

Answer-1:

Management Information System(MIS)

Computer-based or manual system that transforms data into information useful in the support of decision making. 'MIS' is a planned system of collecting, processing, storing and disseminating data in the form of information needed to carry out the functions of management. MIS can be classified as performing three functions:

- (1) To generate reports-for example, financial statements, inventory status reports, or performance reports needed for routine or non-routine purposes.
- (2) To answer what-if questions asked by management. For example, questions such as "What would happen to cash flow if the company changes its credit term for its customers?" can be answered by MIS. This type of MIS can be called [Simulation](#).
- (3) To support decision making. This type of MIS is appropriately called [Decision Support System \(DSS\)](#). DSS attempts to integrate the decision maker, the data base, and the quantitative models being used.

For the design of an Information System, it is important to understand the requirements, carry out system study and analyze information. After undergoing this Course, the student will be able to study, analyze and design a system for the user. Thus, System analysis have a role in building MIS.

Answer-2

Systems development life cycle (SDLC)

The systems development life cycle (SDLC) is a conceptual model used in [project management](#) that describes the stages involved in an information system development project, from an initial feasibility study through maintenance of the completed application.

Various SDLC methodologies have been developed to guide the processes involved, including the [waterfall model](#) (which was the original SDLC method); rapid application development ([RAD](#)); joint application development ([JAD](#)); the fountain model; the [spiral model](#); build and fix; and [synchronize-and-stabilize](#). Frequently, several models are combined into some sort of hybrid methodology. Documentation is crucial regardless of the type of model chosen or devised for any application, and is usually done in parallel with the development process. Some methods work better for specific types of projects, but in the final analysis, the most important factor for the success of a project may be how closely the particular plan was followed.

In general, an SDLC methodology follows the following steps:

1. The existing system is evaluated. Deficiencies are identified. This can be done by interviewing users of the system and consulting with support personnel.
2. The new system requirements are defined. In particular, the deficiencies in the existing system must be addressed with specific proposals for improvement.
3. The proposed system is designed. Plans are laid out concerning the physical construction, hardware, operating systems, programming, communications, and security issues.
4. The new system is developed. The new components and programs must be obtained and installed. Users of the system must be trained in its use, and all aspects of performance must be tested. If necessary, adjustments must be made at this stage.
5. The system is put into use. This can be done in various ways. The new system can be phased in, according to application or location, and the old system gradually replaced. In some cases, it may be more cost-effective to shut down the old system and implement the new system all at once.
6. Once the new system is up and running for a while, it should be exhaustively evaluated. Maintenance must be kept up rigorously at all times. Users of the system should be kept up-to-date concerning the latest modifications and procedures.

System Analysis is an important Phase in SDLC. In this phase, the current system is studied in detail. A person responsible for the analysis of the system is known as analyst. In system analysis, the analyst conducts the following activities.

Needs Analysis

This activity is known as requirements analysis. In this step the analyst sums up the requirements of the system from the user and the managers. The developed system should satisfy these requirements during testing phase.

Data Gathering

In this step, the system analyst collects data about the system to be developed. He uses different tools and methods, depending on situation. These are:

Written Documents

The analyst may collect the information/data from written documents available from manual-files of an organization. This method of data gathering is normally used if you want to computerize the existing manual system or upgrade the existing computer based system. The written documents may be reports, forms, memos, business plans, policy statements, organizational charts and many others. The written documents provide valuable information about the existing system.

Interviews

Interview is another data gathering technique. The analyst (or project team members) interviews, managers, users/ clients, suppliers, and competitors to collect the information about the system. It must be noted that the questions to be asked from them should be precise, relevant and to the point.

Questionnaires

Questionnaires are the feedback forms used to collect Information. The interview technique to collect information is time-consuming method, so Questionnaires

are designed to collect information from as many people as we like. It is very convenient and inexpensive method to collect information but sometimes the response may be Confusing or unclear and insufficient.

Observations

In addition to the above-mentioned three techniques to collect information, the analyst (or his team) may collect Information through observation. In this collect technique, the working, behavior, and other related information of the existing system are observed. It means that working of existing system is watched carefully.

Sampling

If there are large numbers of people or events involved in The system, we can use sampling method to collect information. In this method, only a part of the people or events involved are used to collect information. For example to test the quality of a fruit, we test a piece of the fruit.

Data Analysis

After completion of "Data Gathering" step the collected data about the system is analyzed to ensure that the data is accurate and complete. For this purpose, various tools may be used. The most popular and commonly used tools for data analysis are:

- DFDs (Data Flow Diagrams)
- System Flowcharts
- Connectivity Diagrams
- Grid Charts
- Decision Tables etc.

Analysis Report

After completing the work of analysis, the requirements collected for the system are documented in a presentable form. It means that the analysis report is prepared. It is done for review and approval of the project from the higher management. This report should have three parts.

- **First**, it should explain how the current system works.
- **Second**, it should explain the problems in the existing system.
- **Finally**, it should describe the requirements for the new system and make recommendations for future.

Systems Analysis Techniques

- [Aims and Objectives](#)
- [Requirements Elicitation](#)
- [Data Flow Modelling](#)
- [Normalisation](#)

Aims and Objectives

There has always been a blurred borderline between techniques for systems analysis and techniques for systems design. In this document analysis techniques are considered relevant for requirements elicitation and for modelling existing systems. The most appropriate techniques for modelling existing systems are Data Flow Modelling and Normalisation, however we begin by discussing the nature of corporate information and by discussing the interview as a technique for requirements elicitation.

[Contents Page](#)

[Home Page](#)

Requirements Elicitation

Prior to discussing the techniques available for identifying and recording information a brief description of the nature of organisational information will be useful.

Flow of Information

Information is the life blood of many organisations. Information flows into and out of organisations and between the different levels in an organisation. Typically at the lower levels of an organisation, information is very detailed, e.g. relating to individual customers, orders, suppliers, invoices etc. As information flows up the hierarchy of an organisation it tends to become summarised. As an example consider a banking environment. A teller is interested in the specific information about the account they are currently dealing with, such as account number and balance. At the end of each day the branch manager may receive a summary report showing the total of all balances of accounts at that branch, together with a short list of those individual customers with balances of less than -£500 or greater than £5,000 (an example of exception reporting). At the end of each week the area manager or director may receive a list of customers with balances of greater than £10,000. The flow of information in an organisation is described in more detail in Unit 3 (Organisations and Information technology). It is important in this context because much of the work of the systems analyst/designer is concerned with identifying the flow and structure of information within an organisation.

Different Types and Levels of Data/Information

There is a distinction between data and information which can be simply described by the statement 'information is data which has been processed such that it becomes meaningful'. In other words information is data which has been placed in a specific context. Consider the number 153. On its own this is meaningless data. When placed in a particular context, e.g. £153, the number becomes more meaningful because we now know that we are referring to 153 pounds. However this is still data rather than information. £153 only becomes information when you find out that it is the balance of account number 01234567.

Uses of Information

There are many uses of information :-

- To fulfil legal requirements;
- To provide background knowledge;
- Decision support (better information results in better decisions);
- Enquiry processing;
- Analysis of trends (forecasting).

Standard Documents & Sources of Data/Information

There are many standard documents commonly in use in organisations, some have been mentioned already, e.g. profit and loss statements, balance sheets, bills, payslips. Others include order forms, application forms, delivery notes, invoices, business letters etc. These can provide very useful information to the systems analyst. Other sources of data include existing computer systems and their documentation, the internet, tables of data in magazines and newspapers

Information Gathering Techniques / Interviews

The purpose of an interview is to identify, how a person currently does their job (how the existing system works), the problems they face (what is wrong with the existing system) and how they would like to do their job (what is required of the new system).

A major factor in conducting interviews are the attitudes of the interviewee and interviewer. Remember that your not there to impress the interviewee with your knowledge of computers, so don't talk about megabytes, hard disk sizes and processor speeds.

Q. Who uses computers?

A. Mothers, Fathers, Aunts, Uncles, Brothers, Sisters etc.

Think of your friends and relations, what is their range of computing experience? The people you interview may be experts or complete novices, but they are likely to be apprehensive about the impact that a new computer system will have on their jobs (will I still have a job?, will I have to learn new skills? will I have to change the way I work?). As a systems analyst you have to build trusting relationships with these people in order to get the best information from them. An open, friendly, reassuring attitude is required.

Interview essentials :-

- ***Preparation*** You need to be well prepared, you have to inspire the confidence of the people you interview by demonstrating that you can understand what they do

- and appreciate the problems they face. Some detailed background reading is advisable;
- **Convenience** A further point is that you should interview people at their convenience, at their place of work. Don't expect to get very far if you summon people to your office at short notice;
 - **Dress Code** Respect the dress code of the people you are interviewing, don't expect to inspire confidence if you are interviewing a suit while dressed in tee-shirt, jeans and trainers. Not that dressing smartly will automatically inspire confidence or that it is impossible to inspire confidence if you are dressed casually, just that dressing casually in a semi-formal environment puts you at a disadvantage and you will have to work harder in order to inspire confidence
 - **Body Language** How many of you have felt uncomfortable when a relative stranger sits too close and invades your space or when someone won't look you in the eye (or indeed when someone looks into your eyes too much). Try not to use threatening body language when interviewing, a useful tip is not to sit across a table from someone (this is uncomfortably reminiscent of helping the police with their enquiries). On the other hand don't sit right next to someone since you can get too close and it is difficult then to have eye contact. The ideal position to sit is at right angles to someone, this isn't threatening or too close, allows eye contact but also allows eye contact to be broken

Answer-3

Re-engineering of an organisation

Reengineering the organisation

Reengineering is the radical redesign of organisational processes to achieve major gains in cost, time and provision of services.

Reengineering is appreciated rather than merely computerizing the existing manual procedure. It forces the organisation to start from scratch to redesign itself around its most important processes rather than beginning with its current form and making incremental changes. It assumes that if a company had no existing structure, departments, jobs, rules, or established ways of doing things, reengineering would design the organisation as it should be for future success. The process starts with determining what customers actually want from the organisation and then developing a strategy to provide it.

Once the strategy is in place, strong leadership from top management can create a core team of people to design an organisational system to achieve the strategy. Reengineering is the process of designing the organisation that does not necessarily result in any particular organisational form.

Answer -4

We can use our sample **software evaluation form and free spreadsheet** for **comparing and evaluating your business and ERP applications**. There are no annoying registration forms that require you to submit all your personal details before you get access to it. The **Software Evaluation Form (a spreadsheet)** gives you decision support and takes the subjectivity out of your business software purchase. It ensures that all the evaluation criteria and factors that are important to you in your software selection are compared or weighted to reflect this and to ensure that you choose the "**best of breed software**" that best matches your requirements.

Software Evaluation can become an emotive issue in some companies. Those tasked with finding software or those involved in the **software selection and comparison process** or **software requirements specification** often base their selection on the fact that they like "the look" of a particular software application. Whilst the range of features available in modern business software applications can be overwhelming this approach is clearly wrong. Your approach to the process of selecting software must be measured and scientific using appropriate software evaluation criteria. You must list the features that you require and check to ensure that the applications you are considering can effectively provide them.

Your decision on the selection of business or **enterprise software applications** can seriously affect the way you run your company. Get it wrong and the consequences of failure could be very serious. There have been recorded instances of failed **business** or **ERP software implementations** actually causing companies to go into liquidation.

A major problem in using computer-based programs is the difficulty in finding appropriate software. Good quality software is available, but only to those who know how to identify it. Evaluation of educational software is a fundamental step for teachers and trainers who are in the process of adopting it as learning resource.

The European network for Multimedia Educational Software (ERMES) is one of the European Multimedia Support Networks, mainly composed of European suppliers of Multimedia Educational Software (MES), cooperating to address the global market and to produce high quality multilingual and multicultural learning packages .

A key role in ERMES is played by the MES evaluation that represents both the integration points among the cooperative efforts and the ways to support teachers and trainers to evaluate MES and to find quality products in the process .

ERMES evaluation methodology usually consists of identifying the aspects of the object under evaluation, and then defining the quality indicators related to these aspects. To this purpose the characteristics of multimedia educational software are grouped under four broad evaluation criteria or categories, which refer to, respectively:

"Evaluation of software performance on a given customised processor is an important step in the design space exploration of [embedded system](#) architectures. Such evaluations help system designers in taking early design decisions regarding the hardware architecture most suitable for the target application, but simulation-based performance evaluations, although very accurate, can be prohibitively slower," wrote researchers Soumyajit Dey, Monu Kedia and Anupam Basu.

Their approach consists of a one-time initial simulation run followed by analysis of intermediate level (IR) application code by an evaluation engine. This method, they said, showed that the evaluation engine could estimate the execution cycles of applications or of application tasks on a given customised embedded processor with more than 95 percent accuracy much more quickly than current methods can.

Answer-5

Software QA involves the entire software development PROCESS - monitoring and improving the process, making sure that any agreed-upon standards and procedures are followed, and ensuring that problems are found and dealt with. It is oriented to 'prevention'.

While all projects will benefit from testing, some projects may not require independent test staff to succeed.

Which projects may not need independent test staff? The answer depends on the size and context of the project, the risks, the development methodology, the skill and experience of the developers, and other factors. For instance, if the project is a short-term, small, low risk project, with highly experienced programmers utilizing thorough unit testing or test-first development, then test engineers may not be required for the project to succeed.

In some cases an IT organization may be too small or new to have a testing staff even if the situation calls for it. In these circumstances it may be appropriate to instead use contractors or outsourcing, or adjust the project management and development approach (by switching to more senior developers and agile test-first development, for example). Inexperienced managers sometimes gamble on the success of a project by skipping thorough testing or having programmers do post-development functional testing of their own work, a decidedly high risk gamble.

For non-trivial-size projects or projects with non-trivial risks, a testing staff is usually necessary. As in any business, the use of personnel with specialized skills enhances an organization's ability to be successful in large, complex, or difficult tasks. It allows for both a) deeper and stronger skills and b) the contribution of differing perspectives. For example, programmers typically have the perspective of 'what are the technical issues in making this functionality work?'. A test engineer typically has the perspective of 'what might go wrong with this functionality, and how can we ensure it meets expectations?'. Technical people who can be highly effective in approaching tasks from both of those perspectives are rare, which is why, sooner or later, organizations bring in test specialists.

b).utility of Decision Tree and Decision Tables

A decision tree represents complex decisions in the form of a tree. Though visually it is appealing, it can soon get out of hand when the number and complexity of decisions increase. An example is given below.

First the textual statement is given and then the corresponding decision tree is given:

Rules for electricity billing are as below:

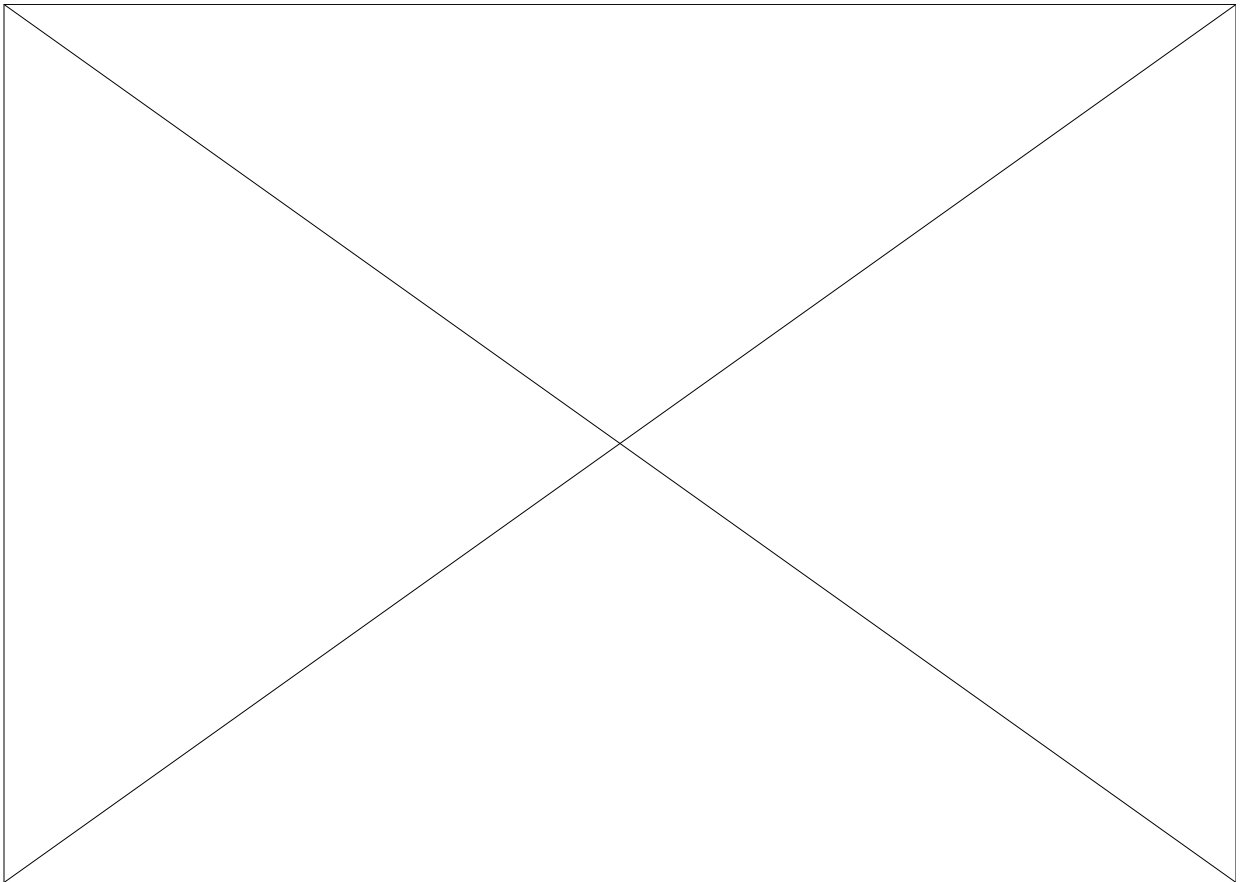
If the meter reading is "OK", calculate on consumption basis(i.e. meter reading)

If the meter reading appears "LOW", then check if the house is occupied

If the house is occupied, calculate on seasonal consumption basis

otherwise calculate on consumption basis

If the meter is damaged, calculate based on maximum possible electricity usage



There are two types of decision tables, binary-valued(yes or no) and multi-valued. An example follows:

ELECTRICITY BILL CALCULATION BASED ON CUSTOMER CLASS

If a customer uses electricity for domestic purposes and if the consumption is less than 300 units per month then bill with minimum monthly charges.

Domestic customers with a consumption of 300 units or more per month

are billed at special rate.
 Non-domestic users are charged double that of domestic users (minimum and special rates are double).

BINARY-VALUED DECISION TABLE

Domestic Customer	Y	Y	N	N
Consumption < 300 units per month	Y	N	Y	N
Minimum rate	Y	N	N	N
Special rate	N	Y	N	N
Double minimum rate	N	N	Y	N
Double special rate	N	N	N	Y

MULTI-VALUED DECISION TABLE

Customer	D	D	N	N
Consumption	≥ 300	<300	≥300	<300
Rate	S	M	2S	2M

Like decision trees, binary-value decision tables can grow large if the number of rules increase. Multi-valued decision tables have an edge. In the above example, if we add a new class of customers, called **Academic**, with the rules:

If the consumption is less than 300 units per month then bill with concessional rates. Otherwise bill with twice the concessional rates. then new tables will look like the following:

BINARY-VALUED DECISION TABLE (three rows and two columns are added to deal with the extra class of customers)

Academic	N	N	N	N	Y	Y
Domestic customer	Y	Y	N	N	N	N
Consumption < 300 units/month	Y	N	Y	N	Y	N
Minimum rate	Y	N	N	N	N	N
Special rate	N	Y	N	N	N	N
Twice minimum rate	N	N	Y	N	N	N

Twice special rate	N	N	N	Y	N	N
Concessional rate	N	N	N	N	Y	N
Twice concessional rate	N	N	N	N	N	Y

MULTI-VALUED DECISION TABLE (only two columns are added to deal with the extra class of customers)

Customer	Domestic	Domestic	Non-domestic	Non-domestic	Academic	Academic
Consumption	≥ 300	<300	≥ 300	<300	≥ 300	<300
Rate	Special	Minimum	Twice special	Twice minimum	Twice concessional	Concessional

c). Edge of Database System over File System

File System - Acts a local, saves in a temp file and synchronizes with the database later part of time (scheduled). File System - Acts a local, saves in a temp file and synchronizes with the database later part of time (scheduled).

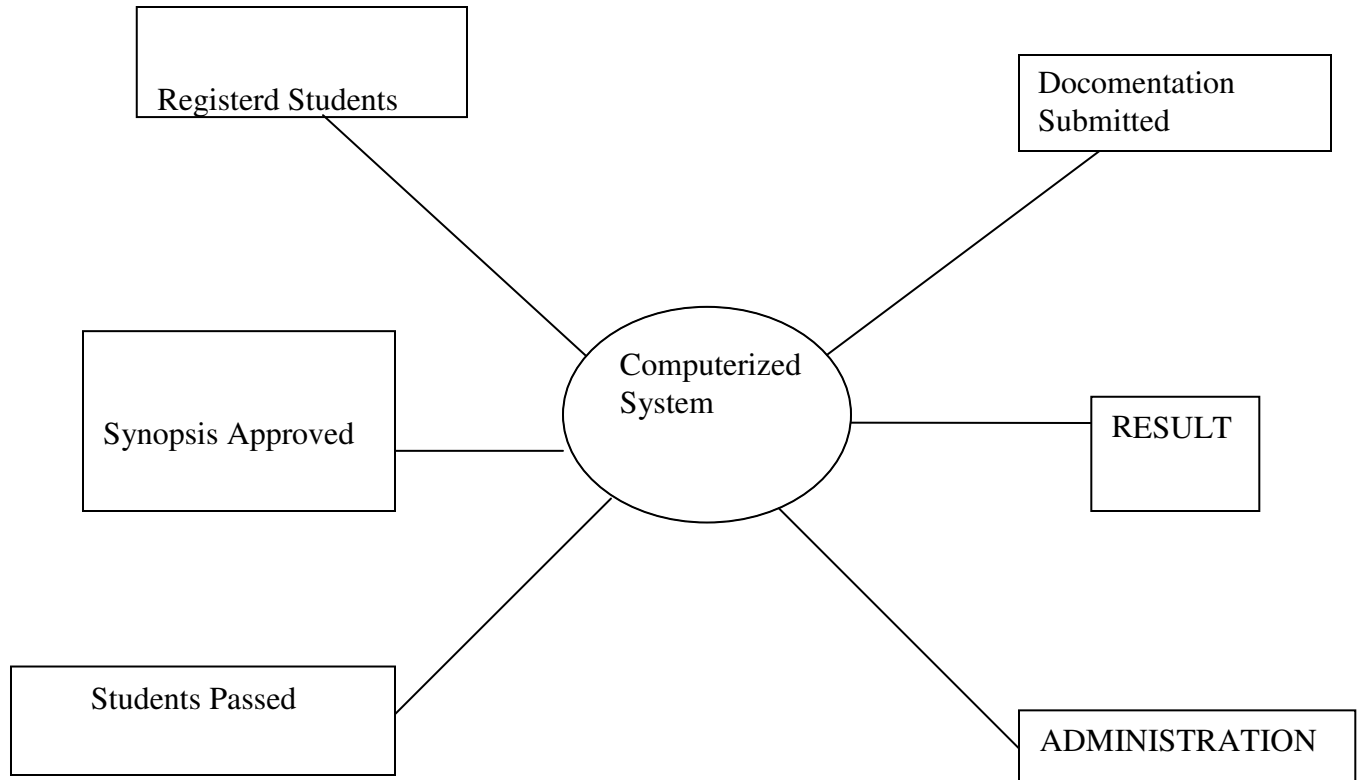
whereas,

DB System - Directly saves into the database, whatever the transaction be(whatever the time taken - this may lead to low performance. A new concept for file management is the concept of a database-based file system. Instead of, or in addition to, hierarchical structured management, files are identified by their characteristics, like type of file, topic, author, or similar [metadata](#).

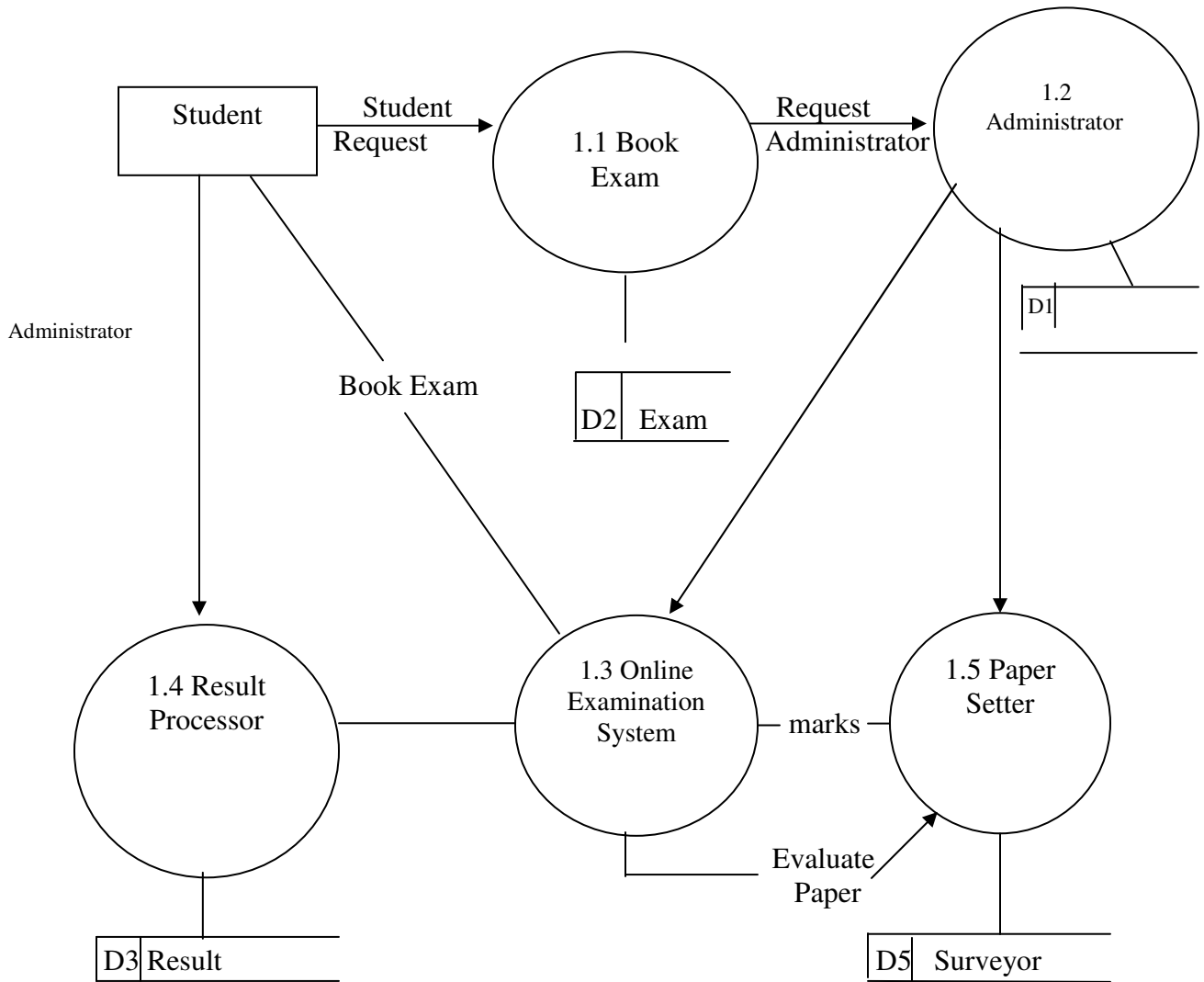
Answer 6:-

IGNOU Computerized System

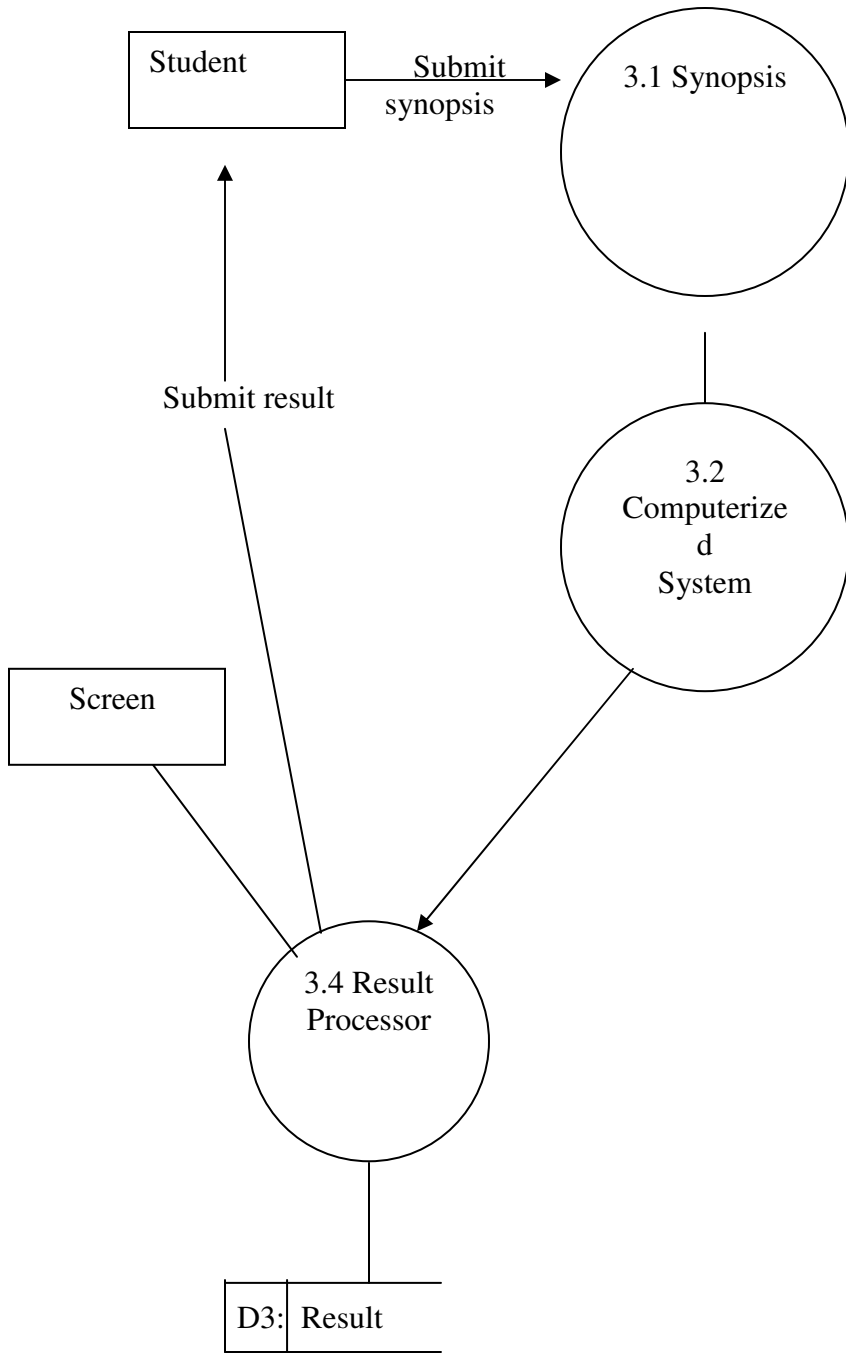
Context level Diagram



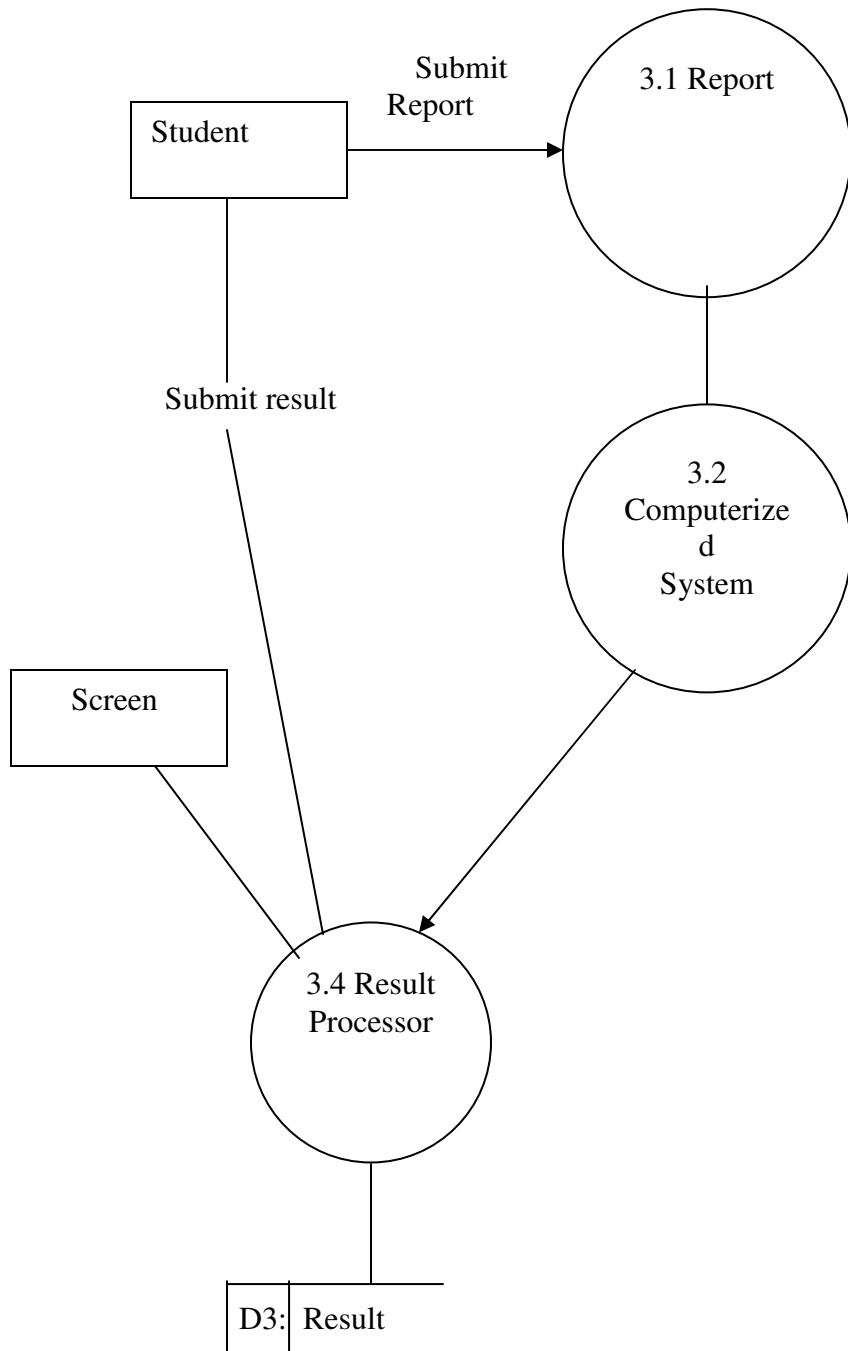
I level DFD For Online Examination System



II Level DFD for Result Process of Synopsis



DFD for Result Process of Report



A Complete Structure

The limited time and resources have restricted us to incorporate, in this project, only the main activities that are performed in news sites, but utmost vehiclee has been taken to make the system efficient and user friendly.

For the optimum use of practical time it is necessary that every session is planned. Planning of this project will include the following things:

- Topic Understanding.
- Modular Break – Up of the System.
- Processor Logic for Each Module.
- Database Requirements.

Topic Understanding:

It is vital that the field of application as introduced in the project may be totally a new field. So as soon as the project was allocated to me, I fully went through the project to identify the requirements of the project.

Modular Break –Up of the System:

- Identify The Various Modules In The System.
- List Them In The Right Hierarchy.
- Identify Their Priority Of Development
- Description Of The Modules:

Modules

Module 1: Password Module

In this module, this website is for multiple users. If a User enters a password and the software checks its validity. If the password is valid then option is given to change the password, otherwise “Invalid

User/Password” message is displayed. There is an option for password recovery, log out, login, new users sign in. The Administrator can also update changes in the site after login.

Module 2: Creating new Entities (Users , Students , Administrator)In this module, whenever a new entity is required to be added the corresponding forms are opened and the database is manipulated to check whether the data is already existing or not. If it already exists, then it prompts that “Entry already existing” and if not than the data is entered with the various validation checks.

Module 3: Modifying / Updating Existing Entities

In this module, whenever an existing entity is required to be modified the corresponding forms are opened and the database is manipulated and the data is fetched. Now the administrator can made the required changes and then accordingly, he updates the data. Again, the checks are followed in case there is any invalid entry.

Module 4: Searching

In this module, whenever an existing entity is required to be searched the corresponding forms are opened and the database is manipulated and the data is fetched. Again, the checks are followed in case there is any invalid entry.

Module 5: Student

This module provides all information about Student and details of Performance and provide all the details of any student information and their result, and Fine expenditures.

Module 6: Appoval/disapproval of Project

This module provides the user facility to maintain the result store records.

Module 7: Result

To automate the complete process of Result maintenance.

Module 8: Validation of Data Entered by the User & Error Handling

In this module, the validity of data entered by the user during the various business processes is checked through various validation checks. For example, there should not be any characters entered in the numeric fields, likewise if there is any error occurs than it should handle that particular error and give the required messages.

Process Logic for Each Module:

In the **first** module, validity of password is checked against a particular user.

In the **second** module, whenever a new entity is entered it should be checked for the duplicate data.

In the **third** and **fourth** module, just like the first module it should have the proper checks for every entity being modified or updated.

In the **fifth** and **sixth** module, again the validation checks are made and the different reports are generated to ease the business processes and decision making.

Validation & Checks

In this project we have used following validation checks.

- While entering the data into the form it will check for the name of the client is properly filled & it should not be null.
- Whenever we enter the data for the student details it will automatically check the details from the database tables and also generate the connection number automatically.
- Similarly in the record tables updated automatically.
- Entered text / number should not exceed the limit (width).
- Almost for all fields we have used the validation for example if name of the fields requires the text type of data then it will check for the string and if the data is numeric then it will check if the number entered is proper numeric or not.