Table of Contents

What is an Information System?

Information System can be defined technically as a set of interrelated components that

- Collects or Retrieve
- Process
- Store
- Distribute Information

To support decision making, coordination and control of an organization

An information system can be mainly divided into two parts.

- Information systems based on computer technology.
- Information systems that are not based on computer technology.

A system is an orderly grouping of interrelated and interdependent components linked together according to a plan to achieve a specific objective.

An open system has many interfaces with its environment. It permits interaction across system boundary. The system receives inputs from and delivers outputs to its environment.

A closed system is isolated from environmental influences.

Ex. Human Respiratory System - Open system

Human Blood Circulation System - Closed System

The systems which are made by man are called manmade systems whereas nature made systems are called natural systems.

Ex. Transportation System of a Country - Manmade System

Human Blood Circulation System - Natural System

Living systems are open self-organizing living things that interact with their environment. The systems consist of non-living things are categorized into nonliving systems.

Ex. Solar System - Nonliving System

Human Respiratory System - Living System

Organizations and Information Systems

Information systems and organizations influence one another. Information systems must be aligned with the organization to provide information that important groups within the organization need. At the same time, the organization must be aware of and be open to the influences of information systems in order to benefit from new technologies.

What is an Organization?

An organization is a stable, formal, social structure that takes resources from the environment and processes them to produce outputs.

Organizational Levels

Top Management: is responsible for long range, strategic time frame measured in one or more years. Establishes overall company policies and goals, determines when organizational restructuring must occur and sets new directions for the company in terms of producers' services and acquisitions.

Middle Management: Focuses on the short range, tactical time frame, usually in the range of one month to one year.

Lower Management: Supervises operational personnel and establishes the day-to-day plans, based on direction and longer range plans from higher levels of management.

Operational personnel: Perform a company's repetitive day-to-day functions following well defined procedures.

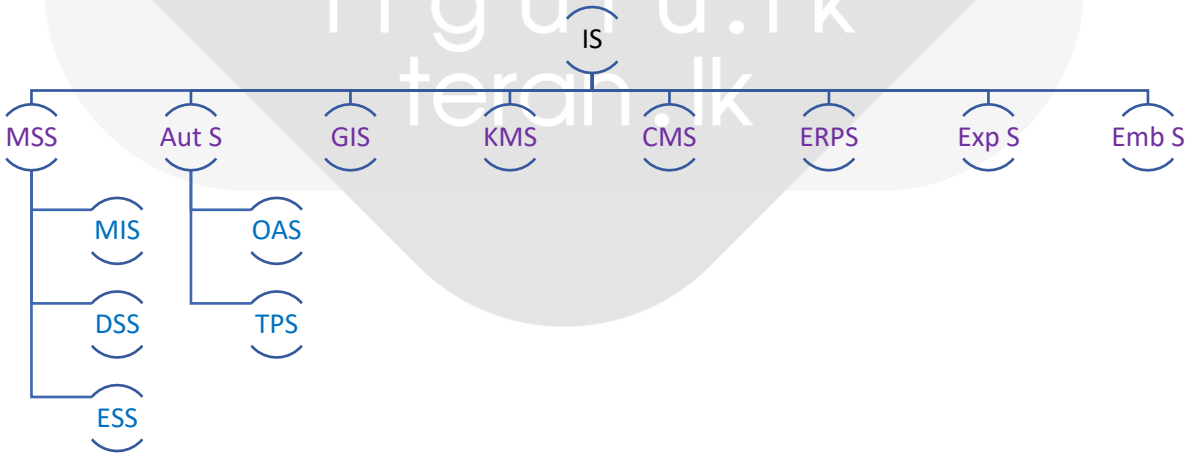
The Role of Managers in the Organizations

Managers play a key role in organizations. Their responsibilities range from making decisions, to writing reports, to attending meetings, to arranging birthday parties, etc...

Managers and Decision Making

Decision making is often a manager's most challenging role. Information systems have helped managers communicate and distribute information; however, they have provided only limited assistance for management decision making. Further decision-support system (DSS) assists management decision-making by combining data, sophisticated analytical models and tools, and user-friendly software into a single powerful system that can support semi structured or unstructured decision-making.

Types of Information Systems



AUTOMATED SYSTEMS (AS)

These types of systems are used to automate day to day operations of an organization.

Office Automation System (OAS)

Office Automation Systems (OAS) are computer systems such as word processing, electronic mail and work scheduling systems that are designed to increase the productivity of employees in an office.

Transaction Processing Systems (TPS)

Transaction Processing Systems (TPS) are computerized systems that perform and record the daily routine transactions necessary to conduct a business and serve the operational level users of an organization.

There are two ways to process transactions: using batches and in real time. In a batch processing system, transactions are accumulated over a period of time and processed as a single unit, or batch. In a real-time processing system, transactions are processed immediately as they occur without any delay to accumulate transactions. Real-time processing is also referred to as online transaction processing, or OLTP.

batch- Payroll

real time- ATM

Geographic Information Systems (GIS)

Geographic Information Systems (GIS) are computer systems that allow to map, model, query and analyze large quantities of data within a single database according to their location. GIS facilitates to create maps, integrate information, visualize scenarios, present powerful ideas and develop effective solutions.

Knowledge Management System (KMS)

Knowledge Management Systems (KMS) comprise a range of practices used in an organization to identify, create, represent, distribute and enable adoption of insight and experiences. Such insights and experiences comprise knowledge, either embodied in individual or embedded in organizational processes and practices.

A knowledge management system is made up of different software modules served by a central user interface. Some of these features can allow for data mining on customer input and histories, along with the provision or sharing of electronic documents. Knowledge management systems can help with staff training and orientation, support better sales, or help business leaders to make critical decisions.

Content Management Systems (CMS)

Content Management Systems (CMS) are computer applications that support the creation and modification of digital content. It supports to multiple users working in a collaborative environment. Examples for CMS are web-based publishing, format management, history editing and version control, indexing, search and retrieval etc. CMS supports the separation of content and presentation.

The goal of a CMS is to provide an intuitive user interface for building and modifying webpage content. Each CMS also provides a web publishing tool that allows one or more users to publish updates live on the Web.

Eg : wordpress, joomla,Drupla

ENTERPRISE RESOURCE PLANNING SYSTEM (ERPS)

Enterprise Resource Planning (ERP) Systems are business process management systems that allow organizations to use integrated applications to manage the businesses. An ERP system automates many back-office functions related to technology, services and human resources. ERP software integrates product planning, development, manufacturing, sales and marketing.

An ERP system generally includes all the IT systems that enable ERP software to execute and operate efficiently. An ERP system allows an organization to design, execute, operate and maintain an enterprise information system.

EXPERT SYSTEM (ES)

Expert Systems are computer applications that use artificial intelligence. To design an expert system, one needs a knowledge engineer, an individual who studies how human experts make decisions and translates the rules into terms that a computer can understand.

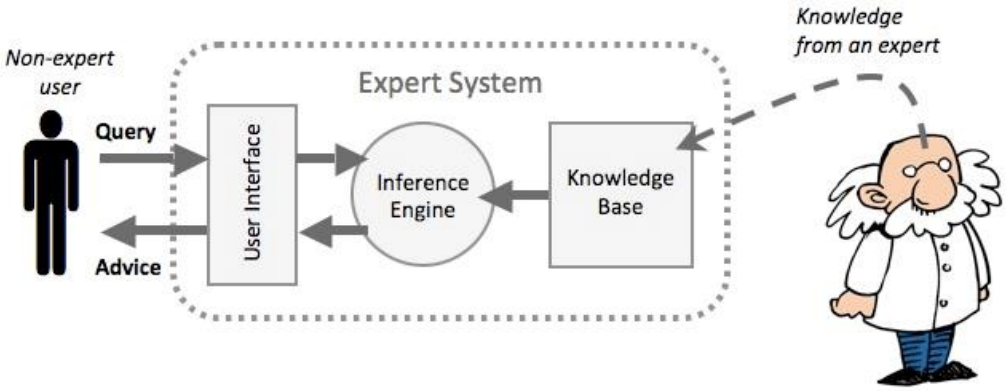
Artificial intelligence based system that converts the knowledge of an expert in a specific subject into a software code. This code can be merged with other such codes (based on the knowledge of other experts) and used for answering questions (queries) submitted through a computer. Expert systems typically consist of three parts:

(1) a knowledge base which contains the information acquired by interviewing experts, and logic rules that govern how that information is applied;

(2) an Inference engine that interprets the submitted problem against the rules and logic of information stored in the knowledge base;

(3) Interface that allows the user to express the problem in a human language such as English.

Expert systems now have commercial applications in fields as diverse as medical diagnosis, petroleum engineering, and financial investing.



EMBEDDED SYSTEMS (ES)

An embedded system is a computer system with a dedicated function within a larger mechanical or electrical system, often with real-time computing constraints. It is embedded as part of a complete device often including hardware and mechanical parts. Embedded systems control many devices in common use today.

MANAGEMENT SUPPORT SYSTEM (MSS)

Supports managers of a company in terms of future planning and decision making.

Management Information System (MIS)

The term management information systems (MIS) also designates a specific category of information systems serving management-level functions. Management information systems (MIS) serve the management level of the organization, providing managers with reports and often online access to the organization's current performance and historical records. Typically, MIS are oriented almost exclusively to internal, not environmental or external, events. MIS primarily serve the functions of planning, controlling, and decision making at the management level. Generally, they depend on underlying transaction processing systems for their data. MIS summarize and report on the company's basic operations. The basic transaction data from TPS are compressed and are usually presented in long reports that are produced on a regular schedule.

Decision Support System (DSS)

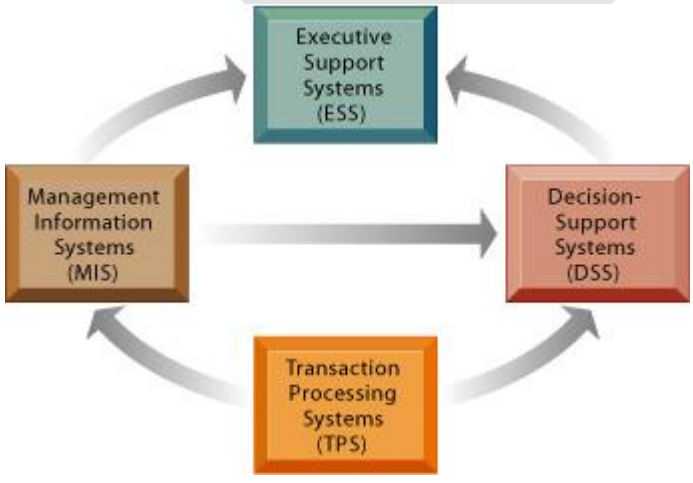
Decision-support systems (DSS) also serve the management level of the organization. DSS help managers make decisions that are unique, rapidly changing, and not easily specified in advance. They address problems where the procedure for arriving at a solution may not be fully predefined in advance. Although DSS use internal information from TPS and MIS, they often bring in information from external sources, such as current stock prices or product prices of competitors.

Executive Support System (ESS)

Senior managers use executive support systems (ESS) to help them make decisions. ESS serve the strategic level of the organization. They address nonroutine decisions requiring judgment, evaluation, and insight because there is no agreed-on procedure for arriving at a solution. ESS are designed to incorporate data about external events, such as new tax laws or competitors, but they also draw summarized information from internal MIS and DSS. They filter, compress, and track critical data, displaying the data of greatest importance to senior managers. ESS is also known as Executive Information System (EIS).

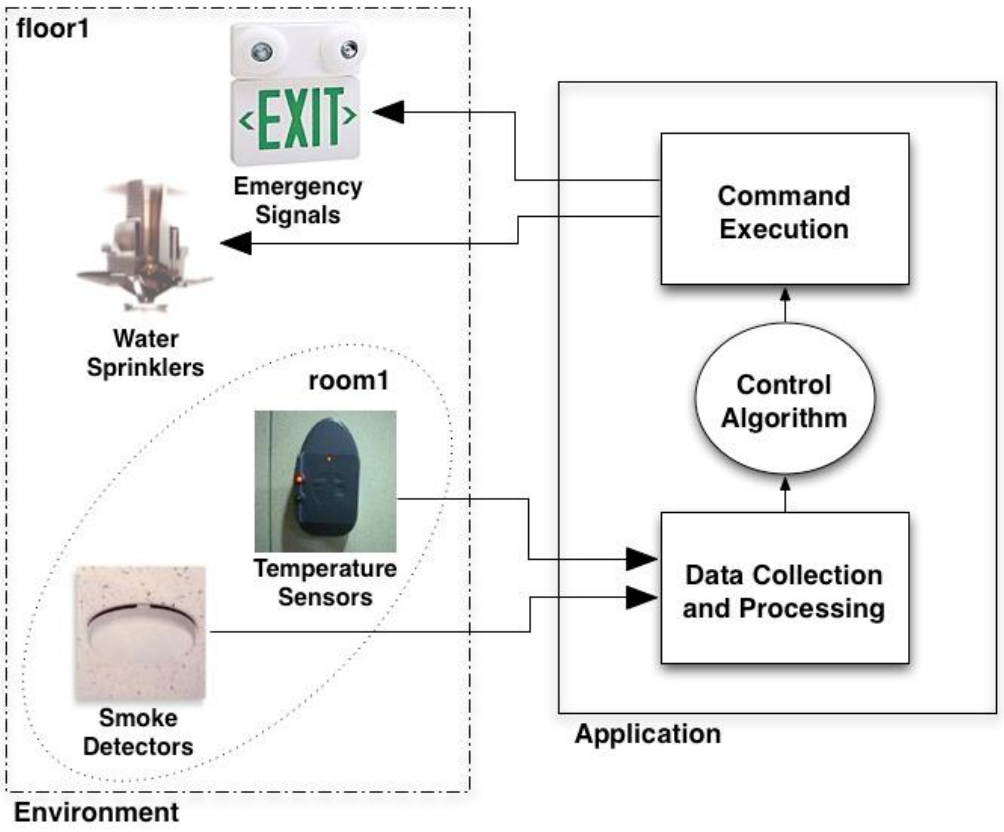
Relationship between TPS, MIS and DSS and EIS

Illustrates how the systems serving different levels in the organization are related to one another. TPS are typically a major source of data for other systems, whereas ESS are primarily a recipient of data from lower-level systems. The other types of systems may exchange data with each other as well. Data may also be exchanged among systems serving different functional areas. For example, an order captured by a sales system may be transmitted to a manufacturing system as a transaction for producing or delivering the product specified in the order or to a MIS for financial reporting.



SMART SYSTEMS

Smart Systems make decisions based on the available data in a predictive or adaptive manner, by means of sensing, actuating and controlling. Smart actions incorporate the ways to describe and analyze a situation. Smart systems can be attributed to autonomous operation based on closed loop control, energy efficiency, and networking capabilities.



System Development Life Cycle Models

WATERFALL MODEL

The Waterfall Model was the first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

The waterfall Model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete. In this waterfall model, the phases do not overlap.

Phases of the Model

1. Problem Definition
2. Feasibility Study
3. System Analysis
4. System Design
5. Development
6. Testing
7. Implementation
8. Maintenance

1. Problem Definition

Identification of problems that the system will have to address when providing system as a solution

2. Feasibility Study

Different Areas of Feasibility: Technical, Operational, Economic, Social, Legal, Organizational.

Technical feasibility

This evaluates whether the developers have ability to construct the proposed system. The technical assessment helps answer the question such as whether the technology needed for the system exists, how difficult system will be to develop, and whether the developers have enough experience using that technology.

Operational feasibility:

This assesses the willingness and ability of the users to support and use the proposed system. Will the system be used when it is developed and installed? Will there be resistance from users to the system development?

Economic feasibility

This study cost and benefits to evaluate whether the benefits justify the investments in the system development. Can the development cost be justified? An important outcome of the economic feasibility study is the cost benefit analysis.

Social Feasibility - .....

.....

.....

.....

Legal Feasibility - .....

.....

.....

....

Organization feasibility

This determines the extent to which the proposed system supports the objectives of the organization's strategy. In here, the system is taken as a subset of the whole organization.

The result of the feasibility study is a formal document, a report detailing the nature and scope of the proposed solution. Once the feasibility study is done, the project is approved or disapproved according to the results of the study. If the project seems feasible and desirable, then the project is finally approved otherwise no further work is carried out.

3. System Analysis

Requirement analysis is the process of studying and analyzing the user needs to arrive at a definition of the problem domain and system requirements. The main objective of requirement analysis is to discover the boundaries of the new system and how system must interact within the new problem domain. Requirement analysis helps to detect and resolve conflicts between (user) requirements.

Functional requirements: What activities that the system should carry out

Non-Functional requirements: How well or with in what limits requirement should be satisfied

Essential requirements are defined with "**Shall**" whereas nice to have requirements are defined with "**Should**"

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Outcome of this phase is System Requirement Specification which is a detailed document consist of all the functional and non-functional requirements.

Both functional and non-functional can be compulsory and optional.

shall should

.....

SRS

.....

functional

.....

.....

.....

.....

.....

4. System Design

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

5. Development

.....

.....

.....

.....

.....

.....

.....

6. Testing

.....

.....

Test Cases

A test case is a set of actions executed to verify a particular feature or functionality of a software application. Test cases are documented by the Quality Assurance team while the software development is going on.

Software testing techniques

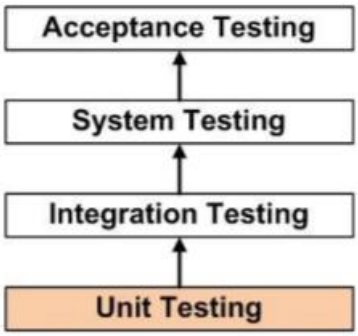
White Box Testing

- takes internal implementation of a software to derive test cases to test the software
- performed in the early stages of the testing process
- usually applied for testing relatively small program units
- analysis of the program code determines how many test cases are required to exercise all the internal components of the software (statements, branches, paths) adequately

Black box Testing

- software to be tested is treated as a black box and its behavior is examined by studying the inputs and outputs
- knowing the intended functionalities of a software, tests are conducted to see whether the software can deliver them
- test cases are derived from the requirement specification of the software to be tested

Software Testing Types



In software development,

- software systems are built with sub-systems
- sub-systems are built with individual program units such as functions or classes.

Levels of Testing –

- starts with testing of these individual program units (Unit Testing, usually carried out by programmers, white box techniques are used)
- continues with the testing of the integration of these units (Integration Testing, usually carried out by integration testers or test teams, can either be white box or black box)
- and the testing of the system's functionality as a whole (System Testing, usually carried out by test teams independent of the programmers who developed the system, black box techniques are used)
- finally ends with testing to see whether the system is acceptable to the users (Acceptance Testing, usually carried out by test teams independent of the programmers and/or users who developed the system, black box techniques are used)

7. Implementation (Software Deployment)

Software deployment includes all the activities that make a software system available for use. The general deployment process consists of several interrelated activities with possible transitions between them. These activities can occur at the developer's side or at the user's side or both.

.....

.....

.....

.....

.....

Implementation is also known as deployment and there are several approaches: Direct, Parallel, Phased, Pilot.

Direct - .....

.....

.....

.....

.....

Advantages - .....

.....

Disadvantages - .....

.....

Parallel - .....

.....

.....

.....

.....

Advantages - .....

.....

Disadvantages - .....

.....

Phased - .....

.....

.....

.....

.....

Advantages - .....

.....

Disadvantages - .....

.....

Pilot - .....

.....

.....

.....

.....

Advantages - .....

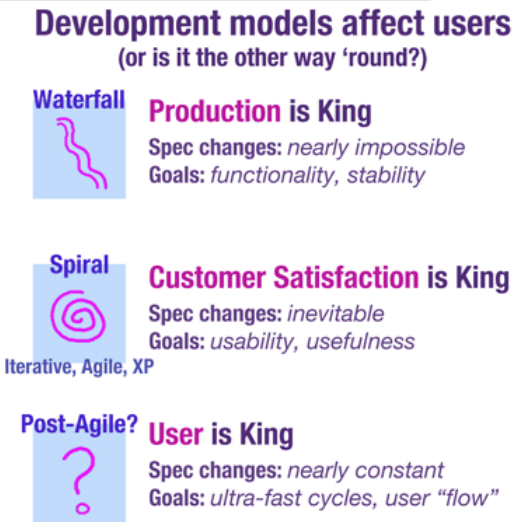
.....

Disadvantages - .....

.....

8. Maintenance

The maintenance phase of the SDLC occurs after the software is in full operation. The maintenance phase involves making changes to hardware, software, and documentation to support its operational effectiveness. It includes making changes to improve a system's performance, correct errors, deal with security issues, or address new user requirements. To ensure that the modifications do not disrupt operations or degrade a system's performance or security, organizations use change management standards and procedures.



SPIRAL MODEL

What is Spiral Model?

The spiral model is similar to the incremental model, with more emphasis placed on risk analysis. The spiral model has four phases: Planning, Risk Analysis, Engineering and Evaluation. A software project repeatedly passes through these phases in iterations (called Spirals in this model). The baseline spiral, starting in the planning phase, requirements are gathered and risk is assessed. Each subsequent spirals builds on the baseline spiral.

This Spiral model is a combination of iterative development process and sequential linear development process. Spiral model is suitable for systems where project risk is medium to high, requirements are complex, need an evaluation to get cleared them and significant changes are expected during the development.

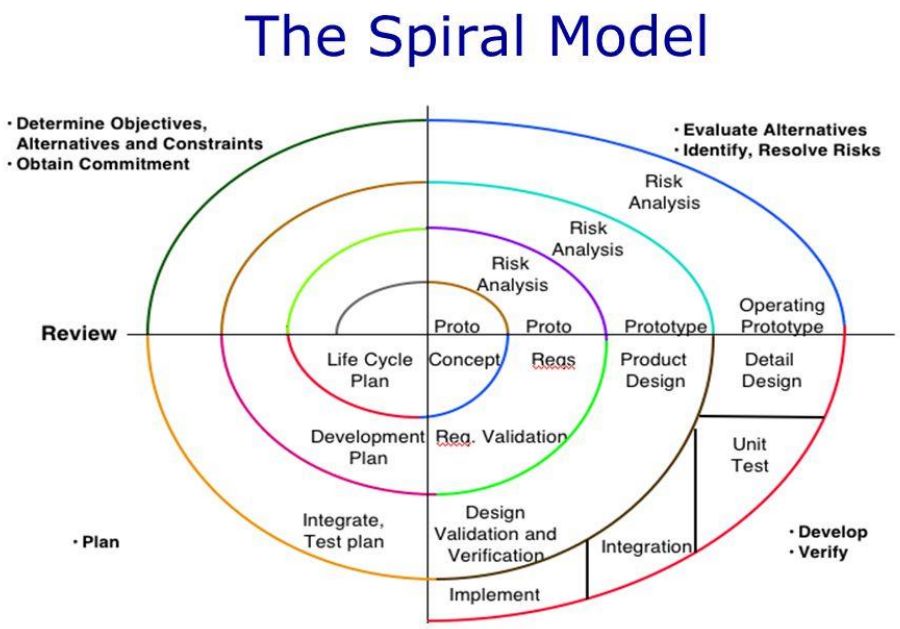
Phases of Spiral Model

1. Determining Objectives (Planning Phase): Requirements are gathered during the planning phase. Requirements like 'BRS' that is 'Business Requirement Specifications' and 'SRS' that is 'System Requirement specifications.

2. Identify and Resolve Risks (Risk Analysis Phase): In the risk analysis phase, a process is undertaken to identify risk and alternate solutions. A prototype is produced at the end of the risk analysis phase. If any risk is found during the risk analysis then alternate solutions are suggested and implemented.

3. Development and Test (Engineering Phase): In this phase software is developed, along with testing at the end of the phase. Hence in this phase the development and testing is done.

4. Plan the Next Iteration (Evaluation phase): This phase allows the customer to evaluate the output of the project to date before the project continues to the next spiral.



Advantages of Spiral model:

- High amount of risk analysis hence, avoidance of Risk is enhanced.
- Good for large and mission-critical projects.
- Strong approval and documentation control.
- Additional Functionality can be added at a later date.
- Software is produced early in the software life cycle.

Disadvantages of Spiral model:

- Can be a costly model to use.
- Risk analysis requires highly specific expertise.
- Project's success is highly dependent on the risk analysis phase.
- Doesn't work well for smaller projects.

When to use Spiral model:

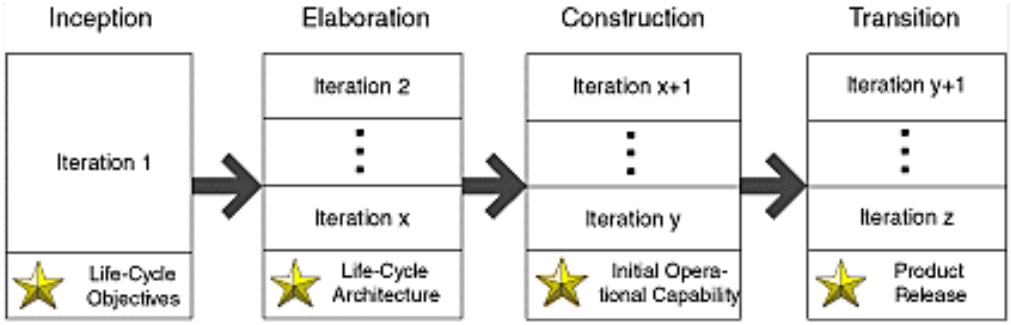
- When costs and risk evaluation is important.
- For medium to high-risk projects.
- Long-term project commitment unwise because of potential changes to economic priorities.
- Users are unsure of their needs.
- Requirements are complex.
- New product line.
- Significant changes are expected (research and exploration).

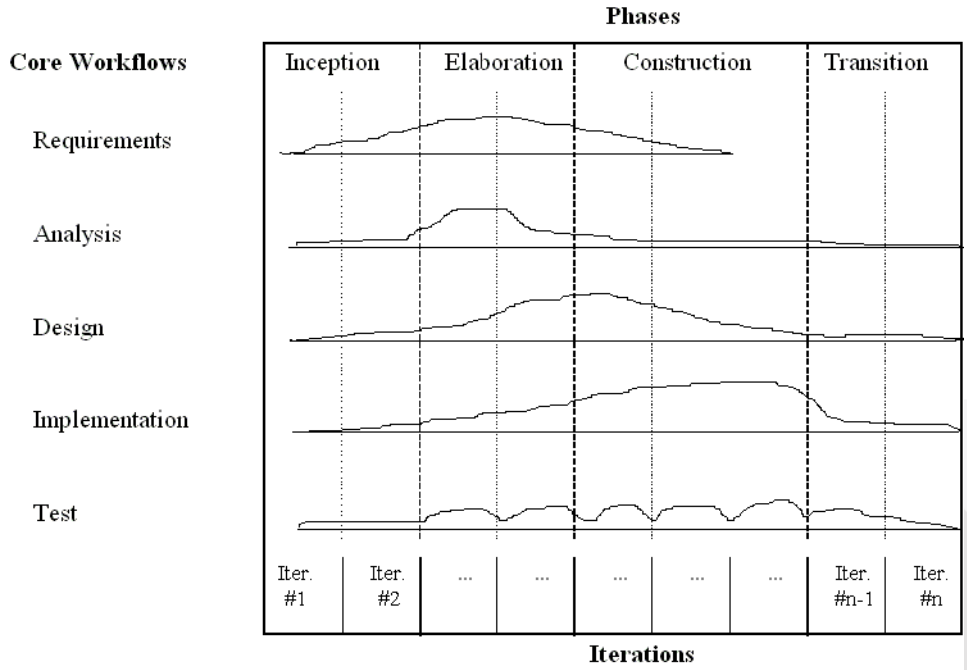
UNIFIED PROCESS MODEL

What is Unified Process Model?

The life of a software system can be represented as a series of cycles. A cycle ends with the release of a version of the system to customers.

Within the Unified Process, each cycle contains four phases. A phase is simply the span of time between two major milestones, points at which managers make important decisions about whether to proceed with development and, if so, what's required concerning project scope, budget, and schedule.





You can see that each phase contains one or more iterations.

Phases of Unified Process Model

Inception

The primary goal of the **Inception phase** is to establish the case for the viability of the proposed system.

- Outlining a **candidate architecture**, which is made up of initial versions of six different models
- Identifying critical risks and determining when and how the project will address them

Elaboration

The primary goal of the **Elaboration phase** is to establish the ability to build the new system given the financial constraints, schedule constraints, and other kinds of constraints that the development project faces.

- Expanding the candidate architecture into a full **architectural baseline**, which is an internal release of the system focused on describing the architecture

The major milestone associated with the Elaboration phase is called **Life-Cycle Architecture**. The indications that the project has reached this milestone include the following:

Construction

The primary goal of the **Construction phase** is to build a system capable of operating successfully in beta customer environments.

The major milestone associated with the Construction phase is called **Initial Operational Capability**. The project has reached this milestone if a set of beta customers has a more or less fully operational system in their hands.

Transition

The primary goal of the **Transition phase** is to roll out the fully functional system to customers.

The major milestone associated with the Transition phase is called **Product Release**.

Advantages of RUP

- Process details are expressed in general terms, allowing local customization
- Heavy emphasis on documentation (UML)
- Can embrace incremental releases
- Evolutionary approach can lead to clean implementations

Disadvantages of RUP

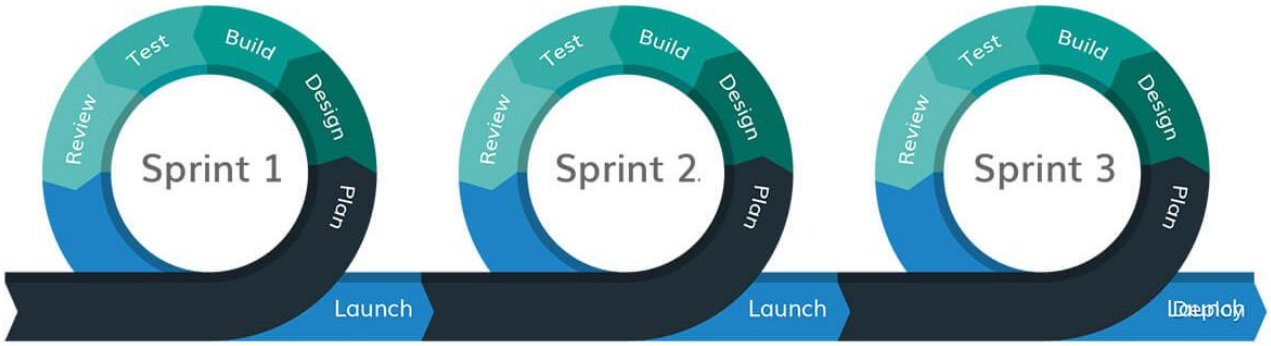
- Process details are expressed in general terms, providing minimal guidance and requiring local customization
- Complex
- Heavy documentation can be expensive

AGILE METHOD

Agile method Agile method uses the existing methods and modifies them to suit the project requirements. The tasks are divided to time slices to deliver specific features for a release.

This is an iterative approach and the software build is delivered after each iteration. Each build is an increment and the final product contains all the functionalities that the user requested.

The agile model is suitable for systems where requirements are fixed or changing.



PROTOTYPING

Prototype is used to displays the functionality of a product under development, but not the full functionality. It enables to understand the customer requirements in the early stages of software development as the prototype is demonstrated to the users to get their feedback. This helps developers to understand the exact user requirements.

RAPID APPLICATION DEVELOPMENT MODEL

What is RAD?

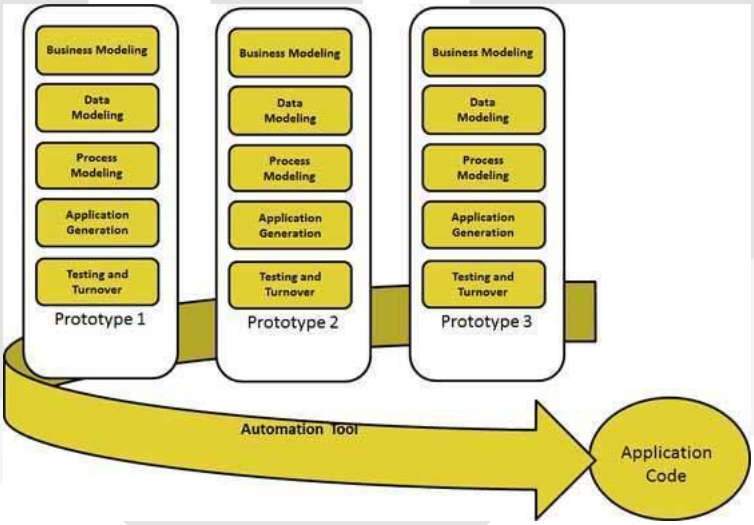
The RAD (Rapid Application Development) model is based on prototyping and iterative development with no specific planning involved. The process of writing the software itself involves the planning required for developing the product.

In this model, the functional modules are developed in parallel as prototypes and the final product is made by integrating them enabling a rapid delivery.

In RAD model the functional modules are developed in parallel as prototypes and are integrated to make the complete product for faster product delivery.

Since there is no detailed preplanning, it makes it easier to incorporate the changes within the development process. RAD projects follow iterative and incremental model and have small teams comprising of developers, domain experts, customer representatives and other IT resources working progressively on their component or prototype.

The most important aspect for this model to be successful is to make sure that the prototypes developed are reusable.



RAD MODEL DESIGN

RAD model distributes the analysis, design, build, and test phases into a series of short, iterative development cycles. Following are the phases of RAD Model:

Business Modelling: The business model for the product under development is designed in terms of flow of information and the distribution of information between various business channels. A complete business analysis is performed to find the vital information for business, how it can be obtained, how and when is the information processed and what are the factors driving successful flow of information.

Data Modelling: The information gathered in the Business Modelling phase is reviewed and analysed to form sets of data objects vital for the business. The attributes of all data sets is identified and defined. The relation between these data objects are established and defined in detail in relevance to the business model.

Process Modelling: The data object sets defined in the Data Modelling phase are converted to establish the business information flow needed to achieve specific business objectives as per the business model. The process model for any changes or enhancements to the data object sets is defined in this phase. Process descriptions for adding, deleting, retrieving or modifying a data object are given.

Application Generation: The actual system is built and coding is done by using automation tools to convert process and data models into actual prototypes.

Testing and Turnover: The overall testing time is reduced in RAD model as the prototypes are independently tested during every iteration. However the data flow and the interfaces between all the components need to be thoroughly tested with complete test coverage. Since most of the programming components have already been tested, it reduces the risk of any major issues.

<u>Advantages</u>

- Changing requirements can be accommodated.
- Progress can be measured.
- Iteration time can be short with use of powerful RAD tools.
- Productivity with fewer people in short time.
- Reduced development time.
- Increases reusability of components.
- Quick initial reviews occur.
- Encourages customer feedback.
- Integration from very beginning solves a lot of integration issues.

<u>Disadvantages</u>

- Dependency on technically strong team members for identifying business requirements.
- Only system that can be modularized can be built using RAD.
- Requires highly skilled developers/designers.
- High dependency on modelling skills.
- Inapplicable to cheaper projects as cost of modelling and automated code generation is very high.
- Management complexity is more.
- Suitable for systems that are component based and scalable.
- Requires user involvement throughout the life cycle.
- Suitable for project requiring shorter development times.

System Development Methodologies

STRUCTURED METHODOLOGY

Provides a framework (structure) with a set of well-defined guidelines through steps of tasks

OBJECT ORIENTED METHODOLOGY

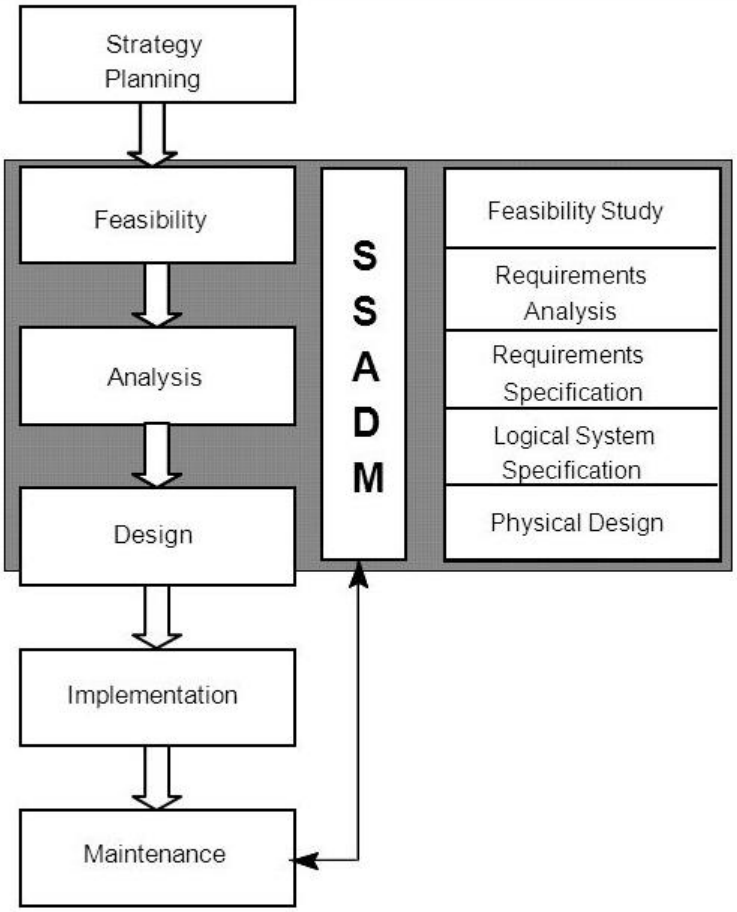
Models system as a collection of objects that work interactively.

Structured System Analysis and Design Methodology (SSADM)

Involves study the present system and sketches a blueprint to develop a new system or to modify the existing system

- Introduced in early 1980s
- One of the structured methods for System Analysis & Design
- Adopted as a standard by Central Computer and Telecommunications Agency(CCTA), UK
- Covers most of the System Development Life Cycle (SDLC) from Feasibility Study to System Design
- More focus on analysis and design

STAGES OF SDLC COVERED BY SSADM



Feasibility Study
The business area is analyzed to determine whether the system development is feasible.

Requirements Analysis
The requirements of the system to be developed are identified and the current business environment is modeled in terms of the processes carried out and the data stored

Requirements Specification
Detailed functional and non-functional requirements are defined and new processes are introduced to define the required processing and data storage

Logical System Specification
System to be developed as specified logically without taking technical constraints into consideration

Physical Design
Logical design is transformed into a physical design taking technical constraints into consideration.

Analyzing the Current System

The following analytical tools can be used to analyze the system.

- **Business Activity Modeling**
  - Business activity model
- **Data Flow Modeling (DFM)**
  - Data Flow Diagrams (DFD)
  - Elementary Process Descriptions (EPD)
- **Logical Data Modeling (LDM)**
  - Logical Data Structure (LDS)

Business Activity Modeling

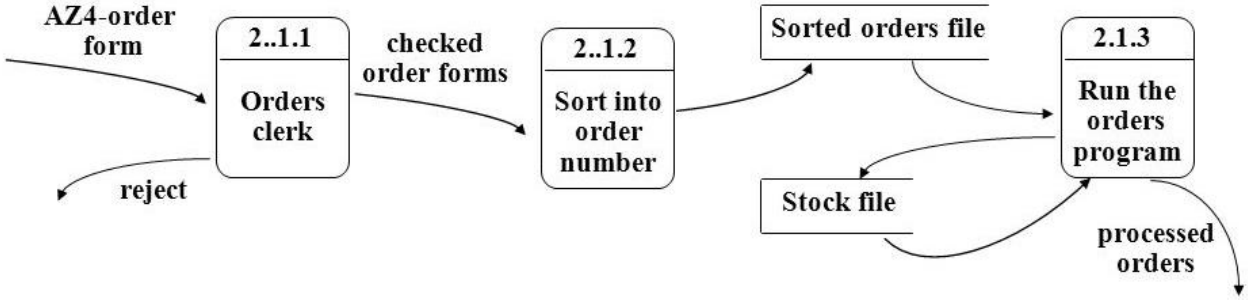
Business Activity Modeling A start-off technique for understanding what's going on in the system under investigation. It is used to show the business activities that the actors in the system's environment do and their associations.

DESIGNS THE PROPOSED SYSTEM

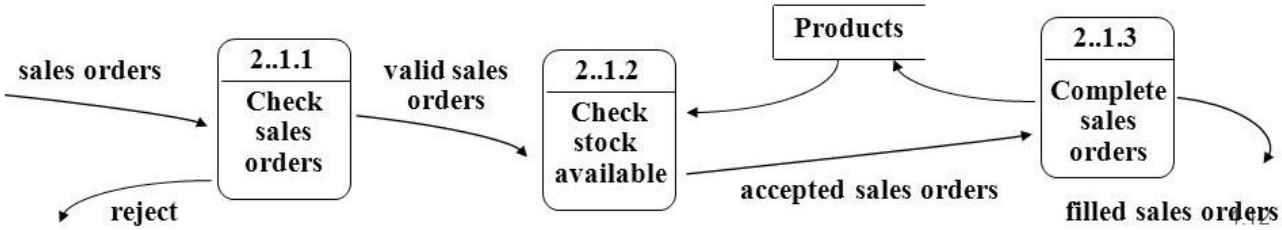
Logical Data Flow Modeling Vs Physical Data Flow Modeling

- **Physical DFM shows**
  - How data is actually processed and
  - Where data is actually stored in the current system
- **Logical DFM shows**
  - How data should be processed and
  - Where the data should be stored in the proposed system
- Logical DFM consist of a set of DFDs and associated textual descriptions
  - adds new entities required to support any new functionalities of the proposed system to the LDS
  - provides logical data stores (entities) identified in the LDM for logical DFM
  - removes any temporary data store that simply halts data temporarily

an example physical DFD for part of an order processing system



a logical DFD derived from the physical DFD above



Logical DFM of the proposed system

- starts from elementary processes
- as location of a process only indicates a physical constraint, removes it from all Processes
- as the person who actually did a process in physical DFM now feeds inputs to the

corresponding process in the logical DFM, transforms the location into an external entity
- removes any purely human activity and replaces it with an external entity
- adds new processes required to support any new functionalities of the proposed system
- reconstructs the hierarchy by regrouping logical processes based on their functionality
- describes the functionality of elementary processes in pseudo code.

PHYSICAL DESIGN OF A DATABASE

Maps logical schema to relational schema.

Logical Schema	Relational Schema
Entity	Table
Attribute	Field
Instance of an entity	Record of a table
Unique attribute	Primary key

- normalizes all the relations in relational schema to 3NF
- tabulates a Table Specification and a Record Specification for each relation in the normalized relational schema
- provides physical design for the construction of the database

DATA DICTIONARY

- is an integral part of database
- holds information about the database and the data that it stores (data about data -metadata)
- contains the actual database descriptions used by the Database Management System (DBMS)

System Implementation with COTS (ready-made)

Advantages of using Commercial Off-The-Shelf (COTS) Packages

COTS software systems tend to be far more configurable than custom developed software systems. In other words, they can accommodate change by adjusting parameters to a far greater degree than is usual for custom developed software systems. Moreover, package environments often come with capabilities that users do not to use immediately but are there for future exploitation as and when required.

A COTS software system can be implemented in significantly less time than a custom software system development and often at substantially less cost.

Disadvantages of using COTS Packages

COTS software systems can be highly complex and usually include many features that will be never used. Given the software tends to be large and complicated, it may take a long time to learn how to use the system properly. They may also require to alter the way that the work is done in order to fit in with the way that the system has been designed.

If the organization's business processes are unique, then the COTS software systems may need to be customized or the business processes may need to be changed to accommodate them.

As the competitors can also buy and use the same COTS software system, it is very difficult to gain any competitive advantage from using it.

Advantages of using custom developed software systems

Custom developed software systems are designed specifically to the requirements and built to operate exactly as needed. Moreover, they can be modified as the business changes or altered to further improve their performance. Custom developed software systems can also be integrated with the existing systems and any future systems as required.

The custom developed software systems can provide functions that are needed and none that that are not needed. They also operate in the way that users are used to work which means that they will be easier to use.

Being able to perform tasks that the competitors cannot do means that organizations can gain real competitive advantage with custom developed software solutions. Given that custom developed software can be listed as an asset of an organization, it can add value to the business.

Disadvantages of using custom developed software systems

Custom developed software systems require a large initial investment and the development process can take a long time

6. (a) A blood testing centre has the following activities:

The patient hands over the test request slip to the receiving counter. Receiving counter issues an invoice to the patient and sends a copy to the cashier. The patient checks the invoice, approves it and hands it over to the cashier with the payment. Cashier issues a receipt to the patient and also sends a copy of the receipt to the laboratory. Patient hands over the receipt to the laboratory. The laboratory verifies the patient and conducts the blood test and returns the updated receipt marked as 'done' to the patient. The laboratory sends the report to the receiving counter. Later, the patient hands over the updated receipt to the receiving counter and the receiving counter hands over the report to the patient with the re-updated receipt marked as 'issued'.

(i) The *context diagram* for the above activities, with missing *data flows* Ⓟ, Ⓠ, Ⓡ, Ⓢ and Ⓣ, is given in Figure 1 below.

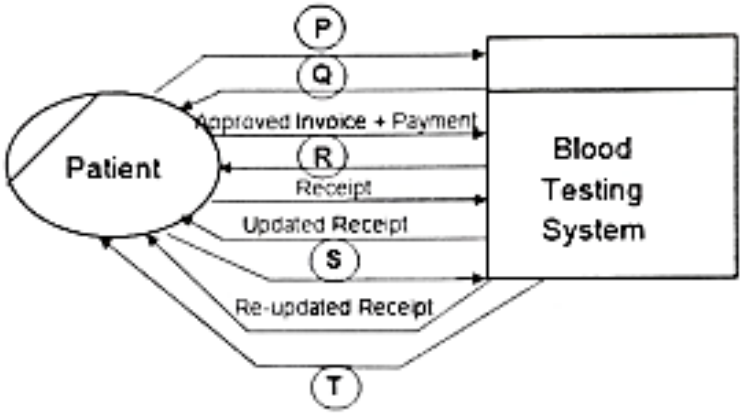


Figure 1

Identify the **five** missing *data flows* from the description given above and write them down.

(ii) Level 1 of the DFD for the above context diagram is shown in Figure 2.

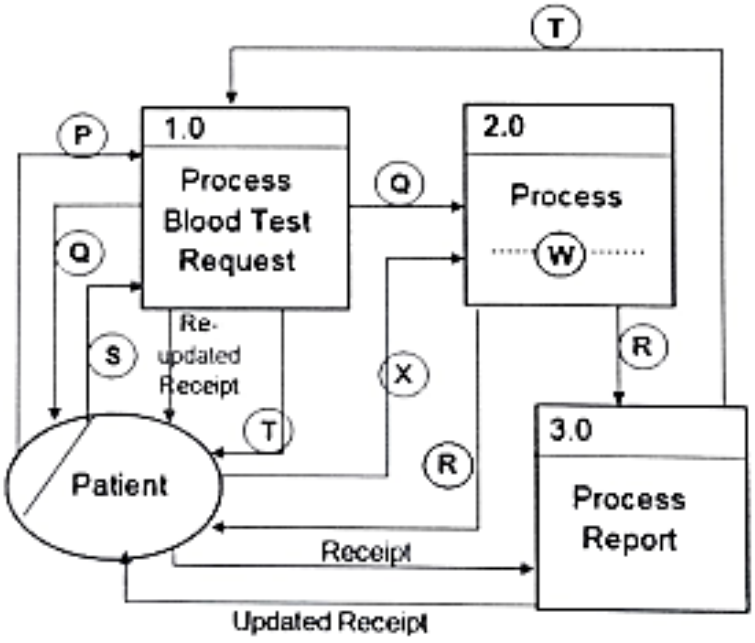


Figure 2

(A) Write a suitable term to replace the label Ⓦ in Process 2.0.

(B) Identify and write down the missing data flow labelled Ⓧ.

(b) (i) What is *requirement analysis*?

(ii) List **two** advantages of requirement analysis.

(iii) Give **one** method that can be used to verify whether a functional requirement is satisfied in a system.

(iv) The following list consists of some *functional*, *non-functional* and other requirements of a proposed school library management system where users can borrow and return books in addition to other usual tasks.

(A) The system should authenticate users through username and password.

(B) The system should enable users to search for books based on the *title*, *type*, *ISBN No.* or *publisher name*.

(C) The total cost for the library system should be less than Rs. 500 000.00.

(D) The system should be available 99% of the total time.

(E) The system development should be completed within 9 months.

(F) Book lending details should be preserved even if the system crashes during operation.

(G) The book database of the school library management system must be secured by preventing unauthorized access.

(H) Since the Past Pupils Association has indicated its willingness to develop the system, preference will be given to them.

From A to H, identify and write down the labels of **two** *functional* requirements and **two** *non-functional* requirements respectively.

2015 Paper II – Part B

6. Draw a context diagram to show the overview of the library system described below. Clearly indicate external entities and data flows of your diagram and state any acceptable assumptions that you have made.

The National Information Technology Library (NITL) provides e-books to its users through an online system named "Library Information Processing System (LIPS)".

A person should submit an application to NITL to become a member of the LIPS. The NITL evaluates the application and enters it to the LIPS, if it is approved. After entering the application data, LIPS issues an activation code to NITL which in turn passes it to the relevant person. Once the activation code is received the person becomes a member of LIPS. A member can obtain his/her username and password by providing the activation code to the LIPS. A member can subsequently access e-books by entering his/her username and the password to the LIPS.

2016 Paper II – Part B

6. Draw a context diagram to show the operations of the online assignment management system described below adhering to the standards of the Structured System Analysis and Design Methodology (SSADM). Clearly indicate external entities and data flows of your diagram. State any assumptions that you have made.

The National University of ICT (NUICT) conducts distance learning programs. The students of NUICT are continuously assessed through online assignments. The students of NUICT use an online assignment management system called NetAssign to download assignments and upload answer scripts. The examiners use NetAssign to upload assignments and to download answer scripts. Once the answer scripts are marked, the examiners enter marks into NetAssign. The students can view only the grades determined by NetAssign. To use NetAssign, both examiners and students must provide authentication details to the system. These authentication details are provided by NUICT to its students and examiners.

2017 Paper II – Part A

1. Fill in the blanks in the following Data Flow Diagram (Figure 1.1) of the sales information system of a company called Bookland, only by using the words in **Boldface** in the description given below.

Bookland is a book store that specializes in selling rare books, which are usually not available from other sources. As a practice, Bookland maintains just a one copy of any book at a time due to the higher price of rare books.

Typically, a customer makes a **book enquiry** over the phone from the **Sales Assistant** at Bookland. She then goes through **book details** in the **Inventory** file to check whether such a book is in their stock. If the details of the book is found in the Inventory file, the Sales Assistant then refers to the **Hold-on Requests** tray to make sure that the **book status** is 'available' and makes a **reply to the enquiry**. If the book status is 'available' and the **customer** wishes to reserve the book, she then takes customer's **personal details** and places a **hold-on request** against that book in the Hold-on Requests tray making the book no longer available.

When the customer comes to make the payment and collect the book, the **Cashier** at Bookland refers to the Hold-on Requests tray and finds the relevant hold-on request made by the customer. If there is a valid hold-on request, she then accepts the **payment** from the customer, issues a **payment receipt** to the customer and finalizes the sale. The customer is then allowed to take away the book. The Cashier also files the **copy of the payment receipt** in the Sales file, uses it at the end of the day to compile a **sales report** and sends it to the **Owner** of the Bookland. After every sale, the Cashier updates the book details in the Inventory file and keeps the stock up to date. When the Owner supplies books to Bookland, he sends **details of books** to Cashier and the Cashier adds them one by one to the Inventory file.

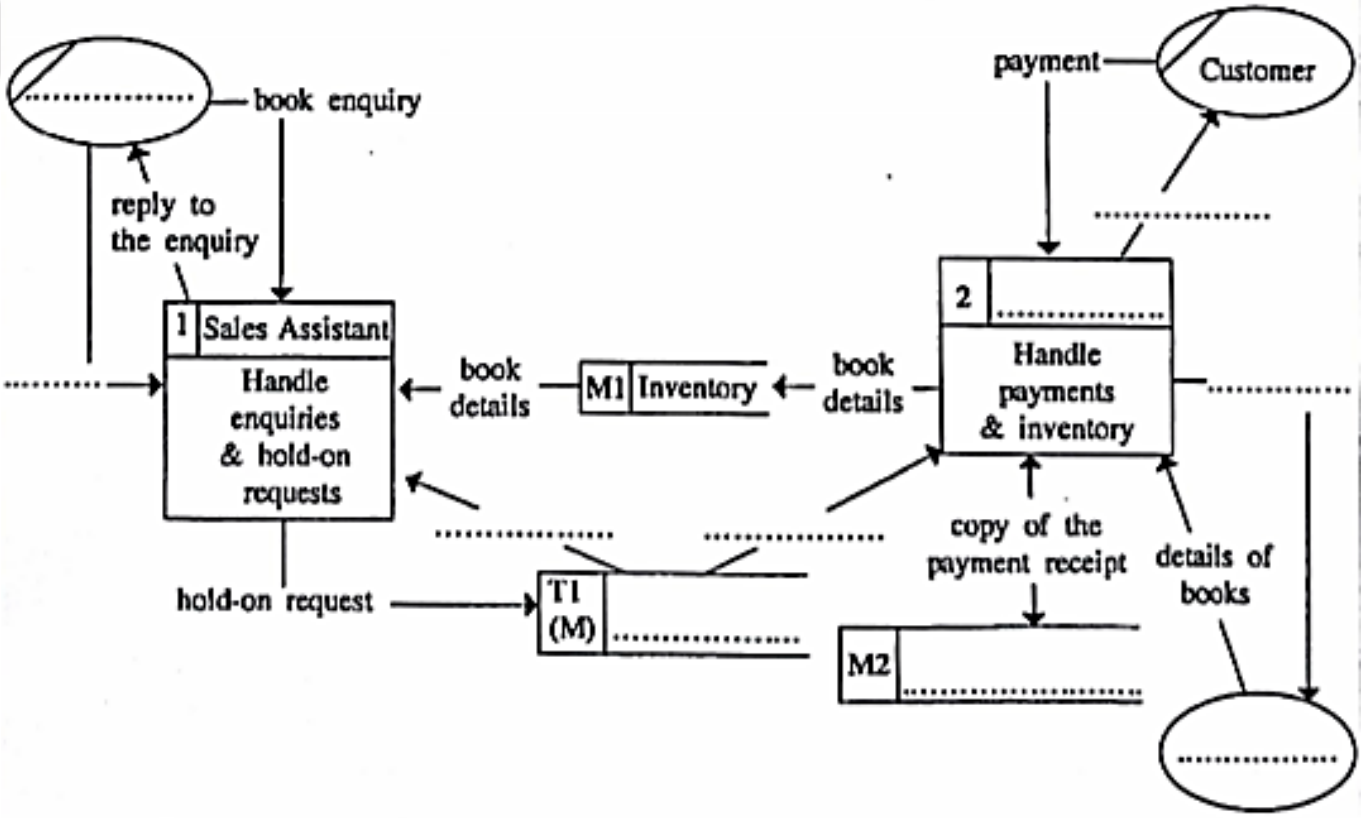


Figure 1.1: Level 1 Data Flow Diagram of Sales Information System of Bookland

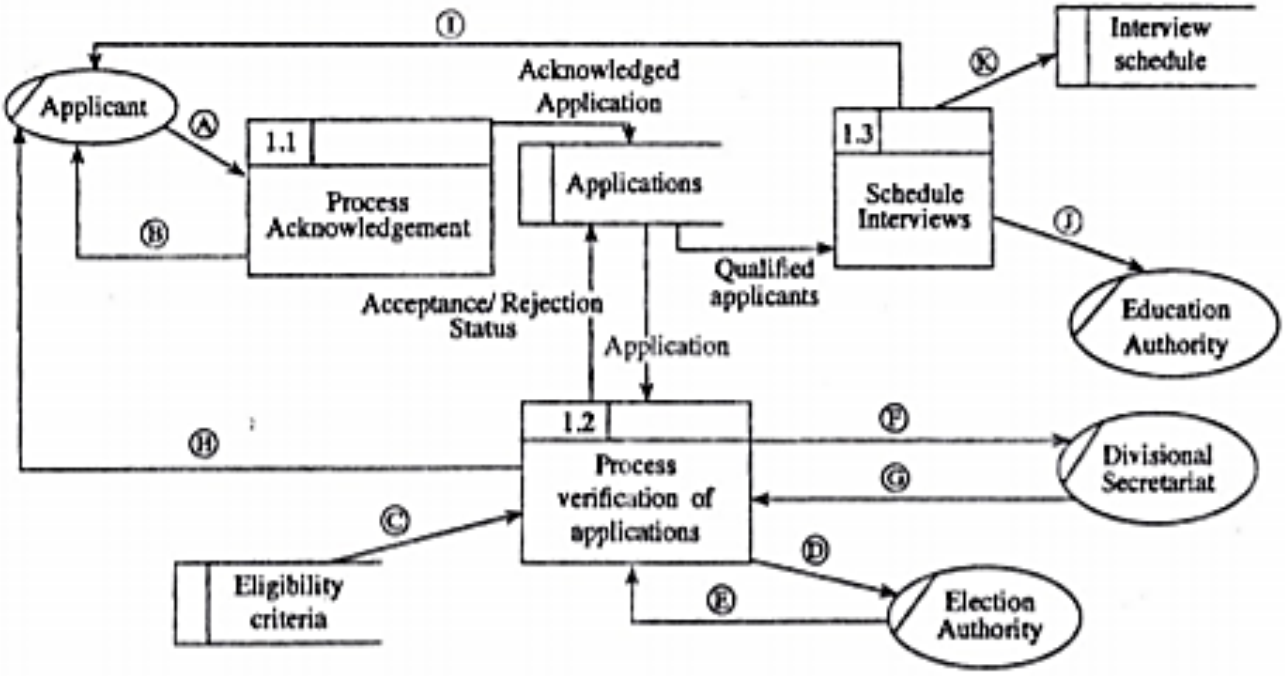
2018 Paper II – Part B

6. (a) The school admission process of a certain country is explained using the description and the data flow diagram given below.

The applicant sends the application to the respective school. The school sends an acknowledgement to the applicant. The school then verifies the information in the application by checking the following.

- Eligibility of applicant : by using the eligibility criteria taken from the data store '*Eligibility Criteria*'
- Registration in the electorate : by requesting the electoral list from the Election Authority (Election Authority sends the Electoral list to the school)
- Residential status : by requesting the confirmation of residence from the Divisional Secretariat (Divisional Secretariat sends the confirmation of residence to the school)

After verification of information, the applicant is informed whether the application has been accepted or rejected which is noted in the application and stored in the data store "Applications". The school obtains the valid applications from the data store "Applications" and schedules the qualified applicants for interviews. Then it calls the applicants for interviews and sends the interview schedule to the Education Authority. The interview schedule is stored in the data store "Interview Schedule".



Level 1 DFD

The *Level 1* Data Flow Diagram for the above scenario with some data flows labelled as Ⓐ - Ⓚ is given in figure. Identify and write down the relevant data flows against the labels Ⓐ - Ⓚ.

(b) (i) Briefly explain the key difference between functional and non-functional requirements as used in the system development life cycle.

(ii) The following list includes some functional and non-functional requirements of a proposed e-commerce web portal that plans to sell products on a catalogue:

A – Enable user to find products based on a variety of item characteristics

B – The system should work on any web browser

C – The system should be easy to use

D – Enable user to submit his/her comments on products and read other users' comments on items

E – Data in the system should be preserved even in the case of a system failure

F – Enable user to create and maintain a wish list of desired products

G – Enable user to browse through products on catalogue

H – The system should be available for use 24 hours a day, 7 days a week and 365 days an year

I – The system should authenticate users through usernames and passwords

J – The system should have versions customized for global users, e.g., French, Japanese, German, etc.

Identify and write down the labels of the **non-functional** requirements in Ⓐ - Ⓙ.