System theory: Definition of a system

A system is a set of components that interact with one another for some purpose.

A system may be considered as an assembly of components/part united by some form of regulated interaction to form an organized whole. An organized set of procedures requires accomplishing a specific function.

TYPES AND EXAMPLES OF SYSTEM

Society and nature amount in system eg. Digestive system, nervous, circulatory system, etc. society organizes legal systems, political systems, educational system, etc. Organizations have information order systems, personnel data systems. Hospitals have record keeping systems, health insurance systems.

TYPES OF SYSTEM

(a) Abstract system: This is conceptual. It is a product of human mind. It is not a system that can be seen or pointed to as an existing entity. Eg. Social systems, theological, cultural systems etc. None of these entities can be photographed or drawn/otherwise physically pictured. However, they do exist and can be discussed, studied and analyzed.

(b) Physical system: This is a set of elements rather than ideas that operate in relation to each other to accomplish a common goal/purpose. Eg. Computer systems and communication systems. Computer systems are collection of hardware elements that work interdependently under some means of control to process data and produce output. Communication systems are collection of components that can represent and transmit bits of information from one point to the other.

SYSTEM ELEMENTS

With the basic definitional framework, the elements that are necessary for the very existence of a system may be identified as follows

1. Environment
2. Boundaries
3. Input/output
4. Input-process-output
5. Subsystems
6. Interface

Environment: All system operates within an environment. The environment surrounds the system both effecting and be affected by it. The environment defines its external relationship. Close systems do not interact with their environment. Open systems interact with their environment by taking information and putting it out. They are dependent on the environment and sensitive to changes within the environment.

System Boundaries: Boundaries separate the environment from the system. The system exists within the boundaries and anything lying outside that constitutes the environment. The system boundary line determines what is included within the system and what is not.
**Input/output:** The system interacts with the environment by means of input and output. Input is anything entering the system from the environment. Output is anything leaving the system and crossing the boundaries to the environment. Eg. In a computer system, data enters the system as input and leaves the system as processed result.

**Sub-system/system components:** System components are smaller systems lying within a bigger system. These smaller units work together with each other to accomplish the goals of their individual units and also the goal of the larger system.

**Input-processing-output:** A system has input, processing and output. The processes are methods of converting inputs into outputs.

**Interfaces:** Interfaces are the meeting points for systems/sub-systems. In other words, interfaces are created when systems or sub-systems boundary meet.

This usually involves some form of resource exchange often in the form of an input-output relationship.

---

![Diagram](image)

**THE RELATIONSHIP OF SYSTEM THEORY TO THE STUDY OF SYSTEM ANALYSIS AND DESIGN**

The following aspect of system theory explains why some techniques are adopted in system projects, ie, the environment, interface, system boundaries, input-process-output characteristics. When a system is to be designed and implemented, it has to be operational in a conducive environment. In an environment where it will take its input from and also send its output to, the design techniques should take into consideration the environment in which the system can be successfully implemented. Another aspect is the input-process-output characteristics. System analysis and design relies on input to the information system and the process that should emanate the output to the audience. Therefore, the input should be identified and the process should be identified in order to get the desire output. The output then will determine the type of output design for the system, either print/electronic. The system boundaries will define any limit and constraint that the system to be designed will have. The boundaries will also determine the type of interfaces that the new system to be designed will have in order to be operational.
INFORMATION SYSTEMS

This is the arrangement of people, data, processes, information presentation and information technology that interact to support and improve the day-to-day operations in a business as well as support the problem solving and decision making needs of management and users. Examples are payroll, inventory account receivable systems, etc., sales system.

SYSTEM ANALYSIS AND DESIGN, THE CONCEPT

The term system analysis and design is usually the process of analyzing business procedures with a view of using computer as a tool for improving efficiency and effectiveness. When broken down into analysis and design, the analysis is defined as a problem solving methodology that decomposes a system into its component pieces for the purposes of studying, how well these component parts work and interact to accomplish their purpose.

System Design also called system synthesis is a problem technique that re-assembles the system component pieces into an improved system. The concepts have evolved from the classical approach (system life cycle) to the use of structure system analysis and design approach and object oriented-approach to provide effective and efficient information systems. These techniques are used by different information workers/stakeholders.

INFORMATION WORKERS

Information workers responsible for the development of system projects are:

1. System owners
2. System users
3. IT Vendors and Consultant
4. System designers
5. System builders
6. System analyst

1. SYSTEM OWNERS: They pay for the system to be developed and maintain. They are the owners of the system and they determine the working framework and design policies for the use of the system. Owners also participate in the use of system analysis and design. They usually initiate the system process and provide information for the fact finding stage of the analysis.

2. SYSTEM USERS: They use the system to perform/support the work of the system. They also participate in system project by defining business requirement and performance expectation.

3. IT VENDORS AND CONSULTANTS: They sell hardware and software services to businesses for incorporation into their information systems but are useful when it comes to selecting hardware and software for the organization.

4. SYSTEM DESIGNERS: They design the system to meet the user requirement. They design all the necessary specification, database files, network etc. examples are database administrators, network architects, security experts, etc.

5. SYSTEM BUILDERS: They construct test and deliver the system into operation. Examples are application programmers, network programmers, system programmers, software integrators etc.
6. SYSTEM ANALYST: They facilitate the development of information system through the interaction of other information workers. They understand both business and computing. They study business problems and opportunities and then transform business and information requirement into specifications for information systems that will be implemented by various technical specialist.

SKILLS OF THE SYSTEM ANALYST

In addition to formal system analysis and design skills, the system analyst must also have the following knowledge, skills and traits.

1. Working knowledge of information technologies: The system analyst must know the technologies that are in book and that are for the future. He must consult IT manuals such as computer world and constantly visit website that shows the trend in IT and also IT for the future.

2. Computer programming experience and expertise: The system analyst must have programming experience in order to appropriately repair adequate business and technical specification for the programming.

3. Knowledge of business process and terminologies: The system analyst must be able to communicate with business experts in order to understand business problems. This skill may be acquired through basic business literacy courses in college. Such courses may include financial accounting, finance, marketing, business law, economics, etc.

4. General problem solving skills: These are skills acquired in college philosophy courses whose content may include problem solving skills, critical thinking, reasoning, etc. These skills will enable one to take a large business problem, break it down and determine problem causes and effects in order to recommend a solution.

5. Inter-personal communication skills: The analyst must be able to communicate effectively in written and verbal form. Communication skills may be learnt in college and may include technical speaking, listening, interviewing, etc.

6. Good inter-personal relation skills: The job of a system analyst involves an interaction with all stakeholders. Therefore, it requires effective inter-personal skills that allow the analyst to deal with new dynamics, business, politics, conflicts and changes. Some of the courses may include teamwork, principles of persuasion, leadership, managing change and conflict resolution.

7. Flexibility and adaptability: An analyst must be flexible and must adapt to unique challenges and situations.

8. Character and ethics: A system analyst must be able to discern between right or wrong. They must abide by standard or computer ethics. They have to be discreet about confidential information of the organization he is working for. He/she must follow the ten commandments of computer ethics as outline by the Computer Ethics Institute of USA. Example, thou shall not use the computer to harm others.
In the mid-60s, a number of large electronic data processing applications failed costing companies’ lots of money. This was due to the lack of poor system development techniques. System development methodologies were found to be important as a solution to these problems. Proposal emerged and the engineering developing process was adopted by designers. It was used for the construction and operation of various types of buildings, power transmission lines, various machines and chemical plants. The successes with which the engineers perform these led to the processes adoption by system developers. The engineering development process is summarized as planning, analysis, design, implementation and maintenance.

Based on the engineering development process, system designers came out with the system development life cycle as activities and functions that all information system developers perform. Regardless of which approach, they use all life cycle methodologies prescribe phases and activities. The number of scope of phases and activities varies from author to author, experts to experts and company to company.

### STRUCTURED ANALYSIS

Structured system analysis is an alternative to the SLC used to develop information systems. It uses the phases of the system development life cycle to plan, analyse, design, implement and support information systems. Structured Analysis uses a set of models to describe a system graphically because it focuses on process that transform data into useful information. Structured analysis is called a process centered technique. In addition to modeling the processes, structured analysis includes data, organization and structured relational databases design and user interface issues. Process modeling identifies the data flowing into the process, the business rules that transform the data and the resulting data flow. Structured analysis introduced a modeling tool called Data Flow Diagram and a design tool called Structure chart. For example, the following is a process model for a school’s registration system.
This is also another technique for developing information systems. It combines data and the processes that act on the data into things called objects while structured analysis treats processes and data as separate components.

**TERMS AND CONCEPTS OF OBJECT ORIENTED**

Object oriented analysis describes an information system by identifying things called objects. An object represents a real person, place, event or transaction, and this is similar to an entity which is a thing of principal interest.

For example when a patient makes an appointment to see a doctor, the patient is an object, the doctor is an object, and the appointment is also an object.

The end product of object oriented analysis is an object model which represents the information systems in terms of object and object oriented concepts. During the implementation phase of the SDLC, system developers can translate object oriented designs directly into object oriented programme code models using languages such as C++ and java.

**OVERVIEW OF OBJECT ORIENTED ANALYSIS**

Object models are developed using unified modeling language which is a widely used method of visualizing and documenting an information system. To be able to do this, we have to understand some basic object oriented term and how they are used to describe information system.

**Object:** This represents the person, place, event, or transaction that is significant to the information system. Example, a customer which is an object may have data such as name, address, account number and current and can perform specific task such as placing and order, paying a bill or changing an address.

**Attribute:** An object has Attribute which are characteristics that describe the object. Example, a car object may have Attribute such as make model, color etc. A student object may have attribute such as names, gender, date of birth, etc.
An object also has methods which are tasks/functions that the object performs when it receives a message or command to do so. A message is a command that tells an object to perform a certain method. These components are illustrated using a car.

**Class:** This is a collection of similar objects. Examples are a class of students, a class of ford cars, etc.

**Instance:** This is a specific person, member of the class. Example, ford explorer 350 is a specific car of the class of ford cars. Therefore, ford explorer 350 is an instance of the ford car class.

Representing the terms and concepts using Unified modeling language (UML). UML represent an object as a rectangle with the object name at the top followed by the object attributes and methods. For example is using a parent with children.

In this diagram, the child object has certain attributes such as name, age and sex. The family has three children, ie, three instances of the child object. A child object performs certain method such as eat, play, go to bed. To signal the child to perform those tasks, you should send certain messages that the child object will understand. Messages such as Kofi come and
eat, Vicky go and play, etc. Objects are similar to nouns. Attribute are similar to adjectives and method defines specific task that an object can perform. Message is a command that tells an object to perform certain method. Eg. ADD STUDENT, DIRECT A STUDENT CLASS, ADD A STUDENT NUMBER, NAME AND OTHER DATA about the student. Similarly, a message DELETE STUDENT tells a student class to delete student instance.

The student class TO ADD A STUDENT understands that, it should add a student number, name, and other data about the student as shown in the diagram.

The message ADD STUDENT signals the student class to perform the ADD Student method. The message DELETE STUDENT signals the student class to perform the DELETE Student method.

**PLANNING (EXAMS)**

Planning is made up of system request, feasibility study and preliminary investigation.

**MAIN NOTE:** Companies develop and maintain IT systems to support their current and future businesses. Some IT needs are immediate. Example is fixing a problem in a payroll system. But in introducing a computer based system, a big factory or restructuring cooperation or a merger with another company, the following issues constitute the planning phase of the SDLC.

- System request
- Feasibility study
- Preliminary investigation

**SYSTEM REQUEST**

System planning phase usually begins with a formal request to the IT department or to the systems analyst and this is called systems request. The System Request describes problems or desired changes in an information system or a business process. These days, system planning is part of the entire business planning. When managers and users develop their business plans, they usually include IT requirements that generate systems request. A system request can come from
A top manager, a planning team, a department head or IT department itself. The request can be major or minor. A major request might involve a new information system or the replacement of an existing one that cannot handle current requirement. However, a minor request might ask for a new feature or a change to the user interface. Many organizations use a special form for systems request which could be online.

<table>
<thead>
<tr>
<th>SYSTEM REQUEST FORM</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATE: ..................</td>
</tr>
<tr>
<td>SUBMITTED BY: ........</td>
</tr>
<tr>
<td>TITLE: ..................</td>
</tr>
<tr>
<td>REQUEST FOR</td>
</tr>
<tr>
<td>[ ] correction of system error</td>
</tr>
<tr>
<td>[ ] system enhancement</td>
</tr>
<tr>
<td>[ ] new system</td>
</tr>
</tbody>
</table>

| Description of Requested Services |

To be completed by the IT Department

[ ] Approved
[ ] Modified
[ ] Rejected

Assigned to IT

Contact person:

User:

Date:

Action:

A properly design form streamline the request process and ensures consistency. When a system request form is received, a system analyst or the IT manager examines it to determine what IT resources (start with time) are required for the preliminary investigation. In most large companies, a system review committee is formed to evaluate systems request.

A typical committee consist of IT director and several managers. In smaller companies, only one person, the system analyst evaluate systems request. Evaluation involves certain priorities if there are many requests requiring review. Some of the following questions are asked:

Which of the project should the firm pursue?

What criteria should be applied?

How should priorities be determined?

To answer these questions, a system analyst or committee must assess the feasibility of each system request.
FEASIBILITY STUDY

The system request must pass several tests to see whether it is worthwhile to proceed further. Feasibility study uses four (4) main yard styles to measure a proposal and these are operational, technical, schedule and economic feasibility.

**Operational feasibility:** A project is operationally feasible if it can be solved using the organization’s available (already owned or obtained) procedures and personnel. This will involve an analyst of the organization current skills base and the timing of the project. The organization may have the resources but may not want to commit to a particular project at a particular time. Operational feasibility criteria measure the urgency of the problem or the acceptability of the solution.

Two aspects of operational feasibility can be considered:

1. Is the problem worth solving or will the solution to the problem work
2. How do the end users and management feel about the problem/solution? In other words, have you got the problem or the solution?

**Technical feasibility:** Looks at what is practical and reasonable. Technical feasibility addresses three (3) major issues:

1. Is the proposed technology/solution practical?
2. Do we currently possess the necessary technology?
3. Do we possess the necessary technical expertise and is the schedule reasonable.

The organization may have the technology but may not have the skills required to properly apply that technology. Example is the organization may have a Database Management System but the kind of analyst and programmers that the organization has, may not know DBMS well enough to properly apply it. If they have to learn it will affect the schedule of the project.

**Schedule feasibility:** This deals with deadlines for completion of projects. Given the technical expertise, are we going according to the right time table? Example, in developing a system project for the government to meet new government regulations, a deadline is very necessary. Penalties associated with missing such as deadline may make meeting it mandatory. Analyst may propose alternative schedules if deadline are not mandatory but desirable.

**Economic feasibility:** A project is economic reliable if:

1. The costs of the project do not outweigh the benefits.
2. The project compares favorably in economic terms with computing uses for the organization scarce resources.

To establish this, the organization can make criteria such as break even analysis, payback period and net present value. The generic term for this type of approach is a cost-benefit analysis. The following are some of the tangible and intangible cost and benefits likely to occur as a result of the new information system.
The conclusion for the cost-benefit analysis and for facts gathered during the feasibility stage will be that the project is either feasible or not feasible. Feasible projects are then considered further by following a project plan. A project plan is a statement of scope timetable, resources and cost and a broad plan for the entire development as well as specific plans for structured analysis.

**Feasibility report**

The final document produced by the feasibility team is likely to include the following sections:

1. **Executive summary** which consist of introduction and summary of findings
2. **Description of the problem** consist of summary of interviews, questionnaires and documentation
3. **Solution objective** which consist of statement of the objective of a new or revised system
4. **Constraints and Restriction on development**
5. **Feasibility study results** consist of operational, technical, schedule and economic feasibility of the proposed system
6. **Development plans** consist of the following:
   i. Scope of development activities
   ii. Detailed list of task and activities
   iii. Timetable of task
   iv. System development team
7. **Potential solution**
8. **Recommendation**

The entire feasibility is evaluated to identify and weed out system request that are not feasible. Example, a request will not be feasible if it requires hardware and software that the company already had rejected. After rejecting system request that are not feasible, the system review committee must establish priority for the remaining items. The highest priority goes to project that produce the greatest benefits at the lowest cost in the shortest period of time.
PRELIMINARY INVESTIGATION

A systems analyst then conducts a preliminary investigation to study the systems request and recommend specific action. After obtaining an authorization to proceed, the analyst interacts with managers and users as indicated below:

SYSTEM ANALYSIS PHASE

System analysis phase is the second of the five phases of the system development life cycle. In the system planning phase, preliminary investigation to learn more about systems request was done. In this phase, the process of gathering facts about the system project, preparing documentations and creating model that will be used to design and develop the system. The overall objective of this phase is to understand the current system. The systems analysis phase includes three (3) main activities and these activities are Requirement modeling, Data and Process modeling, and Development Strategies.

SYSTEM ANALYSIS PHASE: MAJOR ACTIVITIES
**Requirement Modeling:** This involves fact finding to describe a current (prevailing, old, existing or present) system and identification of the requirements for the new system such as outputs, inputs, processes, performance and security. Output refers to the electronic or printed information produced by the system. Input refers to necessary data that enters the system either manually or in an automated form. Processes refer to the logical rules that are allowed to transform the data into meaningful information. Performance refers to system characteristics such as speed, volume, capacity, availability and reliability. Security refers to hardware, software and procedural control to safeguard and protect the system and its data from internal and external threat.

**Data and Process Modeling:** This is when data and processes are represented graphically using traditional Structured Analysis Techniques. Structured Analysis identifies the data flow into the process, the business rules that transforms the data and the resulting output data flow. These techniques are used to develop a logical model of the proposed (new) system and document system requirement. A logical model shows what the system must do regardless of how it will be implemented physically. Later in the design phase, a physical model is built that describes how the system will be constructed. The data and process modeling involves three (3) main tools: a data flow diagram, a data dictionary and process descriptions. The system analyst can also use object oriented analysis, information engineering such as entity relationship diagram or flow chart.

**Development Strategies:** This is the final part of the analysis phase. Activities involved in this include: evaluation of alternative solutions, preparation of the system’s requirement documents and the presentation of the system requirement document to management. System analysis skills required here are analytical skills and inter-personal skills. In addition, because information systems affect people throughout the company, team orientation strategies have to be considered. The traditional model for system development was an IT department using a structured analysis and consulting users when they needed their input. But the contemporary position is that, IT managers invite system users to participate actively in various development tasks. One technique is the Joint Application Development (JAD) which is team oriented technique for fact finding and requirement modeling because they are not linked to a specific development methodology. Developers use JAD whenever group input and interaction are desired. Another popular user oriented method is Rapid Application Development (RAD). This is like a condensed version of the entire SDLC with users’ involvement every step on the way. Whiles JAD focuses only on fact findings and requirement determination, RAD provides a fast track approach to a full spectrum of system development task including planning, design, construction and implementation.

**SYSTEM REQUIREMENT DOCUMENTS**

A system requirement is a characteristic or feature that must be included in an information system to satisfy business requirement and be acceptable to users. System requirement serves as a bench mark to measure overall acceptability of the finished system. It contains the requirement for the new system, describes the alternatives that were considered and makes a specific recommendation to management. Some examples of system requirement are as follows:
1. **Output**: The website must record online volume statistics every four (4) hours and hourly during peak periods.

2. **Input**: The department head must enter overtime hours on a separate screen.

3. **Process**: The human resources system must interface properly with the existing payroll system.

4. **Performance**: The system must support 30 users online simultaneously.

5. **Controls**: The manager of the sales department must approve orders that exceed the customer’s credit limit.

**Attachment 1 (Cont…..)**

**SYSTEM ARCHITECTURE**: Seven specific issues affect the architecture of the choice of a system analyst and these issues are:

1. Enterprise Resource Planning
2. Initial Cost and Cost of Ownership
3. Scalability
4. Web Integration
5. Legacy System Interface Requirement
6. Processing options
7. Security Issues

1. **Enterprise Resource Planning**: Enterprise Resource Planning is used to establish a company wise strategy for using IT resources. Enterprise Resource Planning defines a specific architecture including standards of data, processing network and user interface design. The main advantage of Enterprise Resource Planning is that it describes a specific hardware and software environment which is also called a platform that ensures an activity and easy integration of future systems including in-house software and commercial packages.

2. **Initial Cost and Cost of Ownership**: In system planning and analysis, there was economic feasibility. During the final design, decisions that will have a major impact on the initial cost and total cost of ownership for the new system are made. All previous cost estimates are reviewed by asking the following questions: (a) if a specific package was chosen initially, is it still the first choice? Are newer versions or competitive products available? Have any changes occurred in pricing and support? (b) Have any new type of outsourcing become available? (c) Have any economic, government or regulatory event occurred that could affect the proposed project? (d) Have any significant technical development occurred that will affect the proposed project? The answers to these questions might affect the initial cost and total cost of ownership for the propose system. Re-analysis of system requirement and alternatives are done before the design of systems architecture.

3. **Scalability**: This is also called extensibility. This refers to a systems ability to expand, change or downside easily to meet the changing needs of a business enterprise.
4. **Web Integration**: Systems Analyst determines if a new application will be part of an e-commerce strategy and decide on the degree of integration with other web-based components. A web-centric application can run on the internet, company intranets or extranets.

5. **Legacy System Interface Requirement**: When designing a system, an analyst must determine how the new application will communicate with existing legacy system. (Legacy system are older systems that runs on mainframe computers)

6. **Processing options**: Designers will have to consider how the system will process data whether online or in batches. For example, a high capacity transaction system such as an order entry system requires more network, processing and data storage resources than a monthly billing system that handles data in batches.

7. **Security Issues**: Security is a concern at every stage of system development. As the logical and physical designs are translated into specific hardware and software, the system analyst must consider security issues that relates to system design specifications and determines how the organization will address them.

---

**SYSTEM SPECIFICATION DOCUMENT**

**The structure of system design specification**

The system design specification is also called the *technical design specification*. It is a document that presents the complete design for the new information system along with detailed cost, staffing and scheduling for the implementation phase. It is the baseline against which the operational system will be measured. Unlike the system requirement document which is written for user to understand, the system design specification is oriented towards the programmers who will use it to create the necessary programme. The structure of a system design specification is made up of the following.

1. Executive summary
2. System component
3. System environment
4. Implementation requirement
5. Time and cost estimates
6. Appendices

1. **Executive summary**: System specification starts with an executive summary which provides a brief overview of the project for company manager and executives. It outlines the development efforts to date. It provides a current status report. It summarizes current project costs and costs of the remaining phase. It also has reviews of the overall benefits of the new system. It represents the system development phase schedule and highlights any issue that management needs to address.
2. **System component:** This section contains the complete design for the new system including the user interface, output, input, files, database and network specification as well as source document report and screen layout. Data flow diagrams and all other relevant documentation should be included. It also must include requirement for processing, backup and recovery, startup processing and file retention as well as software information.

3. **System environment:** This section describes the constraints or conditions affecting the system including any requirement that involve operation, hardware, system software or security.

4. **Implementation requirement:** This specifies startup process, initial data entry or acquisition user, user training requirement and software test plans.

5. **Time and cost estimates:** This section provides detailed schedules, cost estimates and staffing requirement for the system development phase and reviews projection for the make up of the system development life cycle. It also looks at total cost to date for the project and compares those cost with earlier estimate.

6. **Appendices:** Supplementary can be included in appendices. At the end of the system design specification documents, from the first three (3) phases that may provide a helpful reference for reader can be included.

**IMPLEMENTATION (EXAMS)**

Introduction: System implementation involves the following

(i) Testing  (ii) Training  (iii) File conversion  (iv) System changeover

(i) **TESTING A NEW INFORMATION SYSTEM**

These tests usually involve analyst, owners, users and builders. The system analyst facilitates the completion of this task and typically communicates testing problems and issues with the project team members. The system owners and users hold the ultimate authority on whether or not a system is operating correctly. System builders of various specialties are involved in the system testing. For example, application programmers, database programmers and networking specialist may need to resolve problems revealed during system testing.

Several types of testing are done in the following sequence:

1. **Realistic test:** The system is presented with a realistic example with the environment in which the system is to operate. It tests the system and the understanding and training of users. It also gives the users confidence before they take over a system.

2. **Contrived test:** This test deals with as many unusual and unexpected events as possible such as incorrect codes, wrong amounts, and inappropriate commands and so on. The intention is to see how the system reacts and whether all answerable anomalies have been catered for in the system.

3. **Volume test:** This test is about presenting the system with a large volume of transaction to see how it reacts particularly in operating a response times.
4. **Acceptance test**: This is undertaken by users after all other system testing is complete. It is designed to test a complete system so that the users can see if the new system is working satisfactorily. As the final test, it is probably the most important and elaborate. It is performed by the users using realistic data over an extended time period. It is an extensive test that addresses three (3) levels of acceptance testing, verification testing, validation testing and audit testing.

(ii) **TRAINING**

After the test, it is followed by training. No system can be successful without proper training whether it involves software, hardware, etc. A successful information system requires for users, managers and IT staffs. For the training, the organization selects the personnel who will both operate and manage the new system and must train them in the use of it and its related activities. The organization must select the appropriate training delivery method depending on who is being trained. So for users, the training will be different.

**USER TRAINING**- Training for users will include the following:

1. If the information system is manual and has been computerized, then users will need training in basic computer literacy.
2. Users will have to learn how to use specific applications and models quickly and in great detail. Examining important procedures commands and data entry requirements.
3. Users should have on the job training, that is, training while they are actively using the new system.
4. Training updates may be required as the users become more familiar with the system and require further knowledge and skill development and consolidation.

**GUIDELINES FOR DEVELOPING A TRAINING PROGRAMME**

When developing a training programme for users, the following guidelines should be kept:

a. Train people in groups as this is a better use of your time and it encourages group learning possibilities. It also helps to practice specific skill as common problems and issues will be addressed.

b. Select the most effective place to conduct the training. Training employees at your company’s location offers several advantages. Employees incur no travel expenses and training can take place in the actual environment where the system will operate.

c. Provide for learning by hearing, seeing and doing. Some people learn best from lectures, discussions and question and answer sections. Others learn best from doing demonstration or from reading documentation and other materials. Most people learn best from hands-on experience. Therefore, training should be provided to support each type of learning.

d. Prepare effective training materials including interactive tutorials and user manuals.
MIDDLE MANAGEMENT TRAINING

Middle management will be trained on elements of the system for which they are responsible. They will need an understanding of a particular business issues, security and control features related to the particular system.

SENIOR MANAGEMENT TRAINING

Senior management should be trained on a much structured manner and should be business focused. Training takes the form of short demonstration, power point presentations, video demonstration and executive seminars.

(iii) FILE CONVERSION (EXAMS)

Files and programmes need to be converted into a suitable format for the new system and for completing documentation. It involves converting large amount of manual files into computer based files by means of transcription on specially designed forms which then have to be keyed into the computer individually. New files must be checked thoroughly for accuracy and completeness. Also, if the files are already computerized, then the above difficulties are usually reduced. Transcription of old computer files to a new computerized is carried out using conversion programmes. Sometimes, a computer bureau is used to carry out a conversion process.

(iv) SYSTEM CHANGEOVER

1. Parallel approach
2. Direct approach
3. Pilot approach
4. Modular approach

Four (4) main methods of system changeover can be used and these are named above.
1. **PARALLEL APPROACH**
   This is the most common form of changeover where the old and the new system operate together for a period of time processing the same current data. The output of the two (2) systems can be compared to determine whether the new system is operating as expected and that there are no processing errors occurring.

![Parallel Approach Diagram]

2. **DIRECT APPROACH**
   This is when the new system completely replaces the old system. The old system ceases to operate. This is a risky business. (It has the highest risk)

![Direct Approach Diagram]

3. **PILOT APPROACH**
   There are two (2) types of changeover under this approach. They are restricted data and retrospective data.
   - **Restricted data:** This involves taking one (1) whole part of the complete system and running it as the new system. If it operates properly, then the remaining elements of the system can be transferred gradually.
   - **Retrospective data:** This involves operating the new system with data already processed by the existing system. The results produced by the new system can be pre-processed results from the existing system.

4. **MODULAR APPROACH**
   This is often used by large system projects or in organization which are geographically dispersed. It involves implementing one sub-system at a time or the whole system into one organization unit at a time. It is a gradual approach. Example, an organization may implement a new accounting information system by first converting the sales order sub-system, then the customer account sub-system, then the purchase order sub-system, etc. Alternatively, an organization may implement a complete system but in one geographical
location at a time. For example, the implementation of a new backing customer enquiring
system branch by branch. Implementation at any branch could be direct or parallel.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

MODULE 1
CONVERTED TO
BRANCH A

MODULE 1
CONVERTED TO
BRANCH B

MODULE 1
CONVERTED TO
BRANCH C

SYSTEMS OPERATION SUPPORT AND SECURITY

This is made up of (1) **post implementation review** and (2) **system maintenance**.

This involves making sure after systems implementation that the system is meeting user needs.
Two (2) major activities are involved which are listed above. The post implementation review is
carried out soon after the systems implementation. The system maintenance is performed in
response to specific user needs or as a result of ongoing system development.

**Post Implementation Review:** This should be carried out as soon as the new system is fully
operational and fully functional, possibly between one (1) month and one (1) year after
changeover. The review actually examines the processes, procedures and effective running of the
information system to establish the satisfaction of user needs, the performance of the new system
and also review the original cost benefit analysis. A post implementation report is then written
and it has the following structure.

**STRUCTURE OF A POST IMPLEMENTATION REPORT**

The structure of a post implementation report may include the following:

1. The system goals and an analysis of how successful the new system achieves those goals.
2. A summary of the overall quality of the system.
3. A summary of those areas where the system is considered to be unsatisfactory together
with recommendation for improvement.
4. An assessment of overall system performance and system development process and
recommendations for improvement if necessary.
5. A cost benefit analysis of comparing the cost benefits identified at the feasibility study
stage with actual cost and benefits.
6. Final section summarizing the recommendation for improving the performance of the
system and recommendation for improving future system development projects.
**System Maintenance:** There are four (4) types of system maintenance and they are corrective maintenance, perfective maintenance, adaptive maintenance and preventive maintenance.

**CORRECTIVE MAINTENANCE:** This is carried out in order to correct errors within a system and it is normally carried out in response to a problem. (It is reactive). Its main function is to ensure that the system can continue to operate.

**PERFECTIVE MAINTENANCE:** This is carried out in order to improve the performance of an application so that the performance is enhanced and inefficiencies are eliminated. It is often carried out in order to extend user capabilities or make user interface more effective.

**ADAPTIVE MAINTENANCE:** This is carried out in order to adjust application to reflect changing business operations and environmental opportunities or threats. This type of maintenance is unlikely to occur soon after implementation rather it is likely to a more mid to long term maintenance process.

**PREVENTIVE MAINTENANCE:** It requires analysis of areas where trouble is likely to occur. The IT department normally initiates this. It often results in increased user satisfaction, decreased down time and reduced total cost of operation.

**SYSTEM DEVELOPMENT PROCESS (GANIT CHART)**

<table>
<thead>
<tr>
<th>ID</th>
<th>TASK NAME</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
<th>Jul</th>
<th>Aug</th>
<th>Sep</th>
<th>Oct</th>
<th>Nov</th>
<th>Dec</th>
<th>Jan</th>
<th>Feb</th>
<th>Mar</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Project Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>System Planning</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>System Analysis</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>System Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Implementation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Operation Security and Support</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SYSTEMS FLOW CHART (EXAMS)

This is basically a physical modeling tool that uses various symbols to identify input and output, operations, representing data or files and show media such as disk, document and reports. It shows the overall structure of an information system. It traces the flow of information and work.

Advantages

1. It helps to define procedures and operations in an information system.
2. It highlights physical media used in the system as well as the sequence of activities.
3. Avoids unnecessary duplication of actions.
4. Helps with allocation of resources of staff to various jobs in the organization.
5. It is used to show all inputs, major files, processing and output for a system. It is therefore used in systems where the information flow entails large number of documents. The flow chart therefore helps to show origination, processing and designation of each document and the procedures employed by users.
6. It is flexible and versatile.

Disadvantages

1. It provides little details on how processes are actually accomplished. In other words, it does not show the details of programmes.
2. It reduces an entire programme or set of programmes to a single box.

SYMBOLS

Most flow chart use a standard set of symbols developed by the American National Standard Institute (ANSI). The symbol shapes indicates their meanings.
NB: - It should ne vertical

EXAMPLE: Ernest is copying Emma’s note.

Compiling lists of names of students from three (3) different halls.
Example: You have been employed by UG to assist with the Record keeping system at the registry. Create a database of all first year students from source document.

**DATA FLOW DIAGRAM (DFD) (EXAMS)**

A data flow diagram is a path for data to move from one part of the information system to another. It uses various symbols to show how the system transforms input data into useful information. It is basically a graphical representation of how data flows in an information system. It is a good starting point for physical designing. It shows the logical sequence of a system. It shows data flow to or from within an information system and the process that transform the data. It also shows where data is stored. It enables non-technical people to understand the system. It can be used to describe an old system and at the same time a new system.
DATA FLOW DIAGRAM SYMBOLS

Data flow diagram uses four (4) basic symbols that represent the following:

1. Processes
2. Data flow
3. Data store
4. Entity/destination/source/input

Several different versions of data flow diagram exist but the two (2) major ones are:

1. GANE AND SARSON
2. YOURDON

Drawing a data flow diagram

1. Identify the four (4) components
2. Identify all the entities
3. Identify data flow by asking yourself the kind of data that comes from the source/entity
4. What kind of transformation should take place or process to produce the required output
5. What kind of transformed data should be stored
Drawing a data flow diagram in a payroll system

Entities are:

1. HR  
2. Account department  
3. Employees  
4. SSNIT  
5. GRA  
6. Bank

**WARNIER-ORR DIAGRAM (EXAMS)**

It is used to show structure in terms of hierarchy in an information system where the entities are broken down.

(NB- student information may have entities like name, address, gender, date of birth, etc. course may have entities like SID, course ID, course name, credit hour, level, etc. residential status may also have entities like SID, hall, room number, etc. academic may also have entities like SID, semester (1st and 2nd), academic year, course name, course code, grade, etc.)
The diagram was discovered Lean D. Warnier and was enhanced by Ken Orr, hence Warnier-Orr diagram. The diagram is used to decompose a process or data by representing the hierarchical structure of the data within a system. When it is used to represent the structure of data in a system, it is called Warnier-Orr Data Structure Diagram. But when it represent the task that a system must perform, it is called Warnier-Orr Processed Diagram.

**Advantages**

1. It is useful for showing the relationship between the different task and between different data items.

**Disadvantages**

1. It does not lead itself to modeling techniques.

**ENTITY RELATIONSHIP DIAGRAM**

An entity is a person, place, thing or event for which data is collected and maintained. Entities are data objects that have an independent life of their own and therefore constitute the element of principal interest for the user of the system. For instance, entities may be students, customers, employees, etc.

**Attributes:** They describe the entity. Examples are name of student, product description, etc.

**Key Attribute:** It is the attribute that uniquely identifies the entity. Examples are student ID, employee number, matrix number, etc.

In some cases, there can be more than one attribute. An example is periodical may have the same ISSN.

**Relationship:** It is the connection between entities which is normally expressed as “is-married-to” and/or vehicle number-assigned-vehicle.

There are three (3) types of relationship:

1. **One-to-one**
   Examples are vehicle registration number assigned to a vehicle, HOD heads department, etc.
2. **One-to-many**
   Examples are parents bearing children, customer places order, etc.
3. **Many-to-many**
   Examples are students enroll in class, student registers courses, etc.
Entities of an airline system: