity for any analyst. The system analyst has to deliver in spite of several constraints hence they must have good management skills to manage time and resources at their disposal.



CHECK YOUR PROGRESS

Q 1: Who is a system analyst?

Q 2: What do you mean by the term "Agent of Change"?

Q 3: Which one of the following is not an interpersonal skill:

a. Communication b. Adaptability

c. Sportsmanship d. Management Skill

3.5 ACADEMIC AND PROFESSIONAL QUALIFICATION

An organization requires system analyst as business managers who have only knowledge of business problems but they do possess the detailed technical knowledge to determine information-based solutions that are possible for their business problems. A system analyst bridges the gap between the business problem and business system through their skills.Therefore, people who understand the full software development life cycle and are able to integrate new and legacy systems and have knowledge of software documentation are preferred for the role of system analyst.

The most direct route to gain this knowledge is to earn a bachelor's degree in computer or information science; however, some system analyst holds bachelor's degrees in another discipline and add computer experience through specialized classes or hands-on experience. For role of system analyst, one must have keen analytical skill and sound technical skill along with academic qualification.

Analytical skills help a system analyst to understand the cohesion and coupling between organization and its functions so that he can identify opportunities and problems and come up with suitable solutions. This type

of analytical skills is called systems thinking because it enables a person to view organizations and information systems as one system.

Technical skills allow a system analyst to understand potential and the limitations of information technology so that he can efficiently implement an information system that will help the users to solve problems and that will guide the system's design and development. Knowledge on different programming languages, various operating systems and computer hardware platforms also help in making informed decision about systems.

3.6 MULTIFACETED ROLE OF SYSTEM ANALYST

A system analyst is a person with multifaceted role who has to deal with each and every function of a system. He is involved in the process right from the development of the software to deployment of it. No other professionals in an organization are so much involved in all processes of a system. A system analyst may play the role of programmer, hardware engineer, project manager as per organizational or business need in design and implementation phase of a project. That's why the role of a system analyst is called a multifaceted role. Let's us discuss these roles briefly.

(a) **Persuader**: A system analyst is an agent of change. He/She persuades or imposes different changes to a system during its development phases or may suggest changes to legacy systems for its enhancement. They must be able to persuade others to bring those changes.

(b) **Investigator**: A system analyst must investigate to find the real problem from existing systems and deduce effective measures to solve them. He/She also monitors programs in relation to time, cost and quality.

(c) **Architect**: Analyst should be able to deduce system design from user's requirements. He/She is the person who helps customers to suggest information systems according to their requirements.

(d) **Manager**: Analyst plays the managerial role as he interacts with different stakeholders of the systems and draws conclusions from these interactions. He/She manages time and resources according

to these conclusions. He/She also helps in user training and motivation to use a new system

3.7 USER INTERFACE ANALYST

A user interface is the part of the system with which the users interact. It includes the graphical displays that comprise of navigation tools to navigate the whole system.It may also include methods for capturing data like user input forms or methods for viewing required data also. A user interface must be simple,easy to understand and navigate and complete.

The role of the User Interface Analyst is to design and plan a new user interface from scratch or to enhance an earlier interface. This includes capturing user requirements through collaboration with business analysts, leading and participating in testing sessions, and coordinating final implementation of the GUI based on feedback.



CHECK YOUR PROGRESS

Q 4: List out the role of a systemanalyst?

Q 5: What are the academic qualifications needed to become a system analyst?

3.8 CONFLICT RESOLUTION

A system analyst must interact with different people during the SDLC of the system. He often may find two or more stakeholders which may have different opinions or interests on the same system and when two people have alternative views on the same topic conflict or clashes occurs. Some of the common ground for such clashes are scarce resources, limited technical knowledge, different priorities and personal issues etc. But whenever a conflict occurs, it lowers the efficiency and productivity of organization, which results in development of low-qualitysystems. That's why we have to resolve conflict during system development.

There are lots of conflict resolution strategies. One of them is known as Thomas-Kilmann's five approaches to resolving conflict which is often

used in conflict resolutions in project management. Let's discuss this strategy in detail. This strategy is mainly comprised of 5 techniques:

1. Withdraw of Avoid

In this conflict resolution technique, we will avoid the conflict or retreat from it and we assume that in due course of time the conflict will resolve by itself. This technique is used if conflicting parties are not influential stakeholders in the project or there is not enough information to apply other techniques. This technique is good for low-level disagreements as it gives ample opportunities for preparation in case of re-emerging conflicts. But withdrawing or avoiding is not really a resolution conflict which may become stronger over time. It may weaken the position of system analyst giving false idea of unfair biasness and lack of conflict resolution skill.'

2. Smooth or Accommodate

In this conflict resolution strategy, we will try to smooth out the rough edges of a tough decision and try to circumvent it. This technique shows a considerable view and tries to accommodate all parties on board. This strategy is applied when there is time constraint or there is immediate need of a temporary solution.

This conflict resolution strategy is effortless, easy to implement and ignores unnecessary clashes. Situations can be handled simply showing harmony and goodwill to the situation which provide enough time to more permanent solutions. But if the approach is not balanced and becomes biased to one party due to more consideration then the other party involved may not be happy and conflict may rise again.

3. Compromise or Reconcile

In this conflict management strategy, we evaluate all suggestions from all parties involved and partially satisfied them. This technique is useful when the stakeholders involved have equal status, need a win situation for all parties, Collaborative and forcing techniques have failed, a temporary solution is must needed. This technique brings quick result and keeps all parties satisfied until a permanent solution has surfaced. In this strategy, all parties are partially satisfied and hence partly unsatisfied also. In long run, conflict may resurface again.

4. Force or Direct

In this strategy, only suggestions from one party have been considered and enforced in project management accordingly. This technique is applied when viewpoints of one stakeholder is right, the gravity of the conflict is not so huge or there is no time to investigate and we need a quick fix. Applying this technique may lead to loss of other valuable viewpoints and this technique cannot be applied when all stakeholders of the conflict are powerful.

5. Collaborate or Problem Solve

In this method, people involved in the conflict come forward to discuss the problem at hand with a very open mind. They focus on resolving the conflict and finding the best alternative/solution for the team. It is used to incorporate multiple views of influential stakeholders and distribute responsibility equally.

This is a real problem-solving technique that provides a solution to the conflict. This leads to a win-win kind of an outcome. Here everyone collaborates. When situations need a quick solution, this method cannot be used. It can only be used for conflicts that may severely affect the main system.

3.9 PLACE OF ANALYST IN MIS ORGANIZATION

MIS (Management information systems) analysts are analytic professionals that help organizations enhance their current information systems or to create a new one from scratch. They can help an organization to identify and create solutions that will move its processes from manual to digital capabilities and paperless through their technical and analytical skills. for this,the MIS analyst examine how an organization's staff members utilize information; the tools they have available and ways they intend to improve their methods and then carry out further investigations through industry documentation and consulting with different stakeholders to identify the best solutions to existing systems. Such improvements may be in terms of hardware technology and software applications.

Management information systems analysts are also required for training the staff for a new system, to draw authentication policies for data and systems They are also responsible for creating strategies for the deployment of new information systems or hardware systems concept. Such strategies include budgeting for equipment, software, and personnel requirements as projected in the annual strategic plan for the organization.

Minimum professional qualification for an MIS analyst is Bachelor's degree in MIS, IT, Computer Science, Operations Research, or Business Analytics, or in a related field. But they also need hands-on experience for this role.

MIS Analyst Job Description Example

MIS analysts performs various functions, which is primarily involved in helping an organization meet, its strategic goals through providing direction and guiding the concerned information processes, security of data, and software application management. Their job duties may vary according to the organization they work.

The main tasks, duties, and responsibilities that commonly defines the MIS analyst's jobs are listed below:

- Carry out analysis and assessment on the organization/clients existing software and database management systems, and information technology needed to identify the shortcomings and solutions to meet clients' needs within a time and cost constraints.
- Responsible for the design and development of software and computer systems to meet the project needs within a time and cost constraints.
- Responsible for the development and maintenance of documentation and metrics to improve operational efficiency.
- Evaluate current systems and draw conclusions for integrating new technologies and services.
- Work together with other members of the organization to train and provide backup to other team members.

- Perform system monitoring, analysis, performance tuning and troubleshooting of system hardware, software networks and operating system.
- Develop system policies to authorize system, data and network access criteria and implement security controls to secure computer system and ensure data integrity.
- Provides IT supports to the user community by interacting with them and resolves their hardware and software problems.



CHECK YOUR PROGRESS

Q 6: Which is the best conflict management strategy?

Q 7: What is the main difference between MIS analyst and System analyst?

Q 8: Mention any two duties performed by MIS Analyst.



3.10 LET US SUM UP

- A system analyst is an important part of an organization in project management.
- A system analyst analyses the requirements of an organization and the problems associated with those requirements and designs suitable solutions for it.
- During the design and planning phase of the solution,the system analyst may perform the role of consultant or supporting expert or agent of change.
- To become an efficient analyst, one must master interpersonal skills like communication management and rational thinking.
- Other skills required for system analysts are analytical skills and conflict resolution skills.
- An MIS analyst is also a system analyst, but he basically deals with management information systems.



3.11 FURTHER READING

1. Award, E.M. (2010). System Analysis and Design. Galgotia Publication Pvt. Ltd.
2. Laudon, K.C., & Laudon, J.P. (2015). Management Information System. Pearson Education India.



3.12 ANSWERS TO CHECK YOUR PROGRESS

Ans to Q No 1: A system analyst is a person who conducts a methodical study and evaluation of an activity such as business to identify its desired objectives in order to determine procedures by which these objectives can be gained.

Ans to Q No 2: An agent of change can be defined as a person who serves as a catalyst for change, develops a plan for change, and works with others in facilitating that change.

Ans to Q No 3: c. Sportsmanship

Ans to Q No 4: Roles of system analyst are – a. Persuader b.Investigator c. Architect d. Manager

Ans to Q No 5: A bachelor's degree in computer or information science; however, some systems analysts hold bachelor's degrees in another discipline and add computer experience through specialized classes.

Ans to Q No 6: Collaborate or Problem Solving is the best conflict management strategy.

Ans to Q No 7: The main difference is MIS analyst works on only management information system but system analyst works on all system.

Ans to Q No 8: Responsible for the development and maintenance of documentation and metrics to improve operational efficiency.



3.13 MODEL QUESTIONS

Q 1: Explain the role and responsibilities of a system analyst in an organization.

Q 2: Why Interpersonal skillsare important for system analyst?

Q 3: Explain the multifaceted role of system analyst with an example.

Q 4: How to resolve conflict during project management?

Q 5: How is MIS analyst different from system analyst?

Q 6: Suppose you are a system analyst of a software company and the company has been assigned with a project of designing a student information system for a university. What will be your role in this project as a system analyst?

*** ***** ***

UNIT 4: SYSTEM PLANNING AND INITIAL INVESTIGATION

UNIT STRUCTURE

4.1 Learning Objectives

4.2 Introduction

4.3 System Planning and Initial Investigation

- 4.3.1 Bases for Planning in System Analysis

4.4 Dimensions in Planning

4.5 MIS Planning

- 4.5.1 Strategic MIS Planning
- 4.5.2 Managerial MIS Planning
- 4.5.3 Operational MIS Planning

4.6 Initial Investigation

4.7 Needs Identification

4.8 Determining the Requirements

4.9 Steps for Initial Investigation

- 4.9.1 Problem Definition and Project Initiation
- 4.9.2 Background Analysis
- 4.9.3 Fact Finding
- 4.9.4 Fact Analysis
- 4.9.5 Determination of Feasibility

4.10 Let Us Sum Up

4.11 Further Reading

4.12 Answers to Check Your Progress

4.13 Model Questions

4.1 LEARNING OBJECTIVES

After going through this unit, you will be able to –

- understand the utility of planning before starting the actual development
- explain how to do effective planning
- investigate on the various needs, data, etc. of the client

4.2 INTRODUCTION

In the previous unit, we learnt about the roles of system analyst. We also learned about the academic and professional qualification and the multifaceted role of system analyst. In this unit, we will learn about system planning and initial investigation. We will learn about topics like MIS planning, determining the requirements and steps for initial investigation. In the next unit, we will explore the topics related to information gathering.

4.3 SYSTEM PLANNING AND INTIAL INVESTIGATION

The most significant and critical phase of any project life cycle is *system planning and initial investigation.* An erroneous investigation and planning might sometimes lead to system failure. The main intention behind the initial investigation is to analyze and check the feasibility of the users' requests. To carry out an effective system investigation, a detailed MIS plan including the steps to be taken, the people to be questioned and the outcome must be made. This plan will result in the successful implementation of the information system.

4.3.1 Bases for Planning in System Analysis

Planning is done to help the organization grow and also to sustain adverse risk situations. In system analysis, planning the information systems mainly includes setting up goals to be achieved and formulating tasks and strategies to attain them. It is important because information associated with the system is a major resource involving finances and it needs to be managed. There is an emergent need for long-term planning in information systems.

4.4 DIMENSIONS IN PLANNNG

Planning information systems in business has to be done keeping in mind certain factors like:

- Good return with moderate or high interest rates
- The price hike scenario
- The idea of business expansion

- Shortage of various resources and other governing limitations
- Increased productivity

4.5 MIS PLANNING

MIS planning is done to improve the overall working process of an organization. The planning is done with respect to two horizons:

- *Time* which states whether the planning is short, medium or long ranged.
- *Focus* specifies whether priority is strategic, managerial or operational.

4.5.1 Strategic MIS Planning

This kind of planning is used to define the main goals in terms of expected outcomes, strategies to achieve those goals, and plans to implement the strategies. The objectives must be defined following the organization's needs and requirements. And the strategies are some directives or guidelines to carry out the implementation plan. While devising the MIS strategic plan, some points need to be looked after as like:

- if any objective or strategy can be drawn from the corporate strategic plan.
- the one who will assess the plan.
- the duration for completion of the plan.
- the main attention of the plan.

4.5.2 Managerial MIS Planning

The strategic plans are blended with the operational plans by managerial MIS planning. To achieve the long-term plans, this kind of planning describes how strategies and policies should be worked.

4.5.3 Operational MIS Planning

To specify system requirements, the MIS operational plan needs more user involvement. The system should be developed as per

the organizational MIS objectives as defined in the corporate plan. The identification of the system's main and important applications at the time of development is crucial.

Therefore, planning for developing a system is of major concern in an organization. While developing the system from the specified action plans, the goal is the attainment of the corporate strategic objectives. Conceptualizing and systematizing the overall planning process makes it easier to gain the support of the upper, mid, and ground-level management.



CHECK YOUR PROGRESS

Q 1: Which phase is the most critical in project development life cycle?

Q 2: List the major areas of concern while planning.

Q 3: What are the needs of MIS planning?

Q 4: Mention the names of the different levels where MIS planning is done.

Q 5: Why is it mandatory to conceptualize and systematize the overall planning process?

4.6 INITIAL INVESTIGATION

The software development lifecycle model starts by finding out the requirements of the user. Users might request for removal or up-gradation of the existing system. However, not all requests can be approved as they may not be feasible. So, some standard procedures are needed to handle such requests. Initial investigation is one such procedure. The initial investigation mainly aims at determining the feasibility of the user's need. Before recommending a user's request, various strategies like asking, extracting the present system's information, and prototyping are used to investigate to conclude.

4.7 NEEDS IDENTIFICATION

To have a successful system, the problem should be very well stated, investigated and the best solution has to be provided. This will be

accomplished only when the actual needs of the customer or users are identified and reported. The users' requirements must be evaluated wisely before being evaluated. Only after quality evaluation, an analyst will get an idea about enhancing a new system or developing a new system. The analyst can request the users to fill up some form that includes the objectives and expected benefits. Needs identification gives a clear picture of the system.

4.8 DETERMINING THE REQUIREMENTS

Determining the actual requirements of the users is a challenging task as the users might not be able to express their needs. Moreover, the users qualitatively express their needs, but the analyst technically needs them. Different ways can be used by the users to describe their requirements:

- **Kitchen Sink:** For inexperienced users, this strategy is used where the user puts up everything in the requirement definition.
- **Smoking Strategy:** This strategy is used for experienced users. Requests that are manageable, realistic, and achievable are only considered. A smokescreen is set up by asking for various system features, out of which only a few are required.
- **Same Thing Strategy:** This strategy tries to tackle the users who themselves do not know what they want and keeps on requesting the same thing in a better way. The analysts have a tough time dealing with such requests and the success rate is very low because of the non-clarity of the actual need.



CHECK YOUR PROGRESS

Q 6: What is the main aim of initial investigation?

Q 7: What are the strategies being used to conclude the initial investigation?

Q 8: Name the strategies being used to determine user's requirements.

4.9 STEPS FOR INITIAL INVESTIGATION

The steps followed while performing the initial investigation are as follows:

4.9.1 Problem Definition and Project Initiation

The initial investigation starts with finding out the user's requirement that has made his request for system change. While defining the problem statement, it should be made sure that the user's problem is well understood and the input, as well as the output in terms of the user's request, must be clearly specified. The main focus should be on the attainment of the results as per the user's requirement. Problem definition must be clearly and correctly specified as follows:

- function of the system, its input, output, constraints (if any), and the environment variables
- skillset required, manpower
- individual components of the system along with their utility and overall flow of activities

4.9.2 Background Analysis

Once the initiation starts, a background analysis of the customer's organization is to be done. The system analyst tries to extract the basic information like the physical working process, people associated with the organization, etc. After the analysis, the analyst prepares a detailed chart of the people with their associated functions.

4.9.3 Fact Finding

After analyzing the background information, the next step in the investigation is to find facts. The analyst, through various means, tries to collect data on the input, output, and cost of the existing system. The analyst can retrieve this information by the following ways:

- **Review of written documents:**For some problems, various documentations like manuals, company studies, government publications are searched and reviewed. However, such a method is not fruitful as it is time-consuming, costly, pieces of information may be outdated. Updated and good manuals might save search time.
- **On-site observations:** It is one of the best methods of fact-finding as the analyst can visit the site all by himself and observe and analyze all the ongoing activities, operations, and functions of the system. The analyst can also interact with the people around to extract all the relevant and meaningful information. This way of fact-finding is fruitful as the information found is unbiased. The analyst gets a clear picture of the actual scenario of the system rather than relying on the documents and manuals. However, this approach takes time and it greatly depends on the quality of inferences drawn by the analyst.
- **Interview and questionnaires:**The third method to find facts is through interviews and questionnaires. By surveying and interviewing, the analyst knows people's awareness, feelings, or motivations. In this approach, information is gathered by posing a set of questions among a large number of people and then analyzing the data.

4.9.4 Fact Analysis

The facts obtained from the previous stage must be analyzed. The objective of this phase is to organize and evaluate the data to generate some formal reports to get the final approval from the user. The fact analysis tools those used includes:

- **Input/ Output Analysis:** Input/output analysis tries to find out the features associated with the inputs and outputs of a given system. Various diagrams like Flow-charts and Data Flow Diagrams (DFD) are used to perform such kinds of analysis. A flowchart is a pictorial representation of the overall flow of information throughout the system using various symbols,

shapes, and words. And DFD is used to describe the modeling of the entire process in detail where the system might have some constraints.

- **Decision Tables:** Diagrammatic representation of the data flow is not possible for complex processes. So, decision tables are used to represent the data flow within a system. Decision tables are easy and simple data analysis tools. It serves as a bridge between technical and non-technical staff.
- **Structure Chart:** A structure chart is a hierarchical approach for fact analysis. It starts from preparing a single input-output processing chart, tracing out the module associated with the hierarchy chart, and finally finding the data elements along the line linking the module to a higher level. This tool is a brilliant way to keep track of the data collected for a system.

4.9.5 Determination of Feasibility

After the detailed analysis, the analyst has a thorough knowledge of the system. With various information at hand like the interview records, updated system documentation, flowcharts, and pros-cons of the system, the analyst then evaluates the feasibility of the proposal and checks for the possibility of alternative solutions.



CHECK YOUR PROGRESS

Q9: What should problem definition specify?

Q10: What is the output of the background analysis step?

Q11: What are the methods used toanalyse facts?



4.10 LET US SUM UP

- In system analysis, planning the information systems mainly includes setting up goals to be achieved and formulating tasks and strategies to attain them.

- System planning and the initial investigation are very essential to avoid system failures.
- The initial investigation tries to check for the feasibility of the user's requirements. So, the needs and the requirements of the user must be identified to develop the system correctly.
- Flow-charts and Data Flow Diagrams (DFD) are used to find out the features associated with the inputs and outputs of a given system.
- Decision Tables are used to represent the data flow within a system.
- Structure chart is used for fact analysis.



4.11 FURTHER READING

1. Awad, E. M. (1985). Systems analysis and design. Homewood, Ill: R.D. Irwin.
2. Chauhan A & Mishra R. (2009). System Analysis and Design. Pragati Prakashan.



4.12 ANSWERS TO CHECK YOUR PROGRESS

Ans to Q No 1: System planning and initial investigation is the most critical in project development life cycle.

Ans to Q No 2: The major areas of concern for planning includes good return with moderate or high interest rates, price hike, business expansion, shortage of various resources and other governing limitations and increased productivity.

Ans to Q No 3: MIS planning is done to improve the overall working process of an organization.

Ans to Q No 4: MIS planning is done at three levels namely strategic, managerial and strategic.

Ans to Q No 5: Conceptualizing and systematizing the overall planning process makes it easier to gain the support of the upper, mid and ground level management.

Ans to Q No 6: An initial investigation mainly aims at determining the feasibility of the user's need.

Ans to Q No 7: Strategies like asking, extracting present system's information and prototyping are used to investigate in order to conclude.

Ans to Q No 8: The strategies used for determining user's requirements include kitchen sink strategy, smoking strategy and same thing strategy.

Ans to Q No 9: Problem definition should specify function of the system, its input, output, constraints (if any), environment variables, skillset required, manpower, individual components of the system along with their utility and overall flow of activities, economics in terms of cost, value, price, etc.

Ans to Q No 10: The output of the background analysis step is a detailed chart of the people with their associated functions.

Ans to Q No 11: Input-Output analysis, decision tables and structure charts are used to analyse facts.



4.13 MODEL QUESTIONS

Q 1: Why are system planning and initial investigation important?

Q 2: What should an MIS plan include?

Q 3: Explain in detail the three levels of MIS planning.

Q 4: Why is it important to identify the needs of the user?

Q 5: Explain how can the users describe their needs?

Q 6: Discuss the various steps that are used in initial investigation.

Q 7: Which method of fact finding is efficient and why? Justify your answer.

Q 8: Why is it necessary to analyse facts in initial investigation?

*** ***** ***

UNIT 5: INFORMATION GATHERING

UNIT STRUCTURE

5.1 Learning Objectives

5.2 Introduction

5.3 Information Gathering

5.4 Types of Information

5.5 Origin of Information

5.6 Review of Literature

5.7 Procedures and Forms

5.8 Interviews and Questionaries

5.9 Let Us Sum Up

5.10 Further Reading

5.11 Answers To Check Your Progress

5.12 Model Questions

5.1 LEARNING OBJECTIVES

After going through this unit, you will be able to –

- explain the need of information gathering for system design
- identify different types of information for system design
- describe the importance of literature review
- describe about different information gathering techniques

5.2 INTRODUCTION

In the previous unit, we have learnt about system planning and initial investigation. We have learned in detail about MIS planning and steps for initial investigation. In this unit, we will learn about information gathering. Topics like types of information, origin of information, review of literature, procedure and forms are covered in detail in this unit. In the next unit, we will explore the tools used for structured analysis.

5.3 INFORMATION GATHERING

Information gathering is a part of the feasibility analysis process. It is both an art and a science. It is a science because effective information-gathering needs a proper methodology and tools. It is art too because it requires creativity to find the exact information required. To design a system, we need lots of information about the system. That's why software engineers gather and analyzes information like:

a) **Interrelationship between different system components:** To design a system, one must know how the components of the system are related to each other. The degree of cohesion and coupling between different systems components must be known.

b) **System entities and their attributes:**Entities are the concept or objects where data of a system are stored and managed, and their properties are called an attribute. We should know how to store and manage data to design a system efficiently.

c) **System operations and their interrelatedness:**Operations are a piece of code or instructions that enables the functionality of systems. We should understand the functionality of a system to build it.

d) **System business rules:** Business rules are guidelines that specify how the system should operate. These relate to data access, data flow, relationships among entities, and the behavior of system operations.

System engineer or analyst applies various methodologies to collect such information so that they can design the system more accurately and comprehensively.

5.4 TYPE OF INFORMATION

Various types of information are required for the design and development of a system. We are looking for information that will help us to define the requirement of the systems accurately. This process is known as the requirements analysis phase, and it leads to a document called

Software Requirements Specification (SRS).This information can be categorized into 3 types:

a) **Information about the Firm/Organization:** These are the guidelines for how the system should operate. This guideline includes business policies, goals, objectives, and the organization of the system.

b) **Information about the User Staff:** It is about the organization of the user staff who manages the system. It includes job functions of different staff members, their authority level, information requirement at each user level, etc.

c) **Information about the Workflow**: It refers to the workflow of the system or how the system works. It includes information like interrelationship between different system components, system entities with their attributes, system operations with interrelatedness.

5.5 ORIGIN OF INFORMATION

So, the next important aspect of the discussion is from where we will get all of this information. Most of this information can be gathered from two principal resources: (a) written documents from the organization (b) the organization's environment.

These written documents can be external such as Government policies, vendor policies, public opinion about the system, or internal such as system documentation and manuals, User reports, financial reports, etc.



CHECK YOUR PROGRESS

Q.1: In which part of SDLC, information gathering fall?

Q.2: What does Business policiesdefine?

Q.3: What is an attribute?

5.6 REVIEW OF LITERATURE

The process through which we gather information from the written document is called a literature review. A literature review surveys written

documents and any other sources relevant to a particular domain. It provides a description, summary, and critical evaluation of these works concerning the problem being investigated. The main purposes of literature review are:

- place each work in the context of its contribution to understanding the problem.
- describe the relationship of each work to the others under consideration.
- identify new ways to interpret prior works.
- resolve conflicts amongst seemingly contradictory previous works
- point the way in fulfilling a need for new and additional works towards the problem.

There is not a fixed methodology to gather information from the organization's environment. Some of the technique used for this has been mentioned below.

5.7 PROCEDURE AND FORMS

Procedural manuals can give a good picture of the system to be studied. For example: (a) how the existing system works (b) what are the assumptions (c) what information flows in and flows out (d) what are the dependencies of the external system, etc. Problems that one can encounter are the lack of updated manuals or documents, or sometimes they need for possession of the correct documents. Hence, this is just one of the means of gathering information. However, procedural forms can capture some important information like: (i) Who are the users of the forms? (ii) Do the forms have all the necessary information? (iii) How readable and understandable are they? (iv) How does the form help other users to make better decisions?

On Site Visits and Observations:

The main objective of an onsite visit is to get as close to the system as possible. The person who visits onsite is a keen observer and is knowledgeable about the activities that can occur within the system. Various types of observation methods are used:

Direct or Indirect:

The analyst can observe the subject or the system directly like how do the workers perform a job on the factory floor? Using some devices like video cameras or videotapes, indirect form of observations can be done.

Structured or Unstructured:

In a structured observation, the specific actions are recorded. For example, before a shopper buys a product, how many related products did he see before selecting the final product? An unstructured method would record whatever actions would happen at a given point in time.

5.8 INTERVIEWS AND QUESTIONNAIRES

The interview is a face-to-face interpersonal meeting designed to identify the relations and verify the information to capture information. An interview is a flexible and better tool than a questionnaire for the evaluation of the information. It is an art that requires experience in arranging the interview, setting the stage, and establishing rapport. The questions should be designed clearly, avoiding misunderstandings and carefully evaluating the responses. The disadvantage of this technique is that the preparation time required is more. Since only one person at a time can be engaged with the process, a longer time is required for gathering the results.

Types of Interviews

(a) **Structured Interview**

The skill of the interviewer helps in getting the interviewee to respond and move to the next questions without diversion. The questions are presented to the interviewee with similar wording and order.

(b) **Unstructured Interview**

In the unstructured Interview, the respondents are allowed to answer freely in their own words. The responses are not forced. They are self-revealing and personal rather than general and superficial. The interviewer has encouraged the respondents to talk freely.

Unstructured interviews provide an opportunity to delve more deeply into complex topics than is possible with surveys.

(c) **Questionnaire**

Questionnaire is a self-administered tool that is more economical and requires less skill to administer than the interview. At any point in time, unlike the interview, feedback from many respondents can be collected at the same time. Since questionnaires are usually self-administered, it is critical that questions be clear and unambiguous.



CHECK YOUR PROGRESS

Q. 4: What is literature review?

Q. 5: Mention two methods to gather information from the organization's environment.

Q. 6: Which type of interview is better for information gathering?



5.9 LET US SUM UP

- Information gathering is a part of the feasibility analysis process.
- Information can be categorized into 3 types:
  Information about the Firm/Organization, Information about the User Staff, and Information about the Workflow.
- This information can be gathered from two principal resources: - Written documents from the organization and from the organization's environment.
- There is not a fixed methodology to gather information. So, we use various methods like Literature Review, Procedure and Forms, Interviews and Questionaries to collect information



5.10 FURTHER READING

1. Awad, E. M. (1985). Systems analysis and design. Homewood, Ill: R.D. Irwin.
2. Chauhan A & Mishra R. (2009). System Analysis and Design. Pragati Prakashan.



5.11 ANSWERS TO CHECK YOUR PROGRESS

Ans to Q No. 1: Feasibility study.

Ans to Q No. 2: Business rules are guidelines that specify how the system should operate.

Ans to Q No.3: Attributes are the properties of entity by which we can describe an entity.

Ans to Q No.4: The process through which we gather information from written document is called literature review.

Ans to Q No.5: Interviews and Questionnaires.

Ans to Q No.6: Structured Interview.



5.12 MODEL QUESTIONS

Q 1. What type of information are required for the design and development of a system?

Q 2. Define literature review. Why is it required?

Q 3. How procedures and forms are useful in information gathering?

Q 4. How interview and questionnaires are used in information gathering process? Explain.

UNIT 6: TOOLS OF STRUCTURED ANALYSIS

UNIT STRUCTURE

6.1 Learning Objectives

6.2 Introduction

6.3 Data Flow Diagram (DFD)

6.3.1 DFD Symbols

6.3.2 Constructing a DFD

6.4 Data Dictionary

6.5 Structured English

6.6 Decision Tree

6.7 Decision Table

6.8 Let Us Sum Up

6.9 Further Reading

6.10 Answers to Check Your Progress

6.11 Model Questions

6.1 LEARNING OBJECTIVES

After going through this unit, you will be able to –

- describe about different structured analysis tools and techniques.
- use the process of defining and constructing a data flow diagram.
- use various notations used for designing DFDs.
- make a data dictionary for a system from the constructed DFDs.
- use decision table which can visualize a system in a tabular format.
- analyze a system by using a decision tree when complex branching occurs in a structured decision process.
- conceptualize a process in a sentence like form by using structured english tool.

6.2 INTRODUCTION

Structured analysis is a development method that allows the analyst to understand the system and its activities logically. After gathering the required information for a system, it is to analyze them with some analysis tools. Following a systematic approach, structured analysis employs some graphical as well as textual tools that analyze and refine the objectives of an existing system and design a new system specification that will be easily understandable for the user and also for the programmer. In this unit, we will discuss various analysis tools like data flow diagram, decision tree, decision table, etc. which can visualize a system to the users in a much easier way. In the next unit, we will explore the topics related to feasibility study.

6.3 DATA FLOW DIAGRAM

A data flow diagram (DFD) is a graphical representation of the flow of data in an information system. It is also termed a bubble chart. It was proposed by American software engineer Larry Constantine to express all functional requirements of a system in a pictorial form. It can depict incoming data flow, outgoing data flow, and stored data throughout the system.

The objective of DFD is to provide an overview of the transformations of the input data within a system to produce the output. A DFD is defined by IEEE as 'a diagram that depicts data sources, data sinks, data storage and processes performed on data as nodes and logical flow of data as links between the nodes.' Since graphical representations are easier to interpret, any layperson/non-technical user can also be able to understand the system details easily and clearly by going through the DFDs. It states an overview of what data a system processes, what transformations are performed, what data are stored, what results are produced, and where they flow.

6.3.1 DFD Symbols

To construct DFD for a system, a set of symbols are used with standardized notations like external entity, process, data flow, data store, output, etc. There are four common methods of notation used

in DFDs: Yourdon & De Marco, Gene & Sarson, Structured systems analysis and design methodology (SSADM), and Unified. All four methods employ the same labels with different shapes to represent the four basic elements of a DFD namely external entity, process, data store, and data flow. Figure 6.1 presents these four elements concerning four categories as mentioned above. Among these, the notations belonging to Yourdon & De Marco method are widely used to construct DFDs. Unified modeling notations are specially used for defining a system in an object-oriented concept.

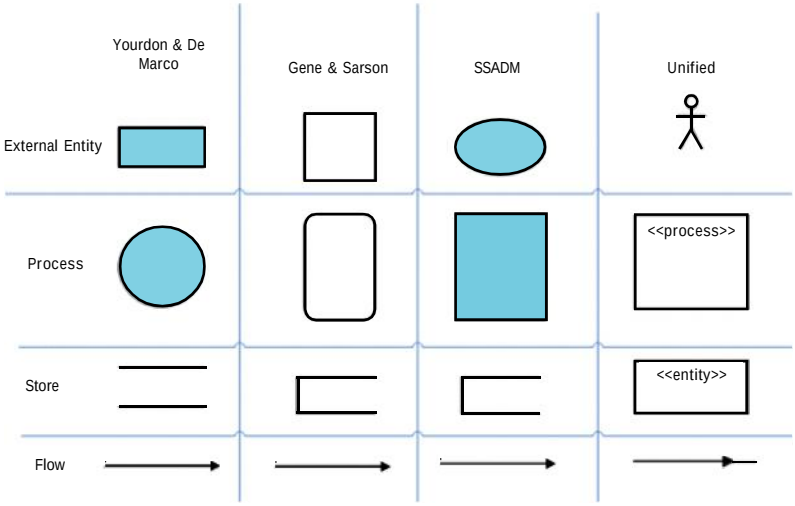


Figure 6.1: Four fundamental notations used in DFDs

External Entity

An external entity, which is also known as the terminator, source, or sink, represents an external user or an external system that feeds or receives data to and from the system. Entities are represented by rectangles with specific names. They are usually placed on the data flow diagram's edges.

Process

Process refers to an operation that can compute and manipulate the data. The corresponding data is fed either by the external user through data flow to the process or the process receives data from data sources to compute or change. Subsequently, it produces an output. Activities and actions taken on the data by processes are represented by circles.

Data Store

Datastores hold information for later use, like a file of documents from which necessary data can be retrieved for processing. Data inputs flow from the external entity to a data store through a process, while data outputs flow out of a data store to the entity through a process. It is to be noted that data flow can never be possible between the entity and the data store. There are two variants of data storage - it can either be represented as a rectangle with the absence of both smaller sides or as an open-ended rectangle with only one side missing.

Data Flow

Data Flow refers to a path by which the system's information passes from external entities to datastores through processes. With arrows along with brief labels, the DFD presents the direction of the data flow. In general, data flow is not possible in between two processes or in between two data sources. However, data movement is shown from the base of the arrow as its source towards the head of the arrow as the destination.

6.3.2 Construction of DFD

Certain standard guidelines are to be followed while creating a DFD. These include the following:

a) DFD notations should be assigned meaningful names by using the proper part of speech. For example, a process name should be a verb whereas nouns should be used for naming external entities, data storage, and data flow.

b) Each process should be numbered uniquely, and numbering should be consistent. For example, the single process in the context diagram should be numbered with 0. Subsequently, processes in 1st level DFD to be numerically assigned as 0.1, 0.2, … so on.

c) Abbreviations should be avoided in DFD notations.

d) Looping concept can't be presented in DFD.

e) A DFD refinement should be continued in a series of levels of DFDs until each process performs a simple operation.

f) The data store should not be depicted in context diagram or Level 0 DFD, but all the data stores required for a system must be included in 1st level or next level DFDs.

g) Each process should have at least one input and one output.

h) Each data store should have at least one data flow in and one data flow out.

i) Data flow should not be given in between two data stores.

Levels of Data Flow Diagrams

There are various levels of DFD ranging from simple overviews to complex, granular representations of a system or process with deeper levels, starting with level 0. These provide details about the input, processes, and outputs of a system. The most common and intuitive DFDs are level 0 DFDs, which are also known as context diagrams.

Level 0 DFD or Context Diagram

Level 0 DFD focuses on high-level system process or function and the data sources that flow to or from them. The entire system is represented by a single bubble (circle), and all the external entities that will feed the data into the system are shown in the context diagram. They are designed to be simple, straightforward overviews of a process or system. It is to be noteworthy that no data store should be included in the context diagram.

Level 1 DFD

Level 1 DFD elaborates the system by splitting the system's single process (bubble) in level 0 DFD into more detailed form showing all the high-level functions of the system and flow of data among various functions. It also includes all the data stores required for the system. However, external entities should not be a part of level 1 DFD or other deeper level DFDs.

Level 2 DFD

The level 2 DFDs present more elaboration by splitting each process of level 1 DFD into granular subprocesses (if required). Decomposing the bubble is called factoring or exploding. Each bubble can be factorized into from 3 to 7 bubbles.

Level 3 DFD

Level 3 and higher-numbered DFDs are usually not to be defined. It is mainly due to the requirements of a large amount of detail which increases the complexity of the system. Higher-level DFDs can be transformed into more specific lower-level DFDs with a deeper level of understanding.

Balancing a DFD

The data flow into or out from a process in Level 1 DFD must be matched with the subprocesses in Level 2 DFD or deeper level DFDs. It is called balancing a DFD.

Synchronous and asynchronous process

When the execution of one process depends on the data coming out from another process directly (without any data store), then that process is known as synchronous process. Otherwise, the process is termed as asynchronous where data flow is maintained between two processes via a data store.

Let us consider an example to explain DFD for a system “Online Examination System”. Online Examination System allows the user (examinee) to appear in an examination in online mode by sitting in an examination venue. This system is handled by two types of users: administrator and examinee. Accordingly, two external entities representing the admin and examinee involved in this system are shown in level 0 DFD (context diagram). The context diagram of this system is presented in figure 6.2.

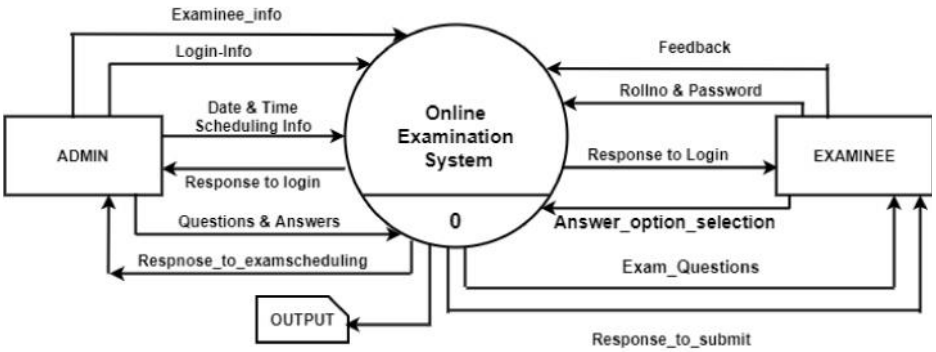


Figure 6.2 Context Diagram of Online Examination system

Further, the bubble in the context diagram is decomposed into Level 1 DFD for Admin and Examinee separately concerning their functionalities in the system. Figure 6.3 and figure 6.4 depict level 1 DFD for Admin and Examinee respectively. Processes in level 1 DFD for Examinee act as synchronous because of their inter dependability as shown in figure 6.4.

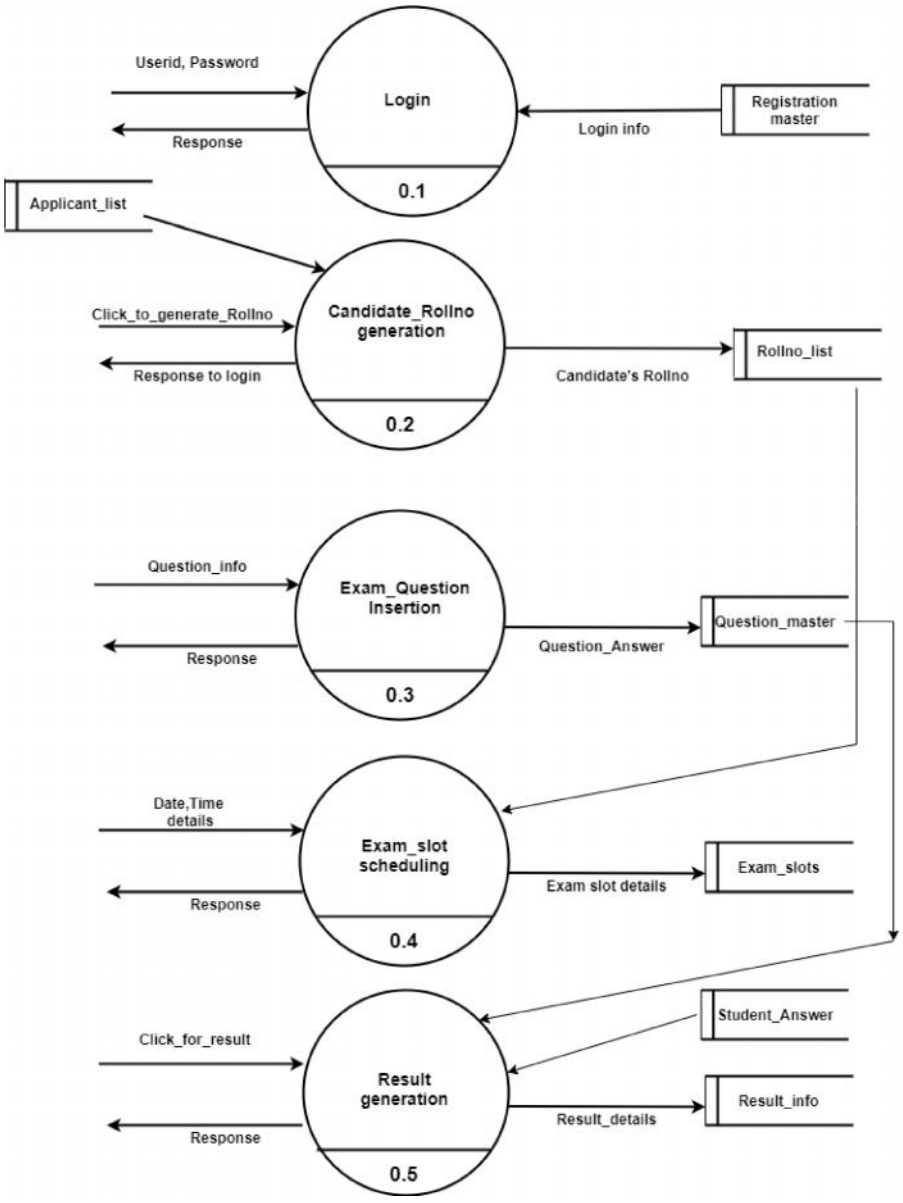


Figure 6.3Level 1 DFD for ADMIN

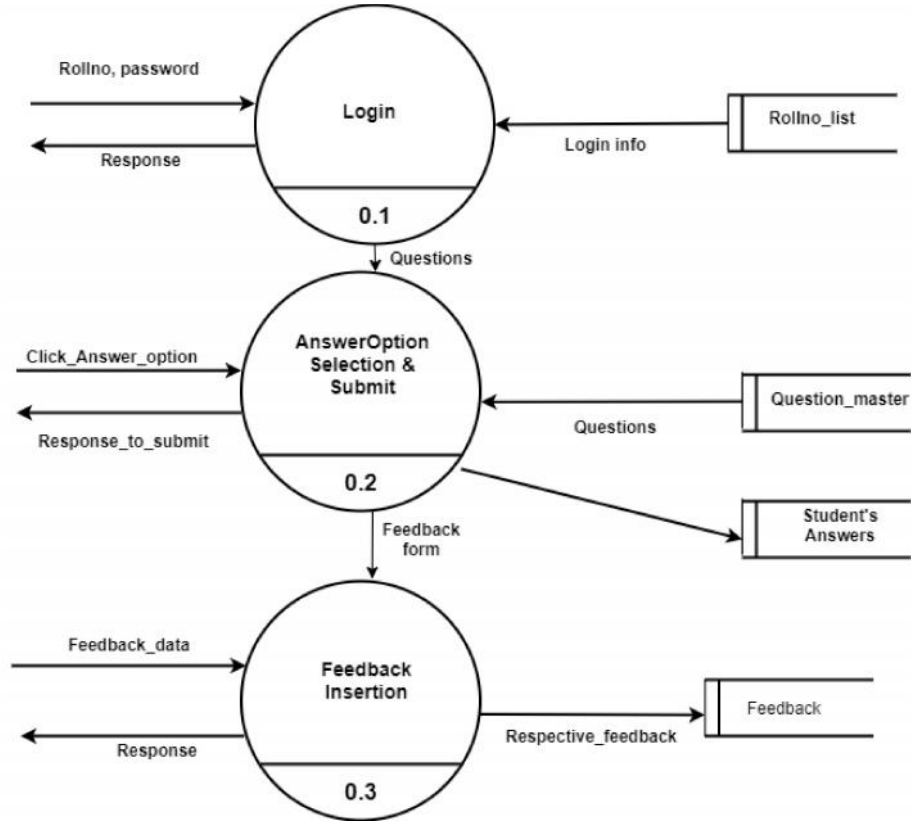


Figure 6.4 Level 1 DFD for EXAMINEE



CHECK YOUR PROGRESS

Q.1: Choose the correct options from the followings:

i) Which of the following is not a component in DFD?

A. Entities

B. Attributes

C. Process

D. Data Flow

ii) A data flow can

A. Only emanate from an external entity

B. Only terminate in an external entity

C. May emanate and terminate in an external entity

D. May either emanate or terminate in an external entity but not both

6.4 DATA DICTIONARY

The first step in analysing a system of interactive objects is to identify each one and its relationship to other objects. This process is called data modelling and results in a picture of object relationships. After each data object or item is given a descriptive name, its relationship is described, or it becomes part of some structure that implicitly describes the relationship. The type of data such as text or image or binary value is described, possible predefined default values are listed, and a brief textual description is provided in the data dictionary.

A data dictionary is a collection of descriptions of the data objects or items in a data model for the benefit of programmers and others who need to refer to them. It is basically a centralized structured repository of data elements in the system. It stores the descriptions of all DFD data elements, i.e., details and definitions of data flows, data stores, data stored in data stores, and the processes. It is often referenced as a meta-data (data about data) repository. It is created along with the DFD model of the software program and is expected to be updated whenever DFD is changed or updated.

It makes it easier for users and analysts to use data properly as well as to understand the format of data objects. As a centralized collection of information about data, data dictionary stores the meaning and origin of data, its relationship with other data, the data format for usage, etc.

Data Dictionary Uses:

The data dictionary may be used for the following reasons:

- Provide documentation.
- Eliminate redundancy.
- Validate the data flow diagram.
- Provide a starting point for developing screens and reports.
- To develop the logic for DFD processes.

Each data flow should be defined with descriptive information and its composite structure or elements.It may include the following information:

- ID - identification number.
- Label, the text that should appear on the diagram.
- A general description of the data flow.
- The source of the data flow which may be an external entity, a process, or a data flow coming from a data store.
- The destination of the data flow.
- Type of data flow, either a record entering or leaving a file.
- The name of the data structure or elements.
- An area for further comments and notations about the data flow.

When developing programs that use the data model, a data dictionary can be consulted to understand where a data item fits in the structure, what values it may contain and what the data item means in real-world terms. For example, a bank or group of banks could model the data objects involved in consumer banking. They can provide a data dictionary for a bank's programmer. The data dictionary would describe each of the data items in its data model for consumer banking, such as "Account holder" and "Available credit."

A data dictionary stores the following information:

Name	**Description**
Name	Name generally includes primary name of all composite data or control item available, name of external entity or data store.
Aliases	Any other word used in place of Name
Where or How it's used?	Data dictionary generally gives information about where and how data or control items are used which may include as an input/ output to process.
Description	A notation for representing content

- Data structures are a group of smaller structures and elements.
- An algebraic notation is used to represent the data structure.

Algebraic Notation

The symbols used are:

- = sign, meaning “consists of”.
- + sign, meaning “and”.
- Braces {} meaning repetitive elements, a repeating element or group of elements.
- Brackets [] for an either/or situation.
- The elements listed inside are mutually exclusive.
- Parentheses () for an optional element.

Data Structure Example

Customer Order = Customer Number + Customer Name + Address +Telephone + Catalog Number + Order Date +{Order Items} + Merchandise Total + (Tax) +Shipping and Handling + Order Total + Method of Payment +(Credit Card Type) + (Credit Card Number) +(Expiration Date)

Data dictionary components

Specific contents in a data dictionary can vary. In general, these components are various types of metadata, providing information about data.

- Data object listings (names and definitions)
- Data element properties (such as data type, unique identifiers, size, nullability, indexes and optionality)
- Entity Relationship (ER)Diagrams
- System-level diagrams
- Reference data
- Missing data and quality-indicator codes
- Business rules (such as for validation of data quality and schema objects)

Example

- Address = House No + (Street / Area) + City + State
- Course ID = Course Number + Course Name + Course Level + Course Grades

Pros and cons of data dictionaries:

Data dictionaries can be a valuable tool for the organization and management of large data listings. Fundamentally, data dictionary includes the following pros:

- Provides organized, comprehensive list of data
- Easily searchable
- Can provide reporting and documentation for data across multiple programs
- Simplifies the structure for system data requirements
- No data redundancy
- Maintains data integrity across multiple databases
- Provides relationship information between different database tables
- Useful in the software design process and test cases

Though they provide thorough listings of data attributes, data dictionaries may be difficult to use for some users. Other cons include:

- Functional details not provided
- Not visually appealing
- Difficult to understand for non-technical users



CHECK YOUR PROGRESS

Q.2: Select the correct option for the following questions:

i) The data dictionary in SDLC contains descriptions of

(a) DFD elements (b) ER Diagram

(c) Class Diagram (d) Decision table

ii) A data dictionary is useful as

(a) it is a documentation aid

(b) it assists in designing input forms

(c) it contains all data in an application including temporary data used in processes

(d) it is a good idea in system design

6.5 STRUCTURED ENGLISH

Structured English is derived from structured programming language that gives a more understandable and precise description of the process. Structured English is a tool for structured analysis that refers to the use of the English language with the syntax of structured programming to communicate to non-technical users by breaking down the program into logical steps using straight-forward English words. This tool aims to get the benefits of both programming logic and natural language. Programming logic helps to achieve precision, while natural language helps with the familiarity of the spoken word.

Most programmers are unaware of the large picture of software due to which they only rely on what their managers tell them to do. It is the responsibility of higher software management to provide accurate information to the programmers to develop accurate yet fast code. Different methods which present a programming logic graphically might be interpreted differently by different people. Structured English uses plain English words in the structured programming paradigm. It is not the ultimate code but a kind of description of what is required to code and how to code it. The following are some tokens of structured programming:

IF-THEN-ELSE, DO-WHILE-UNTIL, etc.

These tokens are used by analysts and designers of the software with Structured English. It is nothing but the description of what is required to code and how to code it. Structured English helps the programmer to write error-free code with the following.

- It is best used when sequences and loops in a program must be considered and the problem needs sequences of actions with decisions.
- It does not have strict syntax rule. It expresses all logic in terms of sequential decision structures and iterations.

For example, the following sequence of actions are written using Structured English

```
 if customer pays advance
   then
   Give 5% Discount
else
   if purchase amount >=10,000
      then
         if the customer is a regular customer
            then Give 5% Discount
         else No Discount
      end if
   else No Discount
end ifend if
```

Writing Structured English:

In order to write structured English, the following conventions can be used:

1. Express all logic in terms of one of four types-sequential structure, decision structure, case structure, and iteration.
2. Use capitalized keywords like IF, THEN, ELSE, DO, DO WHILE, DO UNTIL & PERFORM.
3. Indent blocks of statements to present their hierarchy (nesting) clearly.
4. When words or phrases are defined in a data dictionary, underline those words or phrases to signify that they have a specialized meaning.
5. Be careful when using “and” and “or” and avoid confusion when distinguishing between “greater than” and “greater than or equal to” and like relationships “A and B” means A and B whereas “A or B” indicates either A or B.

Example

Let us consider another example of Customer Authentication in the online shopping environment. This procedure to authenticate customer can be written in Structured English as:

```
Enter Customer_Name
SEEK Customer_Name in Customer_Name_DB file
IF Customer_Name found THEN
Call procedure USER_PASSWORD_AUTHENTICATE()
ELSE
PRINT error message
Call procedure NEW_CUSTOMER_REQUEST()
ENDIF
```

The code written in Structured English is more like day-to-day spoken English. It cannot be implemented directly as a code of software. Structured English is independent of programming language.

The objectives of using Structured English are as follows:

a) to describe computational procedures reasonably precisely which can be understood by any user;

b) to expand a DFD so that a user can understand it;

c) to develop algorithms corresponding to processes in a DFD;

d) to ease writing programs for DFDs.



CHECK YOUR PROGRESS

Q.3: Find out the correct option for the following Multiple choice questions:

i) Structured English statements must be

a. short and clear

b. specified quantitatively

c. specified qualitatively

d. detailed and descriptive

ii) Select correct decision structures from the following:

A. if total marks>=75

then

enter distinction in student record

end if

B. if total marks >=50
then
enter pass in student record
else
enter fail in student record
end if
C. if total marks >=60
then
enter first class in student record
else if total marks>= 50
then
enter second class in student record
else
enter fail in student record
end if
end if
D. if attendance <30%
then
Do not admit in examination
else

a. B and C b. A and B
c. C and D d. A and D

6.6 DECISION TREE

Decision tree is a tree like structure that represents the various conditions and the subsequent possible actions. It also shows the priority in which the conditions are to be tested or addressed. Each of its branches stands for any one of the logical alternatives and because of the branch structure, it is known as a tree. Decision trees are employed when complex branching occurs in a structured decision process. Although it derives its name from natural trees, but it is drawn keeping root of the tree on the left side and branches coming out from the root to the right side. In this way, it

allows the analyst to write on the branches to describe conditions and actions of a logic. In constructing a decision tree, a square is used to indicate an action and a circle is used to represent a condition.

The decision sequence starts from the root of the tree that is usually on the left of the diagram. The path to be followed to traverse the branches is decided by the priority of the conditions and the respectable actions. A series of decisions are taken, as the branches are traversed from left to right. The nodes are the decision junctions. After each decision point there are next set of decisions to be considered. Therefore, at every node of the tree represented conditions are considered to determine which condition prevails before moving further on the path.

This decision tree structure is very beneficial to the analyst. The first advantage of decision tree is that the analyst will be able to depict all the given parameters in a logical format which enables the simplification of the whole decision process with a very remote chance of committing an error in the decision process as all the options are clearly specified in one of the simplest manner.

Secondly, it also helps the analyst about those decisions, which can only be taken when couple or more conditions should hold true together for there may be a case where other conditions are relevant only if one basic condition holds true. A schematic structure of the decision tree is shown in the figure 6.5.

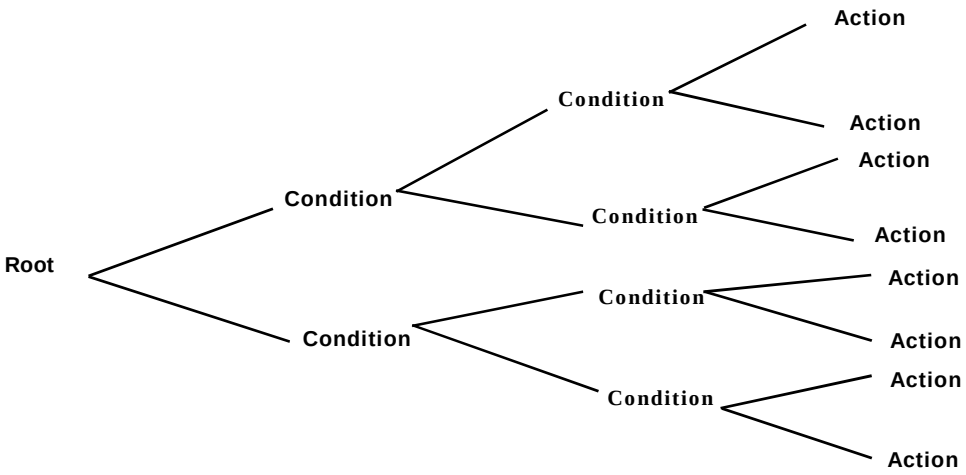


Figure 6.5:Schematic structure of the decision tree

Action

In our day-to-day life, many a times we have to face various complex cases where the most appropriate action under several conditions cannot be determined easily and for such a case a decision tree is a great aid. Hence, this decision tree representation is very effective in describing the business problems involving more than one dimension and parameters.They also point out the required data, which surrounds the decision process. All the data used in the decision making should be first described and defined by the analyst so that the system can be designed to produce correct output data.

Let us consider an example. The discount policy of a saree manufacturer for his customers is fixed as described in the following. According to the policy, the saree manufacturer offers discount to his customers based on the type of customer and size of their order. For the individual, only if the order size is 12 or more, the manufacturer gives a discount of 50% and for less than 12 sarees the discount is 30%. Whereas in case of shopkeeper or retailers, the discount policy is different. If the order is less than 12 then there is 15% discount. For 13 to 48 sarees order, the discount is 30%, for 49 to 84 sarees 40% and for more than 85 sarees the discount is 50%. The decision policy for discount percentage can be put in the form of a decision tree which is presented in the following figure 6.6.

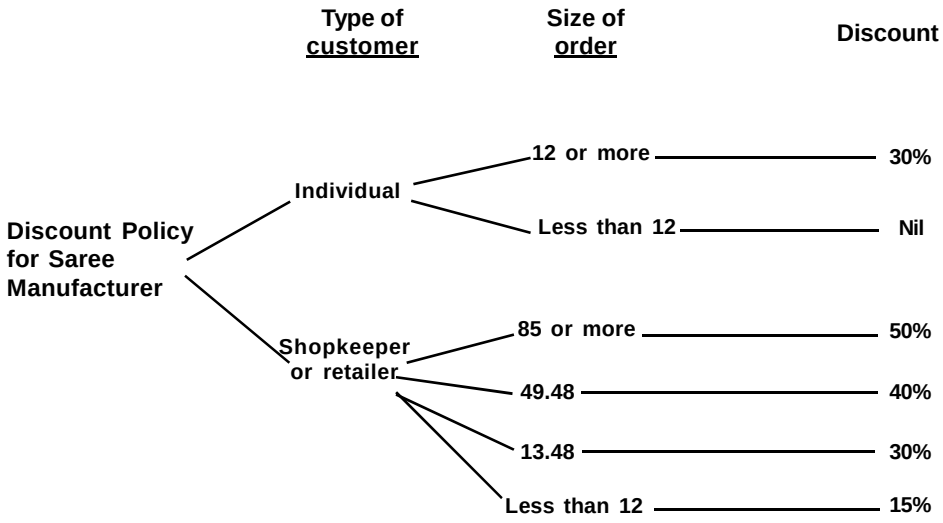


Figure 6.6: Decision tree presenting the discount policy of Saree Manufacturer

The decision trees are not always the most appropriate and the best tool for the decision making process. Representing a very complex system with this tool may lead to a huge number of branches with a similar number of possible paths and options.For a complex problem, analysing various situations is very difficult and can confuse the analyst.

Decision tree is a method for defining complex relationships by describing decisions and avoiding the problems in communication. Through pictorial presentation, it shows alternative actions and conditions within horizontal tree framework. Thus, it depicts which conditions to consider first, second, and so on.

Further, decision tree depicts the relationship of each condition and their permissible actions as shown in figure 6.7. A square node indicates an action and a circle indicates a condition. It forces analysts to consider the sequence of decisions and identifies the actual decision that must be made.

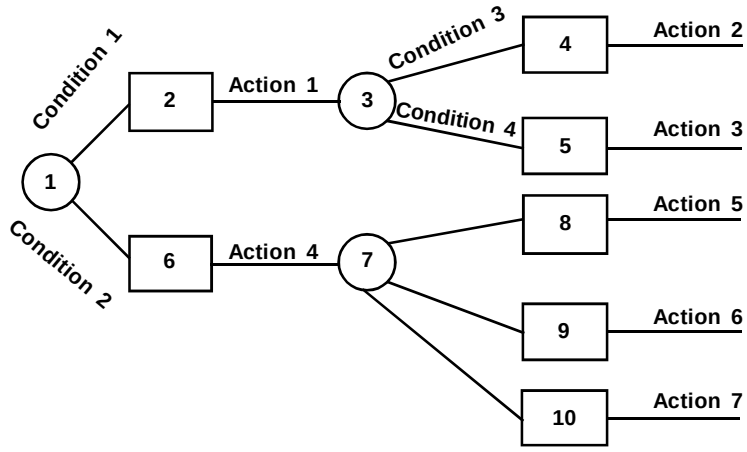


Figure 6.7:The relationship of condition and its corresponding action in decision tree

The major limitation of a decision tree is that it lacks information in its format to describe what other combinations of conditions you can take for testing. It is a single representation of the relationships between conditions and actions. Another example of a decision tree is presented in figure 6.8 describing discount policy of a super market store.

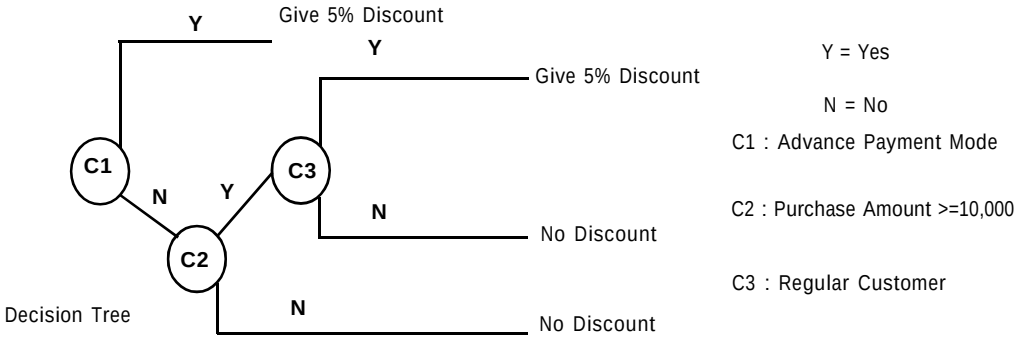


Figure 6.8: Decision tree describing discount policy of a Super market store.

6.7 DECISION TABLE

A decision table is a tabular representation of the logic of a problem. It specifies the possible conditions and the resulting actions to be taken to address them, in a structured tabular format.

It is a powerful tool to debug and prevent errors. It helps to group similar information into a single table and then by combining tables it delivers easy and convenient decision-making.

Decision tables are a method of describing the complex logical relationship in a precise manner which is easily understandable.

- It is useful in situations where the resulting actions depend on the occurrence of one or several combinations of independent conditions.
- It is a matrix containing row or columns for defining a problem and the actions.

A decision table consists of four parts: condition, condition alternatives, action and action alternatives. This tabular structure of a decision table is shown in figure 6.9. Condition stubs include conditions relevant to decision, action stubs specify actions that result from a given set of conditions and rules specify which actions are to be followed for a given set of conditions.

Conditions and Actions	Rules
Conditions	Condition Alternatives
Actions	Action Entries

Figure 6.9: Four sections of a decision table

Creating Decision Table:

To create a decision table, four fundamental steps must be followed:

- Identify all possible conditions to be addressed,
- Determine actions for all identified conditions,
- Create Maximum possible rules,
- Define action for each rule.

It is to be mentioned that decision table should be verified by end-users and can lately be simplified by eliminating duplicate rules and actions.

Constructing a Decision Table:

PART 1. FRAME THE PROBLEM.

- Identify the conditions (decision criteria). These are the factors that will influence the decision.
- Identify the range of values for each condition or criteria.
- Identify all possible actions that can occur.

PART 2. CREATE THE TABLE.

- Create a table with 4 quadrants. (Put the conditions in upper left quadrant and place the actions in lower left quadrant)
- List all possible rules. Put the rules in the upper right quadrant.
- Enter actions for each rule. (In the lower right quadrant, determine what, if any, appropriate actions should be taken for each rule)

Let us take an example of super market store. We begin by identifying all conditions that may be considered in purchasing items from the store and their respective possible challenges. We include all possible problems under column conditions and the corresponding actions under column Actions. Table 1 presents the decision table explaining this super market problem.

Table 1: The decision table for super market problem

CONDITIONS	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**
Purchase amount >Rs 10,000/-	Y	N	N	N
Purchase amount <= Rs 10,000/-	-	Y	Y	N
Regular Customer	-	Y	N	-
ACTIONS				
Give 5% discount	X	X	-	-
Give no discount	-	-	X	X

The entries in the decision table are given by decision rules which define the relationships between combinations of conditions and courses of action. In rules section,

- Y shows the existence of a condition.
- N represents the condition, which is not satisfied.
- A blank - against action states it is to be ignored.
- X (or a check mark will do) against action states it is to be carried out.



CHECK YOUR PROGRESS

Q.4: Choose the correct option for the following questions:

i) ..... is a tabular method for describing the logic of the decisions to be taken.

A. Decision tables

B. Decision tree

C. Decision Method

D. Decision Data

ii) A decision table is preferable when the number of

A. conditions to be checked in a procedure is small

B. conditions to be checked in a procedure is large

C. actions to be carried out are large

D. actions to be carried out are small

Guidelines for Selecting Appropriate Tools

We can use the following guidelines for selecting the most appropriate tool that would suit our requirements to analyse a system-

- Use DFD at highor low level analysis for providing good system documentations.
- Use data dictionary to simplify the structure for meeting the data requirement of the system.
- Use structured English if there are many loops and actions are complex.
- Use decision tables when there are a large number of conditions to check and logic is complex.
- Use decision trees when sequencing of conditions is important and if there are few conditions to be tested.



6.8 LET US SUM UP

- Structured analysis phase employs various tools like DFD, decision table, decision tree, data dictionary, structured English, etc. to present a system in an easiest way.
- As a graphical notation, DFD presents data sources, data sinks, data storage and processes performed on data as nodes, and logical flow of data as links between the nodes.
- A decision table is defined as a table that contains all the possible conditions for a problem and the corresponding results based on

condition rules that connect condition with results. It is composed of rows and columns in the form of a matrix and matrix is formed in four quadrants.

- Decision trees are a method for defining complex relationships by describing decisions and avoiding the problems in communication. A decision tree is a diagram that shows alternative actions and conditions within horizontal tree framework.
- Structure English is derived from structured programming language which gives more understandable and precise description of process. It is based on procedural logic.
- The data dictionary is an organized listing of all data elements that are pertinent to the system, with precise, rigorous definitions so that both user and system analyst will have a common understanding of inputs, outputs, components of stores and intermediate calculations.



6.9 FURTHER READING

1. Awad, E.M. (2010). System analysis and Design. Galgotia Publications Pvt. Ltd.
2. Wasson, C.S. (2005). System analysis, design and development : concepts, principles and practices (Vol. 22). John Wiley & Sons.



6.10 ANSWERS TO CHECK YOUR PROGRESS

Ans to Q No 1: i) B ii) C

Ans to Q No 2: i) a ii) c

Ans to Q No 3: i) a ii) b

Ans to Q No 4: ii) B



6.11 MODEL QUESTIONS

Q 1: Explain how the DFD model can help to understand the working of a software system.

Q 2: How do you define data flow diagram? Name the fundamental notations used to draw a data flow diagram.

Q 3: Draw context diagram for Online Tourism Management system.

Q 4: Give the context diagram and level 1 DFD for E-shopping system.

Q 5: Write the main advantages of using decision table in designing a problem. Explain with an example.

Q 6: What is the purpose of using data dictionary? What are the elements of data dictionary?

Q 7: How does structured English support to describe a structured program? Explain with a suitable example.

Q 8: Prepare the data dictionary for E-shopping system.

*** ***** ***

UNIT 7: FEASIBILITY STUDY

UNIT STRUCTURE

7.1 Learning Objectives

7.2 Introduction

7.3 System Performance Definition

7.4 Statements of Constraints

7.5 Identification of Specific System Objectives

7.6 Description of Outputs

7.7 Feasibility Consideration

- 7.7.1 Economic Feasibility
- 7.7.2 Technical Feasibility
- 7.7.3 Behavioural Feasibility

7.8 Steps in Feasibility Analysis

7.9 Feasibility Report

7.10 Let Us Sum Up

7.11 Further Reading

7.12 Answers to Check Your Progress

7.13 Model Questions

7.1 LEARNING OBJECTIVES

After going through this unit, you will be able to –

- define system performance
- list the objectives of specific system
- explain how to conduct the feasibility study
- describe the different steps in feasibility analysis
- explain how to prepare the feasibility report

7.2 INTRODUCTION

An initial investigation in a proposal for a system to be developed can determine the feasibility for an alternative system. This proposal is then presented to the user for any review. Once it is approved, a feasibility study is required to describe and evaluate candidate systems and provides

for the selection of the best system. This best system can fulfil system performance requirements. In this unit, a brief explanation on feasibility analysis of a system will be stated. Further, we will discuss different steps which are to be carried out during feasibility analysis and also the process of constructing a feasibility report in this unit.

7.3 SYSTEM PERFORMANCE DEFINITION

The performance of a system can be defined by describing its outputs in a user-acceptable format and also aiming at a higher level of detail than what was observed in the initial investigation. The system performance can be attained by the following 3 steps:

1. Statement of constraints
2. Identification of specific system objectives
3. Description of outputs

7.4 STATEMENTS OF CONSTRAINTS

Constraints refer to some factors that can bound the solution of the problem. Some constraints are identified during the initial investigation and are discussed with the user.

A system has three basic constraints:

- **Specific structure and behaviour:** A system must have some structure and behaviour which is designed to achieve a predefined objective.
- **Interconnectivity and interdependence:** These two constraints must exist among the system components.
- **Objectives of the organization:** The objectives of the organization have a higher priority than the objectives of its subsystems.

7.5 IDENTIFICATION OF SPECIFIC SYSTEM OBJECTIVES

After finding out the factors that limit the solution, the system analyst proceeds to identify system's specific performance objectives. These objectives are derived from the general objectives as specified at the end

of initial investigation. The system's benefits are analysed and transformed them into measurable objectives.

7.6 DESCRIPTION OF OUTPUTS

The final step is to describe the outputs required by the customer. An actual sketch of the format and contents of the reports are prepared here. Further, a specification of the media used, their frequency and the size and number of copies required are organizedat this point. Finally, the analyst can go for evaluating the feasibility of candidate system in order to produce these outputs.

7.7 FEASIBILITY CONSIDERATIONS

Feasibility study can be considered as preliminary investigation that helps the management to take decision about whether study of the proposed system should be feasible for development or not. A feasibility study is basically an analysis that considers all relevant factorsof a project including economic, technical, legal, and behavioural considerations in order to ensure the likelihood of completing the project successfully.

Whether a project is feasible or not can depend on several factors, including the project's cost and return on investment which means whether the project will generate enough revenue or sales from consumers.For example, a feasibility study could help to determine whether a hospital can generate enough donations and invest money to expand and build a new cancer centre.

Although feasibility studies can help project managers to determine the risk and return of pursuing a plan of action, several steps and best practices should be considered before moving forward.

A feasibility study is a test of a system proposal according to its workability, impact on the organization, ability to meet user needs, and effective use of resources. It focuses on three major questions:

1. What are the user's demonstrable needs and how does a candidate system meet them?

2. What resources are available for given candidate systems? Is the problem worth solving?
3. What is the likely impact of the candidate system on the organization?

Each of these questions must be answered carefully. They revolve around investigation and evaluation of the problem, identification and description of candidate systems, specification or performance and the cost of each system and final selection of the best system.

The objective of feasibility study is not to solve the problem but to acquire a sense of its scope. During the study, the problem definition is preserved and aspects of the problem to be included in the system are determined. Consequently, costs and benefits are estimated with greater accuracy at this stage. The result of the feasibility study is a formal proposal. This is simply a report- a formal document detailing the nature and scope of the proposed solution. The proposal summarizes what is known and what is going to be done. It consists of the following:

1. **Statement of the problem** - a carefully worded statement of the problem that led to analysis.
2. **Summary of findings and recommendations**- a list of the major findings and recommendations of the study. It is ideal for the user who requires quick access to the results of the analysis of the system under study. Conclusions are stated followed by a list of the recommendations and a justification for them.
3. **Details of findings**- an outline of the methods and procedures undertaken by the existing system followed by coverage of the objectives and procedures of the candidate system. It also includes the discussions of output reports, file structures, and costs and benefits of the candidate system.
4. **Recommendations and conclusions**- specific recommendations regarding the candidate system including personnel assignments, costs, project schedules, and target dates.

Benefits of a Feasibility Study:

There are several benefits to feasibility studies which includes:

1. It helps project managers to determine the pros and cons of undertaking a project before investing a significant amount of time and capital into it.
2. It provides a company's management team with crucial information that could prevent them from entering into a risky business venture.
3. It also helps companies with new business development, including determining how it will operate, potential obstacles, competition, market analysis, and the amount and source of financing needed to grow the business.

Best Practices for Conducting a Feasibility Study:

Although each project can have unique goals and needs, below are some best practices for conducting a feasibility study:

- Conducting a preliminary analysis, which involves getting feedback about the new concept from the appropriate stakeholders.
- Analysing and asking questions about the data obtained in the early phase of the study to make sure that data are perfect.
- Conducting a market survey or market research to identify the market demand and opportunity for pursuing the project or business.
- Preparing an organizational, operational, or business plan, including identifying the amount of labour needed, at what cost, and for how long
- Making a projected income statement, which includes revenue, operating costs, and profit.
- Preparing an opening day balance sheet.
- Identifying obstacles and any potential vulnerabilities, as well as how to deal with them.

A feasibility analysis evaluates the project's potential for success and therefore, apparent objectivity is an essential factor in the credibility of the study for potential investors and lending institutions. There are three key classes of feasibility studies each of which are separately examined in the following:

7.7.1 Economic Feasibility

This assessment typically focuses on cost/ benefits analysis of the project, helping organizations to determine the viability, cost, and benefits associated with a project before financial resources are allocated. It also serves as an independent project assessment and enhances project credibility by helping decision-makers to fix the positive economic benefits to the organization that the proposed project will provide. If benefits outweigh costs, then the decision is finalised to design and implement the system. Otherwise, further review is to be made in the proposed system.

The financial and economic questions raised by analysts during the preliminary investigation include the following: a) The cost to conduct a full systems investigation.b) The cost of hardware and software for the class of application being considered.

7.7.2 Technical Feasibility

This assessment focuses on the technical resources (hardware, software, etc.) available to the organization and to what extent it can support the proposed system. It aids organizations to determine whether the technical resources meet capacity and whether the technical team is capable of converting these new ideas into working systems. The availability of the hardware, software, and other technical requirements like power supply of the proposed system are considered in technical feasibility study.The technical challenges usually involved during the feasibility stage of the investigation which includes the following:

1. Does the necessary technology exist to do what is suggested?
2. Does the proposed equipment have the technical capacity to hold the data required to use the new system?
3. Will the proposed system provide adequate responses to inquiries, regardless of the number or location of users?
4. Can the system be expanded if developed?

5. Are there technical guarantees of accuracy, reliability, ease of access, and data security?

For example, if the proposal includes a printer that prints at the rate of 15,000 lines per minute, a brief search shows that this specification is technically feasible. On the other hand, if a user is requesting voice input to write, read, and change stored data, the proposal may not be technically feasible.On the other hand, if new proposal requires additional hardware, financial liabilities are also to be considered.

7.7.3 Behavioural Feasibility

It evaluates and estimates the user attitude or behaviour towards the development of new system.It is known to all that computer installation in an organization can turnover and change in employee job status. It helps in determining whether the system needs special effort to educate, retrain, transfer, and changes in employee's job status on new ways of conducting business. An estimate should be made of how strong a user staff shows reaction toward the development of a computerised system.

In addition to these three key feasibility considerations, some other feasibility studies can also be reviewed which are explained here.

Legal Feasibility

This assessment investigates whether any aspect of the proposed project conflicts with legal requirements like zoning laws, data protection acts or social media laws. Let's consider an organization wants to construct a new office building in a specific location. A feasibility study might reveal that the organization's ideal location isn't zoned for that type of business. That organization has just saved considerable time and effort by learning that their project was not feasible right from the beginning.

Operational Feasibility

This feasibility study involves undertaking a study to analyse and determine whether and how wellthe organization's needs can be

fulfilled by completing the project. It also examines how a project plan satisfies the requirements identified in the requirements analysis phase of system development.

Scheduling Feasibility

This assessment is the most important for project success.A project may fail if not completed on time. In scheduling feasibility, an organization estimates how much time the project will take to complete.When these areas have all been examined, the feasibility analysis helps to identify any constraints the proposed project may face.



CHECK YOUR PROGRESS

Q.1: Choose the correct options from the followings:

a) Which of the following is among the aspects of technical feasibility study?

i. Demand ii. Location iii. Security

iv. Instruction

b) A feasibility study

i. includes a statement of the problems

ii. considers a single solutions

iii. both (a) and (b)

iv. a list of alternative solution considered

c) The main objective of feasibility study is

i. to assess whether it is possible to meet the requirements specifications

ii. to assess if it is possible to meet the requirements specified subject to constraints of budget, human resource and hardware

iii. to assist the management in implementing the desired system

iv. to remove bottlenecks in implementing the desired system

Importance of Feasibility Study:

The importance of a feasibility study is based on organizational desire to "get it right" before committing resources, time, or budget. A feasibility study might uncover new ideas that could completely change a project's scope. It's best to make these determinations in advance, rather than to jump in and to learn that the project won't work. Conducting a feasibility study is always beneficial to the project as it gives you and other stakeholders a clear picture of the proposed project.

Below are some key benefits of conducting a feasibility study:

- Improves project teams' focus
- Identifies new opportunities
- Provides valuable information for a "go/no-go" decision
- Narrows the business alternatives
- Identifies a valid reason to undertake the project
- Enhances the success rate by evaluating multiple parameters
- Aids decision-making on the project
- Identifies reasons not to proceed

7.8 STEPS IN FEASIBILITY ANALYSIS

Feasibility analysis is accomplished through the following eight steps:

1. Creating a project team and appointing a project leader.
2. Designing system flowcharts.
3. Enumerate potential candidate systems.
4. Describing and identifying the characteristics of candidate system.
5. Determine and evaluate performance and cost effectiveness of each candidate system.
6. Weight the performance and cost effectiveness of each candidate system.
7. Rank the other alternatives and select the best candidate system.
8. Prepare a system proposal of final project directive to management for approval.

Creating a project team and appointing a project leader

The first step in feasibility analysis involves forming a project team. The teamconsists of analysts and user staff. In some cases, an outside consultant andan information specialist are required to join the team until the job iscompleted.Projects are planned to consume a time period ranging fromseveral weeks to months. The senior system analyst is generally appointed asproject leader. This assignment is temporary and lasting till the completion ofthe project. Regular meetings take place to keep up the progress andaccomplish the mission by selecting the best candidate system. A record iskept of the progress made in each meeting.

Designing system flowcharts

The next step in feasibility study is to prepare a system flowchart for the system. Information-oriented charts and data flow diagrams prepared in the initial investigation are reviewed at this time. The system flowchart introduces the importance of inputs, outputs and data flow among key points in the existing system.

Enumerating potential candidate systems

This step identifies the candidate systems which are capable of producing the outputs included in the flowcharts. This requires a transition from logical to physical system models. Another motive of this phase is the consideration of the hardware that can handle the total system requirements such as CPU processing speed and main memory capabilities.

Describing and identifyingthe characteristics of candidate system

In this step, the analysis is mainly based on what each candidate system can and cannot do. For determining the characteristics of candidate system, technical knowledge and expertise in the hardware/software area arechallenging work for determining what each candidate system can and cannot do. The information along with additional data through the vendor focus on the positive and negative features of each system. The constraints unique to each system are also to be mentioned.

Determine and evaluate performance and cost effectiveness of each candidate system:

Here the analyst has to determine and evaluate the system's performance and cost of each candidate system against the system performance requirements set prior to the feasibility study.

The cost incorporates both designing and installing the system. Evaluation for both design and implementation are performed in this phase. It includes user training, updating the physical facilities and documenting etc. Costs can be easily computed if the benefits of the system are found to be tangible and measurable.

Weight system performance and cost data:

According to the performance and cost of the candidate system, some weight is given to each alternative of the system. Then the candidate system with the highest total score is considered acceptable.

The procedure for weighting candidate system is as follows:

1. Assign a weighting factor to each evaluation criterion based on the criterion's effect on the success of the system.
2. Fix a quantitative rating to each criterion's qualitative rating.For example, ratings for poor, fair, good, very good, excellent can be assigned 1,2,3,4,5 respectively.
3. Multiply the weight assigned to each category by the relative rating to determine the score.
4. Find the summation of the score column for each candidate system.

Rank the other alternatives and select the best candidate system:

The system with the highest total score is considered the best system among several candidate systems. This simplifies that the weighting factors are impartial and the rating of each evaluation criterion is accurate and acceptable.

7.9 FEASIBILITY REPORT

The outcome of the feasibility study is a document called feasibility report which is prepared and directed to the management. This report is a formal document for management use. It should be brief, and sufficiently

nontechnical to be understandable. However, it provides the detailed basis for system design. Although there is no standard format for feasibility report, yet a suitable format is considered by the analyst which is acceptable for the user and system.

The following sections are included in a feasibility report:

1. **Cover letter**-It presents the report and briefly specifies the nature, general findings and recommendations to be considered to management.
2. **Table of contents**- It lays down the location of the various parts of the reports.
3. **Overview**- Adescriptive explanation of the purpose and scope of the project, the reason for undertaking the feasibility study and the department(s) involved or affected by the candidate system. It also includes the names of the persons who conducted the study and when it started.
4. **Detailed findings**- Point out the methods used in the present system. The System's effectiveness and efficiency as well as operating costs are considered. This section also provides a description of the objectives and general procedures of the candidate system.
5. **Economic justification**–It presents point by point cost comparisons and preliminary cost estimates for the development and operation of the candidate system. A return on investment assessment of the project is also included.
6. **Recommendations and conclusions**– It proposes the most beneficial and cost-effective system to management. They are written only as a recommendation, not as a command; from which any conclusions from the study may be encompassed.
7. **Appendixes** –It documents all memos and data compiled during the investigation. They are placed at the end of the report for reference.

Thereby, it can be summarised that the feasibility report is a detailed summary of the investigation that is carried out and it outlines the options

and recommendations. After carefully going through this report, management can decide whether the candidate system should be designed. An outline of a written feasibility report is presented in the Table 1.

Table 1. Feasibility report

SL.No	Section	Purpose
1	TITLE PAGE	Defines the name of the project
2	TABLE OF CONTENTS	Includes various parts, features and exhibits showing page numbers
3	SCOPE	Defines a short description of the system boundaries
4	STATEMENT OF THE PROBLEM	Describe current system followed by proposed system
5	SUMMARY/ABSTRACT	Mention a summary of project highlighting benefits
6	COST/BENEFIT STATEMENT	List quantitative benefits and savings
7	IMPLEMENTATION SCHEDULE	Describe implementation plan, human resources requirements
8	HARDWARE CONFIGURATION	List computer configuration, perpherals, communication equipment
9	CREDITS	Give credit to those who contributed to the project study
10	APPENDIX	Include exhibits, correspondence on project



CHECK YOUR PROGRESS

Q2: Choose the correct options from the followings:

a) The rate of ……. is always an important factor for management especially in relation to the rate of return

i. Cost
ii. Benefit
iii. Sales
iv. Investment

b) The Key considerations involved in the feasibility analysis is / are
i. Economic
ii. Technical
iii. Behavioural
iv. All of these

c) ……. gives defining the flow of the data through and organization or a company or series of tasks that may or may not represent computerized processing.
i. System process
ii. System flowchart
iii. System design
iv. Structured System3



7.10 LET US SUM UP

- A feasibility study is carried out to select the best system that meets performance requirements. It involves an identification description, an evaluation of candidate systems, and the selection of the best system.
- System's required performance is defined by a statement of constraints, the identification of specific system objectives, and a description of outputs.
- There are three basic considerations in feasibility analysis which are economic feasibility, technical feasibility and behavioural feasibility. Economic analysis is also called cost/benefit analysis which determines whether the adoption of a new system is cost effective. Technical feasibility centers around the existing hardware

and software system. Behavioural analysis determines how much effort is required for educating, selling, and training the user staff on a candidate system.



7.11 FURTHER READING

1. Awad, E.M. (2010). System analysis and Design. Galgotia Publications Pvt. Ltd.
2. Wasson, C.S. (2005). System analysis, design and development : concepts, principles and practices (Vol. 22). John Wiley & Sons.



7.12 ANSWERS TO CHECK YOUR PROGRESS

Ans to Q No 1: a) ii. b) i c) ii

Ans to Q No 2: a) iv b) iv c) ii



7.13 MODEL QUESTIONS

Q 1: What makes up a system performance definition? With an example, explain the steps to prepare the definition.

Q 2: Evaluate the feasibility study. What were the strong and weak points of the study?

Q 3: Discuss the three key considerations on feasibility study. For a proposed system “Payroll management system”, how do you describe these threeassessments.

Q 4: What makes up a feasibility report? How do you change it? Explain it.

Q 5: Mention the steps involved in feasibility analysis and explain each step briefly.

Q 6: Discuss the elements considered for preparing a feasibility report.

*** ***** ***