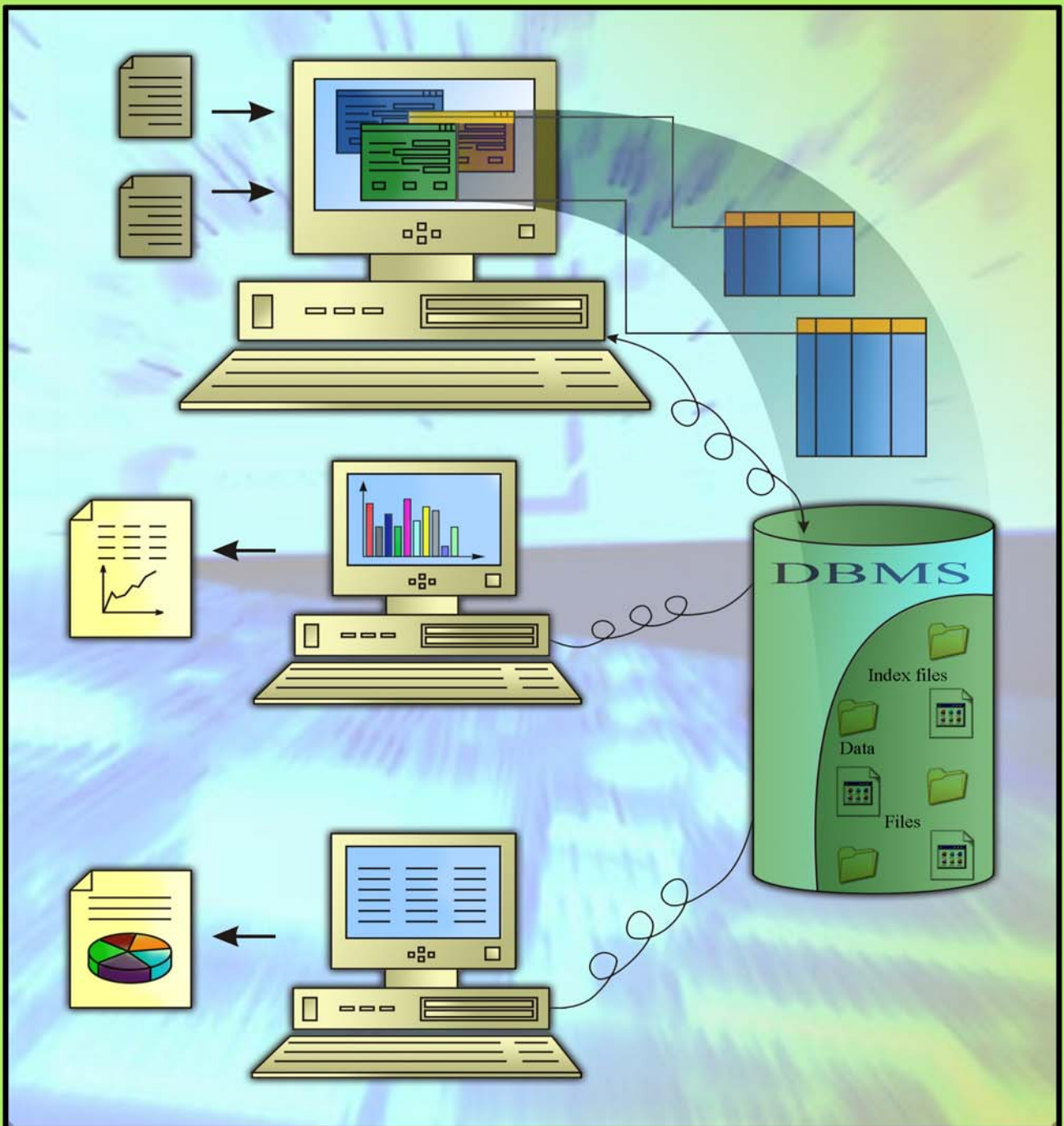




INTRODUCTION TO DATABASE MANAGEMENT SYSTEMS



Block

4

STUDY CENTRE MANAGEMENT SYSTEM: A CASE STUDY

Introduction	3
Introduction to Software	4
Software Development Process: Analysis	14
System Designing	22
Software Development, Testing and Maintenance	39

Programme / Course Design Committee

Prof. Sanjeev K. Aggarwal, IIT, Kanpur
Prof. M. Balakrishnan, IIT, Delhi
Prof. Harish Karnick, IIT, Kanpur
Prof. C. Pandurangan, IIT, Madras
Dr. Om Vikas, Sr. Director,
Ministry of CIT, Delhi
Prof. P. S. Grover, Sr. Consultant,
SOCIS, IGNOU

Faculty of School of Computer and Information Sciences

Shri Shashi Bhushan
Shri Akshay Kumar
Prof. Manohar Lal
Shri V.V. Subrahmanyam
Shri P.Venkata Suresh

Block Preparation Team

Mr Milind Mahajani (Content Editor)
Software Consultant
New Delhi

Prof. (Retd.) A.K.Verma
(Language Editor)
New Delhi

Ms. Suman Madan
Delhi University
New Delhi

Course Coordinator: Shri Akshay Kumar

Block Production Team

Shri T.R. Manoj, Section Officer (Pub.) and Shri H.K. Som, Consultant

Acknowledgement

To Prof. Parvin Sinclair for help in the coordination of this block.

May, 2005

©Indira Gandhi National Open University, 2005

ISBN—

All rights reserved. No part of this work may be reproduced in any form, by mimeograph or any other means, without permission in writing from the Indira Gandhi National Open University.

Further information on the Indira Gandhi National Open University courses may be obtained from the University's office at Maidan Garhi, New Delhi-110068.

Printed and published on behalf of the Indira Gandhi National Open University, New Delhi by the Director, SOCIS.

STUDY CENTRE MANAGEMENT SYSTEM: A CASE STUDY

Section No.	Title	Page Nos.
1.0	Introduction	3
2.0	Introduction to Software	4
2.1	Executive Summary	
2.2	Brief Overview of Study Centre	
2.3	Brief Overview of the Software	
2.3.1	Long-term model for study centre	
2.3.2	Aim of the software development project	
2.3.3	Present working system	
2.3.4	Proposed working	
2.4	Software Life Cycle	
2.4.1	Different Models	
2.4.2	Used Model and its Justification	
3.0	Software Development Process: Analysis	14
3.1	Preliminary Investigation	
3.1.1	Request clarification	
3.1.2	Feasibility Study	
3.1.3	Request Approval	
3.2	Requirements Determination	
3.3	Study of Existing Documentation	
3.4	Completed Forms	
3.5	Reports (Manual and Computerized)	
3.6	Research and Site Visits	
3.7	Observation of Work Area	
3.8	Questionnaires	
3.9	Personal Interview	
4.0	System Designing	22
4.1	Data Flow Diagram for System	
4.2	Entity- Relationship Diagram for the system	
4.3	Data Dictionary	
4.4	A Few Input Designs (Rough layout)	
4.5	A Few Output Designs (Rough layout)	
5.0	Software Development, Testing and Maintenance	39
5.1	Software Development	
5.2	Some Validation Checks	
5.3	Reports Screen Shots	
5.4	System Testing	
5.4.1	Test Plan	
5.4.2	Test Cases	
5.5	Implementation	
5.6	Maintenance Strategy	
5.7	Future Scope of Project	

1.0 INTRODUCTION

In the first two blocks of this course, we have mainly focused on the basic concepts of Database Management System, including Relational Model, ER Model, integrity constraints, normalization, SQL, transactions and many more concepts. These blocks are an attempt to enhance your basic information with some practical inputs.

Block 3 was devoted mainly to the process of an application development without giving much emphasis to systems analysis and design.

This block, Block 4, tries to covers an almost complete application development process along with the required security, based on your familiarity by now with all the

basic DBMS concepts. However, the implementation details in this block are needed to be worked out by you.

Please note that the case study here is small in nature. However, it opens up several development related issues for us. The strength of DBMS as one of the key ingredients of Industry can broadly be visualized from this case study. This case study is an attempt to bring you closer to industry as far as a student projects are concerned. The project/ software development provides you some theoretical and practical aspects of a software project development life cycle.

This block has five sections. Section 1 introduces the case study; Section 2 introduces the software and various models in details and provides an executive summary of the software being developed. Section 3 provides complete details on the Analysis of the proposed software development process. Section 4 lists the design of the software being developed and finally Section 5 provides details on implementation, validation checks and maintenance related issues.

Please note that we have not followed any standard methodology used in the industry. However, this project provides a first good point to move towards the industry standards.

Some of the main learning points of this block are the detailed description of a typical process. However, it should be noted that this is not an ideal process in any way. The industry follows many different standards, which you may study in your 4th Semester.

The tools used here are VB and SQL Server and only a sample code of VB is shown.

Ideally this case study is suited for finding errors, process related problems. But you get a detailed view of how DBMS forms part of an overall system.

Some suggestions about this case study:

- This case study does not attempt industry oriented process.
- There may be analysis/ design/ implementation errors. Discuss these in the group counselling sessions.
- You can improve and implement in software of your choice.

OBJECTIVES

After going through the case study, you should be able to:

- define basic software development issues;
- make simple table and graphical designs;
- describe models like DFDs/ ERDs, and
- revise your DBMS and System Analysis Design Concepts.

2.0 INTRODUCTION TO SOFTWARE

In this section we will discuss some of the basic thoughts about the software that is being developed. The section starts with an executive summary and provides overview of the study centre and the existing system.

2.1 Executive Summary

The software **Study Centre Management System** for IGNOU's Study Centre is a well-organised software developed to provide the user to get relief off from the manual system as it is basically automatized version of management. The software with its capabilities has ability to computerize the whole system. The system has features that are vital to the user such as Report Generation for the Students Marks, fees, attendance, etc. Manually creating a report takes hours but the built-in

functionality in the software allows user to create different reports in a few clicks and types. Of course it requires data entry to computer, yet this data if structured properly can be reused for many different reports.

Some of the key features of this system are:

Auto-complete feature allows user to enter the data in a quicker and faster way with few letter types as well as enhanced validation checks, and maintains the data integrity and consistency.

Backup and Recovery System allows user to take the backup copies of the database and recovery facility and allows recovering the system in case of system failure.

Color Themes and Background Pictures allow user to totally customize the look and feel of the software according to their demand in the same way as there are color themes in Windows.

Help to the operator is provided in terms of Tips Box and user manual so that user could get immediately help support from the software. Also the on-line help, i.e., mini-help on status bar, allows user to quickly know, the use of various forms.

The software has enough facilities for which it can be implemented and used. It provides various advance functionalities, which attracts user to use it. Its look is very attractive which makes the user comfortable to work on it.

That's all of the software which we have built. Now let us have a very brief diagrammatic representation of role of study centre in IGNOU

The figure describes the basic process of delivery of IGNOU courses to the students. The role of study centre is primarily in the student interface input, where they come in contact with the centre of their choice. All of you by now must be familiar with the system.

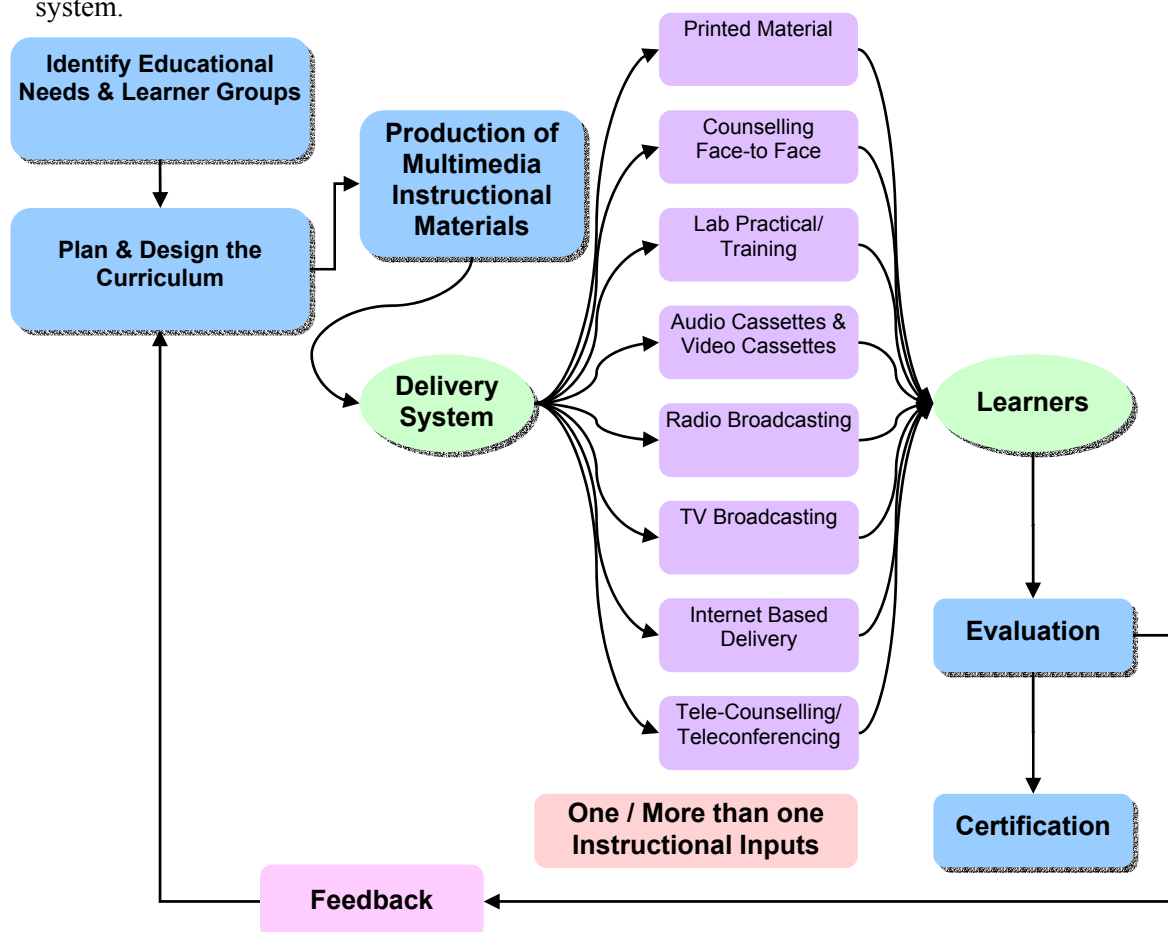


Figure 1: Diagrammatic representation of the instructional system of IGNOU

2.2 Brief Overview of Study Centre

In this system we are stressing more on Study centres because software is to be developed for the Administrative Management of Study Centres. We will discuss in detail its aim, goals and visions in later sections. At present let us have a brief idea of the Study Centres.

The study centres are the student support centres of IGNOU and are controlled by the Regional Centres. The students are serviced by effective networks of 48 Regional Centres and 1068 Study Centres located all over the country. With this large cross network, IGNOU endeavours to provide seamless access to higher education using innovative technologies and methodologies.

Now, let us see the various study centres categorised according to their working:

- Regular Study Centres:
- Programme Study Centres
- Recognised Study Centres
- Sub Study Centres
- Special Study Centres
- Study Centres (North Bihar Project)
- CWDL Down Link Centres
- DLF (Distance Learning Facilitator)
- Army Recognised Centres
- Navy Recognised Study Centres.

Now, let us point out the various functionalities and dimensions of a Study Centre.

Some of the major functions are as follows:

- Appointment of Local Tutors / Counsellors.
- Academic Support through face-to-face services
- Conduct of Practical / Hands on Experience.
- Library Facilities
- Teaching through Media
- Individual Teaching / Counselling
- Evaluation of TMAs
- Maintenance of Learner Records
- Venue for Peer-Group Interaction
- Conduction of Term End Examinations
- Co-ordination of Staff Members.

The main entities of a Study Centre are as follows:

- **Coordinator of study centre:** S/he is one of the executives who will look after the working of the Study Centre and its functionality.
- **Counsellors/ Teachers or Faculty Members:** The teachers are also an important part of the SC as they will use the proposed software and would evaluate students' performance.
- **Different Programmes and Courses:** Programmes and their courses are also an important part of SC. As there are a large number of courses, we need proper handling, so we need the proposed system.
- **Students:** They are also an important part of a Study Centre.

Programmes/ Courses on offer:

A study centre may offer many UG and PG Programmes/ courses as offered by IGNOU.

2.3 Brief overview of The Software

In this section we like to provide a brief information on our software which is named as Study Centre Management System. As the name suggests, the software is for the administration of the Study Centre. We hope that you know about the study centre functioning. So let us have a look at the functionalities, financial planning, courses offered, and administrative support provided by the study centre, which we are going to computerize through this software project. Now let us study each aspect in more detail.

Please note that our attempt is not to replicate the functioning of a single type of study centre of IGNOU. We have added some features, which may be desirable in many different study centres being considered for this software development project.

The main functionalities of our project are the maintenance of all records of Courses, Teachers, Faculty and Students. These are:

- Study Centre Details
- Teacher Details
- Students Affiliated and their Marks, Fees, Attendance Records
- Course Structure
- Schedules Maintenance
- Assignment Submission
- Study Centre Allotment Facility
- HTML Reports Creation of Marks, Fees, Attendance, etc.
- Query about all the Resources available in Study Centre.

A detailed cost benefit analysis may be difficult to work out as many of the benefits are more on the side of providing better service to the students. In this matter this system would definitely be more advantageous. Thus, a study centre person's work now can be focused on intellectual abilities rather than wasting time on doing trivial work. This provides great benefits to the students and staff.

Figure 2 shows a very brief plan for the development of the software project. It is a Gantt chart showing various important activities of the Life cycle for the software project development.

2.3.1 Long-term model for study centre

The student at an Open University undertakes a methodology of instruction which is different from that of conventional universities. The University follows a multimedia approach for instruction. It comprises:

- Self - instructional written material
- Audio-Visual Material Aids
- Counselling Sessions at the Study Centres
- Live one-way video two-way audio Teleconferences
- Practical / Project Work at the Study Centres
- Repeated telecast on Gyan Darshan Education Channel
- Interactive Radio-Counselling through Gyan Vani the Radio Channel.

A student primarily visits the Study Centre for watching audio-video programmes, counselling sessions, live teleconferences (as they are on specific channel at present not widely telecast by cable operators), and practical work. However, in the long run it is expected that IGNOU would reach the home of the students through its Teleconferencing, Radio sessions, and using EDUSAT. Then the role of the Study

Centre in such cases may primarily be to provide on-line Administrative support and practical work.

Figure 2: Gantt Chart of Project

2.3.2 Aim of the Software Development Project

Whenever a Project is made its Scope is determined – whether it will last for a long period of time or short. So, in this section we are going to explain the aim of our project keeping in mind that our software will have adequate life.

The aim of our project can be categorised by the computerization of the following tasks:

- **Record Maintenance of Study Centre Details:** Maintenance of Study Centre Details means that we are going to keep the details of the Study Centre in which our project is currently implemented.
- **Record Maintenance of Teacher Details, Study Centre Resources:** In this section we are going to keep the teacher's complete bio-data which will enable students to find out about their teacher's capabilities, mastery, experiences and knowledge.
- **Record Maintenance of Students Affiliated:** In this we are going to keep complete details of the students of the Study Centre. Using these details students will be frequently asked for suggestions.
- **Programme and its Course Structure:** Complete details about programme structure, its fees, eligibility, duration and many other details which would enable students to select the programme of their choice.
- **Maintenance of Fees, Marks and Attendance details of Students:** Although the fee at present is not being deposited at the study centre, in the long term it may be the case, if every data transfer is sound and secure.

- **Schedule Creation for Classes**
- **FAQ (Frequently Asked Questions) Facility:** This section will enable students to find out answers to their queries. It would contain Frequently Asked Questions with their answers. Using this facility, students can find out the solution of their most common problems during admission and study phase.
- **Report Generation of all the Resources available in Study Centre and Student Support Services like marks, fees, attendance report.**

Some of the standard features of this software project are as follows:

- *Backup/Recovery System.*
- *Automatic creation of Database on a new System.*
- *Multi-user Environment Facility: Support for concurrent transactions.*
- *Password Facility for Administrator and Teachers: Support for security.*

2.3.3 Present working system

At present our system has lots of manual work which needs to be computerized. So, lots of paper work is required. In order to minimize this we need to computerize all those departments which require record maintenance. The different points which illustrate the present working of the System are:

- All the different administrative department works require paper work which includes the record maintenance of current Study Centre, Students Affiliated, Teacher Details or Faculty Details.
- Fees of the students include submission of the form which is duly filled by the student (This feature is not there in IGNOU system but we are assuming it from the long-term plan point of view).
- Records of Marks of Assignment, Theory and Practical are maintained on the paper.
- No FAQ facility, so students who have some confusions and problems regarding courses structure and fees need to talk to specific functionaries who are normally not available on weekdays at study centres as most of the Study Centres operate on Saturdays and Sundays only.
- They are not able to find out their teacher qualifications and experiences.

2.3.4 Proposed working

After the implementation of this project, there would be lots of benefits as there would be less manual work, more computerized working, resulting in faster resolution of students' queries.

It would provide all the goals and aims which we have discussed in the earlier sections. So, making the present system faster, quicker, more active and user friendly to the students, teachers and study centre staff members. The system will become more responsive and fast.

2.4 Software life cycle

In this sub-section, we are going to list the procedure we are going to use in order to build our project. The details of this methodology are defined in the System Analysis Courses and Software Engineering Courses. The term Project Methodology suggests the processes, structures, method or style for building our Project.

Software Life Cycle consists of the set of activities which the analysts, designers and users carry on to develop an information system. Let us have a brief overview of the stages. It consists of following stages:

Preliminary investigation: It consists of request clarification, feasibility study

(software being technical, economical and operational) and approval of request.

Determination of requirements: It consists of the following questions which are needed to be answered:

- What is being done?
- How is it being done?
- How frequently does it occur?
- How great is the volume of transactions?
- How well is the task being performed?
- Does a problem exist?
- If problems exist, the level of seriousness of the problem.
- If problems exist, the underlying cause.

In order to answer these questions the System Analyst will do detailed investigation by study and observation of data collected using fact finding techniques like interviews, questionnaires, etc.

Development of Prototype System: It consists of developing models of the requested task, object oriented methods for easier programming and 4th Generation Language Technologies.

Design of System: It consists of elements of the task to be designed, the design of the report to be printed, the design of input to be provided and design of files.

Development of Software: It consists of the language used, the file description and the user friendliness of the software.

System Testing: It consists of different types of testing of the software including unit testing, system testing and validation testing and user acceptance.

System Implementation: It consists of the training needed by the user, conversion methods, system review and maintenance.

System Maintenance: It requires trained, skilled persons who have good knowledge of system working and of handling the system in case of any failures or adaptations.

But what process Model do we follow to develop this Project? Before answering this question let us recapitulate the different process Models, their advantages and disadvantages and then select one of the most suitable models for our domain.

2.4.1 Different models

Waterfall Model

It consists of a linear set of distinct phases including requirement analysis, specification, design, coding, testing and implementation. It can be represented as in *Figure 3*. Please note the use of two terms, verification and validation, in the figure. Verification is defined as the question “Are we building the product right?” Validation is defined as the question “Are we building the right product?”

Features of the waterfall model:

- Systematic and linear approach towards software development.
- Each phase is distinct.
- Design and implementation phase only after analysis is over.
- Proper feedback, to minimize the rework.

Drawbacks of the waterfall model:

- Difficult for the customer to state all the requirements in advance.
- Difficult to estimate the resources, with limited information.
- Actual feedback is always after the system is delivered. Thus, it is expensive to make changes during the later stages of software development.
- Changes are not anticipated.

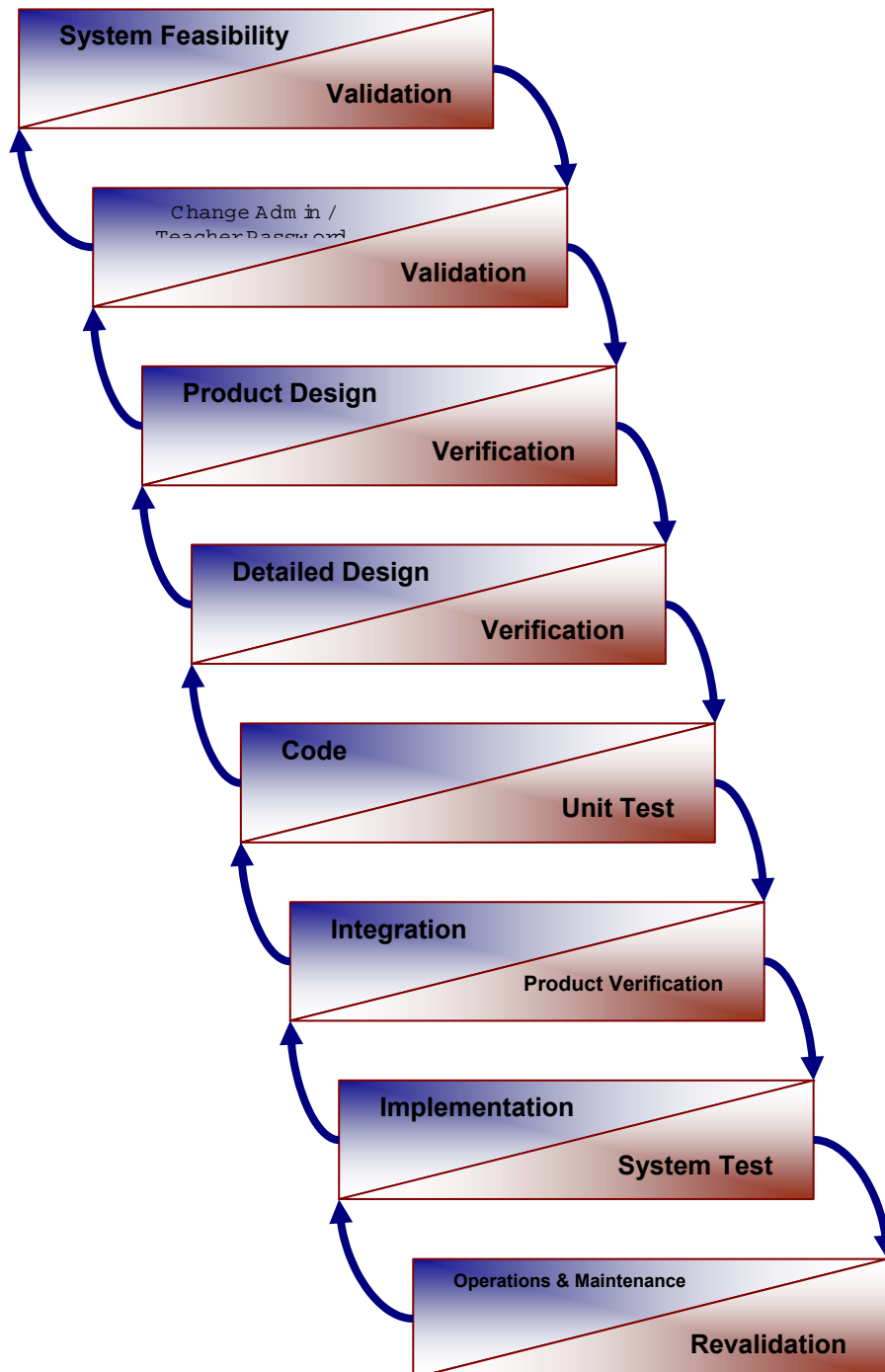


Figure 3: The Waterfall Model

Rapid Application Development Model

RAD is more like a set of parallel waterfalls where the system can be decomposed into largely independent subsystems. It includes a broad array of software tools that automatically generate source code as per the developer's specification.

Prototype Model

It consists of distinct phases including its prototype, for example, demonstration that enables the customer to get a feel of the system. It can be represented as in *Figure 4*.

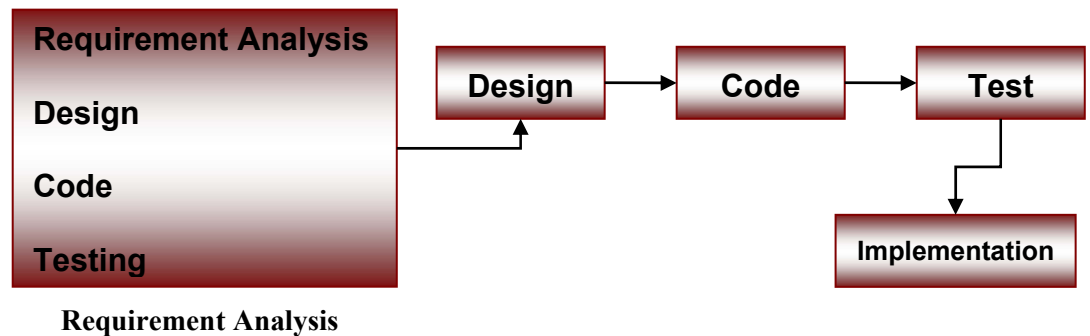


Figure 4: The Prototype Model

Features of the prototype model:

- Reduced functionality.
- Begins with requirements gathering.
- Used in large systems.
- Makes use of existing programs.
- Reduces the risk of project failure.

Drawbacks of the prototype model:

- Customers feel that the cost incurred or provided is too high for the model.
- Developer might use inefficient algorithm during prototype development, which may not ultimately be changed due to customer's non-cooperation.

Iterative Enhancement Model

It consists of distinct phases in which the steps after the designing steps are repeated if the customer is not satisfied. The demerit of this model is that the iteration may never end thus the user can never get the “final” product. It can be represented as *Figure 5*.

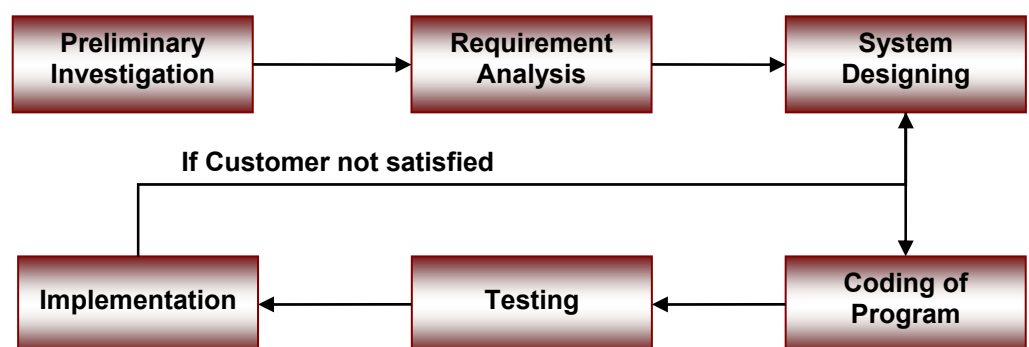


Figure 5: Iterative Enhancement Model

Spiral Model

It consists of distinct phases including both the prototype model and the classic life cycle model. It includes six major phases which are customer communication activities named planning, construction risk analysis, engineering and release customer evaluation. This model can be represented as *Figure 6*.

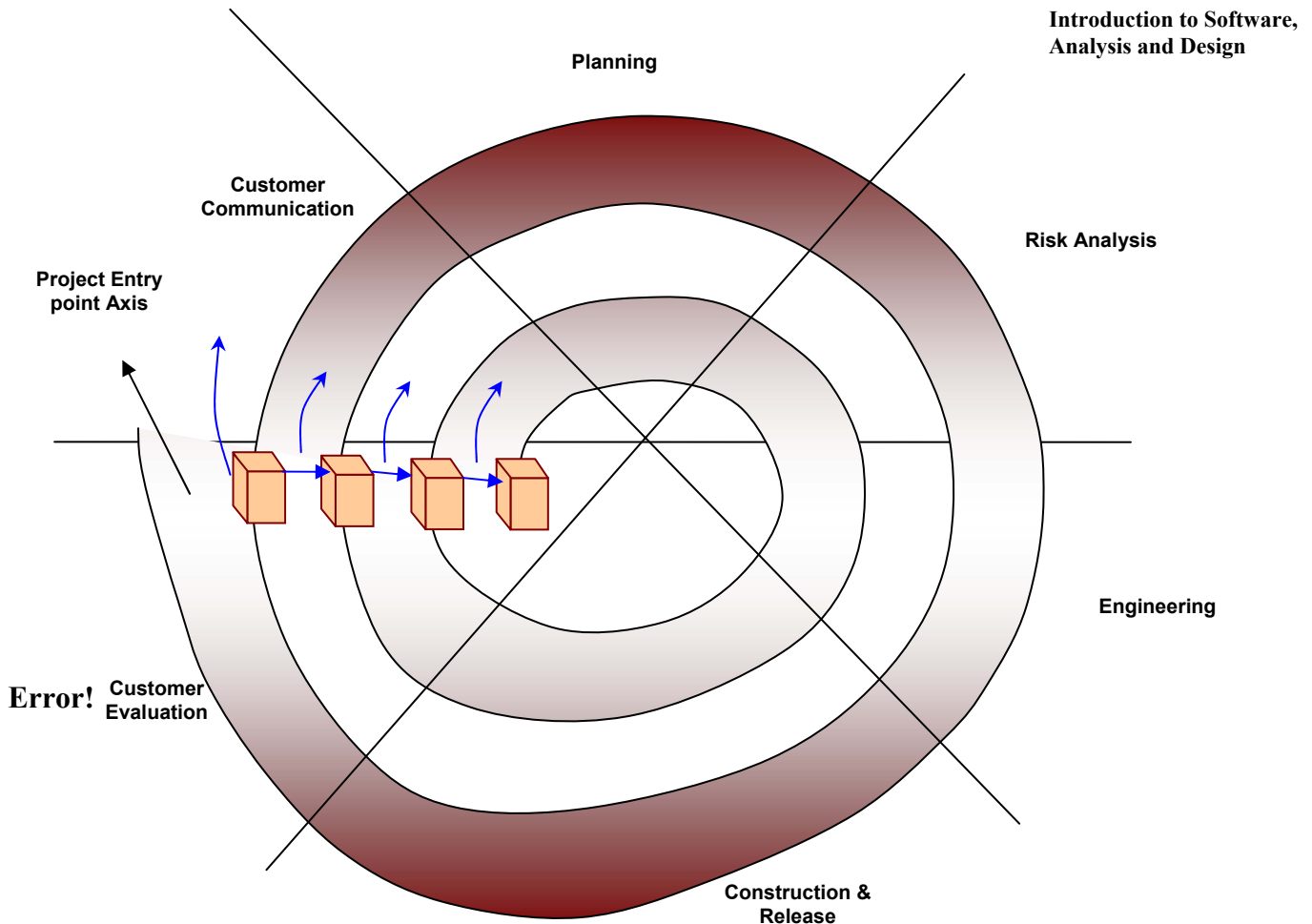


Figure 6: The Spiral Model

Features of the Spiral model:

- Risk assessment is involved at every activity.
- Most realistic approach to development for large systems.

Drawbacks of the Spiral model:

- Risk assessment involves great expertise.
- Comparatively a new approach.

Fourth Generation Techniques

It covers a broad array of software tools that have one thing in common. They are considered to be the high level languages, as they serve the users in a very friendly way. Based upon the specifications given by the customer, the tools available automatically generate the source code. 4GT's include Non-procedural languages for database query, report generators, menu generation, screen generation, code generation, graphics ability, and spreadsheet capability and even more. These tools exist but for very specific application domains.

Features of the 4GT's model:

- Non-procedural language.
- Build application fourth generation language (4GL's).
- Tools for making report and menu and screen generation of the required program.

- Faster query handling.
- Graphics ability and spreadsheet ability in the required program.
- Conversion of requirements gathering to implementation using fourth generation language (4GL's).
- Good for business information system.

Drawbacks of the 4GT's model:

- Customer is not assured of what is required in advance.
- Customer might not be able to specify the facts properly.
- Not good for big systems.
- Poor maintainability, quality, customer acceptance.

2.4.2 Used model and its justification

Spiral Model

As we know, in the **waterfall model** we cannot perform the next step without doing the previous steps because each step of the waterfall model is **dependent on each other**. However, the spiral model allows use of the features of prototype and incremental model which allow us to divide the project into increments. So we can increment the on-going project after some time, i.e., we can add some new features to the project or we can add some new division or some new functionality in our project. So the chances of failure of the project by using this model is minimal because some result is obtained after the completion of the 1st increment which is generally not possible in the waterfall model. If we want to change some feature or functionality or add some functionality, it can be done by this model. In the waterfall model we cannot do this because repetitive changing may cause the process to restart again and again.

The next best feature of this model is that it is an ongoing process for the lifetime of the project. This means that the project cannot die till the user is satisfied.

3.0 SOFTWARE DEVELOPMENT PROCESS : ANALYSIS

Building an information system or software system is not merely putting the computer instructions together but it involves uses of software engineering process methods and tools. Until software system is implemented and delivered to a customer, and even after that, the system undergoes gradual development and evolution.

We have already discussed the stages related to our project in the previous sections. Let us first discuss the type of this system. The system which is being developed by us involves transaction processing and Management decision system. The system includes processing data. **Transaction Processing System** aims at improving the day to day business activities operations on which the organisation depends. Some of the transactions for the proposed system may involve online updating of fee records, marks, attendance, etc. The sole objective behind the system is to increase the efficiency, speed and accuracy of processing on the voluminous data. So, all these facilities result in greater efficiency and quicker response time of the system.

The users of the system can be categorized into administrator, teachers and students. The following *Figure7* shows the type of users and their rights. The classifications of users of a system are best explained by the Neumann Haddass Model which illustrates the various levels of management which looks after the planning and controls of the system. Let us have a brief overview first of the Neumann Haddass model and then of its application to this system. *Figure 7* shows various users of the system.

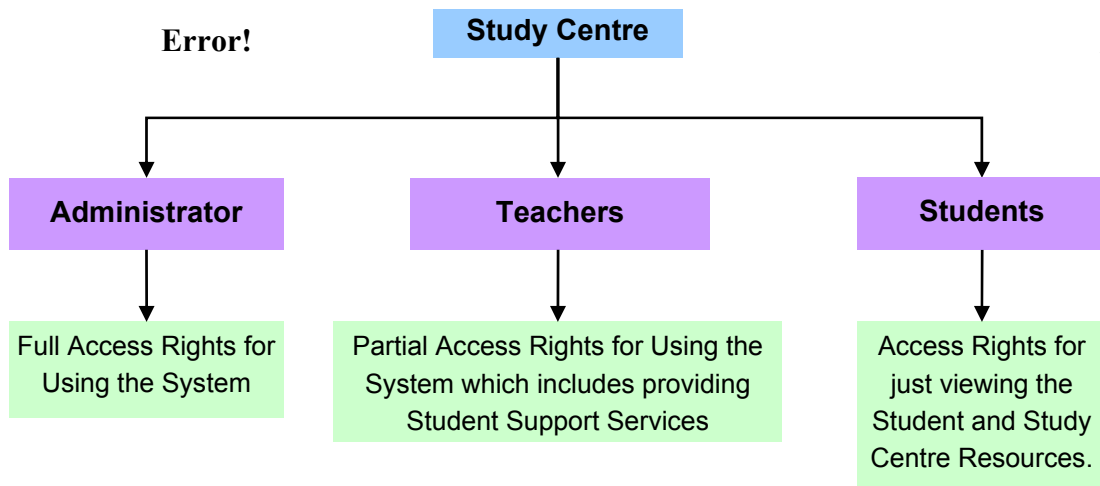


Figure 7: Users of the Proposed System

Neumann Haddass Model

It states that the organisational chart can be divided into three levels of management who look after three different types of planning and control. It is represented by the tree or triangular interdependent systems at different levels. Refer to Figure 8.

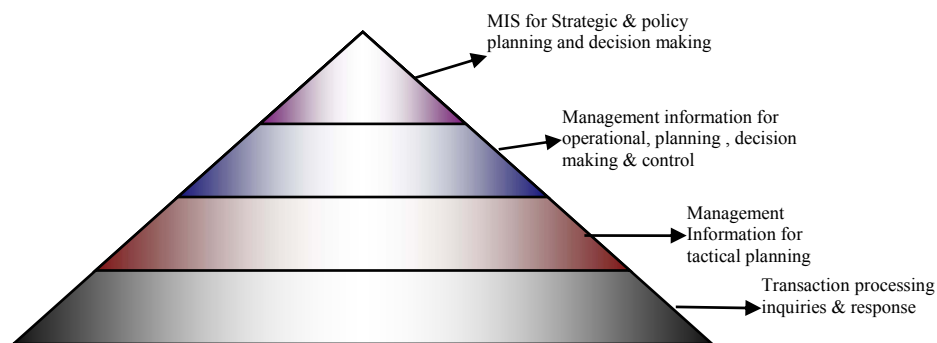


Figure 8: Neumann Haddass Model

The top-level management looks after strategic planning which determines long-term goals and the purpose of the organisation. It also identifies the resource needed to achieve the goal, the government rules, etc.

The second one is the middle level management which monitors the use of resources identified by the top level management. They look after the people, money, equipments, etc. to achieve the goal. They are very important and the progress and speed of working depends on them, not completely but substantially.

The third one is the lower level management. Its working is similar to that of middle level but at a lower level. They look after the day-to-day working and progress of the individual.

In our case, the Top level Management is under the administrator who is either the Study Centre Coordinator or Principal. Next, in the middle are the teachers, or faculty members. However, they sometimes work at lower levels also. The students come at the bottom level or lower level. They can avail of the facilities provided by the system.

Now, what is the use of this analysis phase? It allows you to examine the eight basic characteristics of the system which are:

1. **Goal:** Each system is developed for a reason. It has some purpose for its existence.

2. **Inputs:** The inputs to the system are the resources like manpower, capital, materials, equipment, etc.
3. **Output:** These follow from the system. They are usually goods (may be furniture from a factory or leather handbags from a workshop) or services (legal consultation from a lawyer's office or a project feasibility report from a management consultancy firm).
4. **Boundaries:** Each system has boundaries. In other words, each system has a definite scope. We may wish to include everything in one system but that is practically impossible. Each system has some definable limits. Remember the user's wish list? Some items in the wish list may be outside the scope of the system.
5. **Environment:** Each system exists in an environment that can influence the system in a number of ways, such as making it more or less difficult to perform.
6. **Components:** The components of a system transform the inputs into outputs and the linking together of the components is accomplished by interrelations.
7. **Interrelations:** These describe how resources pass from one system component to another.
8. **Constraints:** These represent the environmental, social, technical or behavioral factors that typically limit system performance.

You need to find out all about the system's environment, boundaries, inputs, outputs, goals, components, resources and interrelations.

Let us now list the important characteristics of the system being developed by us:

Goal	– Printing of Schedules, Fees, Marks, Attendance and all Miscellaneous reports that will help better planning at study centre for student support.
Inputs	– Attendance, Fees, Marks, Schedules details and all Miscellaneous details
Outputs	– Schedule Creation, Student Support Services
Boundaries	– Study Centre Data which includes its Resources and Students Record.
Environment	– IGNOU Terms and Conditions as well as Study Centre Rules and Regulations.
Components	– Various modules that we will develop.
Interrelations	– Computer programs
Constraints	– Data available to administrator and teachers.

3.1 Preliminary Investigation

After having a brief idea of the project, let us undertake the preliminary investigation to ascertain the feasibility of the project. Ideally this stage of analysis should be performed prior to project approval. The three most important stages of preliminary investigation are:

1. Request Clarification
2. Feasibility Study
3. Approval of Request.

3.1.1 Request Clarification

This step is a one of the component of preliminary investigation stages. Many requests from employees and users in the organisation are not clearly stated. So, it should be clear in advance what the originator wants. Problem clarification is a difficult task

because requester is making a request but how to state it or where actually the problem is or what is the problem is not clearly stated. For instance in this project the Coordinator has made the request for adding web-site facility for the study centre but what would be the contents was not stated. Thus, the cost estimation for website could not be done.

3.1.2 Feasibility study

It is an important part of the **Preliminary Investigation** because only feasible projects go to development stages. Let us do a very basic feasibility study for the current project.

1. **Technical feasibility:** Technical feasibility raises questions like, is it possible that the work can be done with the current equipment, software technology and available personnel? And if new technology is required what is the possibility that it can be developed?

In case of our project, the software which we have built up fully supports current Windows OS but it lacks the support for other OS environment like Apple Mac, Unix and Linux. Next, it is not dependent on the number of users so it can handle a very large number of user environments (in principle, but does not happen in practice). Next, the Support for Hardware, it has full support for new hardware, so no hardware compatibilities issues arise as it requires minimum configuration but only non-Mac environment.

Minimum Hardware Support Required

Computer	Intel® or compatible Pentium 166 MHz or higher.
Memory (RAM) ⁱ	64 MB minimum on Windows 2000, 32 MB minimum on all other operating systems 128 MB or more recommended
Hard disk space ⁱⁱ	Minimum Space for Software 10 MB
Monitor	VGA or higher resolution 800×600 or higher resolution required for the Software Project
Pointing device	Microsoft Mouse or compatible
CD-ROM drive	Required for Installation

- (i) Additional memory may be required, depending on operating system requirements.
- (ii) Actual requirements will vary based on your system configuration and the applications and features you choose to install as well as SQL Server Space Allocation.

2. **Economic Feasibility:** It deals with economical impacts of the system on the environment it is used, i.e., benefits in creating the system.

In case of our project we are assuming an economically feasible solution.

3. **Operational Feasibility:** It deals with the user friendliness of the system, i.e., will the system be used if it is developed and implemented? or will there be resistance from the users?

In case of our Project we have done the Operational Feasibility Study with Centre Coordination and Incharge about the usage of the project, and regarding user friendliness we have tried our best to make the software highly user friendly so that a person having only a little knowledge of English can handle it. By the way we have also built on-line as well as special help programs which help in training the user. Also one of the interesting parts of the Project is **Tips of the Day** which

gives some special Tips to the user for proper functioning.

4. **Time Feasibility:** In this type of feasibility study, we examine whether our proposed project can be completed in the specified time frame or not. In our case, our project is finished in the targeted time-frame. So, it is feasible regarding time scope.
5. **Legal Feasibility:** This type of feasibility evaluates whether our project breaks any law or not. According to our analysis, our project doesn't break any laws. So, it is legally feasible too.

3.1.3 Request approval

Not all the requested projects are desirable or feasible. And only those which are desirable or feasible should be put into the schedule. In some of the cases the development can start immediately but in some of the other cases, the staff members are busy on their ongoing projects. Many business organisations develop information systems as carefully as they plan for new products, or new manufacturing programs. After a request is approved its cost, priority, completion time and personnel requirements are estimated.

For example, in our case the management who will be investing the money might feel that there is no requirement of such a system, so unless they approve it work cannot start. However, this project does not have such constraints, as it is just a sample project.

3.2 Requirements Determination

Any software project can succeed only if it has been fully analysed in this stage of Requirements Determination, since Requirement Determination is the heart of the systems analysis, aimed at acquiring a detailed description of all important areas of business that is under investigation. Analysts working closely with employees and managers must study the complete business process for the designers of the system.

In this stage, the system analyst will talk to a variety of people in detail and will take their ideas and opinions. We use fact-finding techniques like Questionnaires to collect the information from people who cannot be interviewed individually and interviews with special appointments for discussion on the project. The detailed investigation includes:

1. Study of manuals and reports.
2. Actual observations of work activities.
3. Collection of sample forms and documents.

In our case we have several interviews with Mr. Goel, centre incharge, about the project discussions. He helped us regarding this project on what to build in it, what are user requirements, etc. We also got ideas from the students who may be users of the system.

Now let us have a brief overview of the fact-finding techniques of the project which we used.

The most commonly used techniques of fact finding by us are as follows:

- Existing documentation, forms, file and records
- Research and site visits
- Observation of the work environment
- Questionnaires
- Interviews and group work sessions.

All these techniques helped us a very great deal in finding out the information related to the project. Let us first have a glance at all these techniques.

3.3 Study of Existing Documentation

You may feel that getting existing information may not be right since you are there to create a new computerized system. So why bother about what happens at present? Think back to the earlier session. What did we say? To build a computer-based system, it is essential to first clearly understand what happens right now. You will be able to build a system only after a correct understanding of the current system. You may even improve it.

So, existing information is absolutely essential. It will give you a clear picture of what actually happens in the department or organisation being studied.

For each document, it is important to know its use in the system. To understand this, it is useful to prepare the document-department routing grid.

So talking about getting existing information, what are the advantages of this technique?

- You don't have to disturb any person while they are working. You can take the files to an empty room and study them at your leisure.
- There is no time constraint. This does not mean that you take 3 months just to study the existing documentation. What it means is that there is no time deadline like an interview. If your interview carries on for 2-3 hours, you are going to have a very unhappy user. Whereas if you spend 2-3 hours studying the document, you will have bothered no one.
- You will know exactly what happens and not what the user wants you to know. You will be able to study the processes free from the bias and personal viewpoint of the user. You will be unaffected by the users' feelings or personal office politics and get an impartial view of the system.

We are talking about getting existing documentation but what are the documents that you should seek? Some of the documents looked into by us were:

After this you can collect documents like:

- Class schedules, faculty schedules.
- Minutes of important meetings.
- Accounting records of the students
- Assignments submission of return records
- Making system and results communication
- Attendance record, student batches record, infrastructure etc.

Let us look at some of the above documents more closely and see how they can help an analyst.

3.4 Completed Forms

The most important documents are the **completed forms** representing actual transactions. Remember that blank forms should not be accepted. This is emphasized because commonly, users prefer to tear out a blank sheet from their file or register when they are asked for a form. A blank form will tell you nothing about the kind of information that is actually filled in the columns. The form design may be obsolete and the users may have devised their own forms within the framework of the old one. The heading to the column may be totally misleading and the user may actually be entering something different out there.

We have said that we should study only completed forms. All right, but how many should we study? Just 1 or 10 or 100 or 1000? What is the correct sample to study?

There exists an empirical formula by which you can find the sample size to be certain

of the quality of data. In our case a sample size of 25-50 forms is ok.

Then comes the question – how do we choose the sample?

There are 2 ways.

The first is **randomization**, i.e., you pick up any 50 documents and use them as a sample.

The other approach is **stratification**, that is, a systematic method for choosing the sample. Let us take an example. Suppose the total number of student documents we have is 5,000. We have to choose 50 documents in our sample. So we choose every 100th document which is $5000/50$.

3.5 Reports (Manual and Computerized)

All **manual and computerized reports** must be collected. This may sound easier than it actually is because many users devise their own reports for follow-up or for statistical analysis. These reports are normally prepared by hand or word-processed from information available in the system. These are considered by the users as ‘their’ property and normally not shown to the analyst. Remember in our last section we talked about some notebook or register maintained by the user privately? Well, these are reports. Therefore, when you ask for all reports, make sure that you really get all the reports. The reports prepared manually by the user are actually the most important because they are the ones being used regularly. You may wish to modify the existing computer reports to absorb the manual report.

If the organisation has **standard operating procedures (SOPs)** in place, it makes life easier for the analyst. The analyst will know how things ought to be done and then, when the procedure is being reviewed with user, the gaps or inaccuracies will be easily identified. In case of missing SOPs, the analyst will have to write the procedures himself after identifying all responsible parties and determining the tasks appropriate to each.

It is very important to review the SOPs very carefully to verify that they are complete and accurate. Any mismatch between the actual procedures carried out and the SOPs should be identified and resolved. You may find a number of duplicate information flowing from one department to the other or a particular procedure being eliminated in actual practice or different information recorded on the same form.

Let us now see how it helped us in our project.

“You may find that a different type of form was used to record the details of Students Marks, Fees and Attendance Schedules, etc. We questioned about the formats, the conclusion may well be that no one really knew why those formats were used but they were doing it because it was always done that way.”

Therefore, the point to be made is that collection of existing documentation is important because it clarifies your understanding of the current system. It also throws up gaps, missing parts and other questions about the current system which you may need to clarify with the users.

3.6 Research and Site Visits

Let us consider the next fact-finding technique, that is, research and site visits. This particular technique is not used very often because it is not relevant in many projects. Here research means **studying the application and the problem areas**. You could do this by studying the trade journals or you can visit reference rooms in libraries to find out more about the application and the problems. You can discover how others have solved similar problems and whether any tools (whether mechanical or manual) exist for solving the problems. You can visit other companies or departments in other organisations who have already gone through the analysis exercise and get some tips from them. This technique of fact finding was not employed in this project.

3.7 Observation of Work Area

This technique involves observation of place of work and work procedure. In this technique, the analyst watches a person perform certain tasks and tries to understand the system. This technique is not very commonly used.

Can you think of the reasons?

- To create a 'good impression' on the analyst, very simple or routine work may be shown to the analyst. In which case the analyst will not be able to observe the special or irregular cases.
- The tasks may be carried out at odd times, e.g., there may be a graveyard shift (night shift) which may be difficult for the analyst to observe or there may be too many interruptions.
- In case persons have been performing a task contrary to the standard operating procedures, s/he may perform it correctly temporarily while under observation.

Hawthorne Effect

People perform differently on stage as compared to real life, i.e., if someone knows that s/he is being observed, s/he becomes aware of it and subconsciously performs better. The Hawthorne Effect may prove this point.

What is the Hawthorne effect?

It has been described as the rewards you reap when you pay attention to people. The mere act of showing people that you are concerned about them usually spurs them to better job performance. That is the Hawthorne Effect.

Suppose you have taken a management trainee and given her specialised training in management skills she doesn't now possess. Without saying a word, you have given the trainee the feeling that she is so valuable to the organisation that you will spend time and money to develop her skills. She feels she is on track to the top, and that motivates her to work harder and better. The motivation is independent of any particular skills or knowledge s/he may have gained from the training session. That is the Hawthorne Effect at work.

When people spend a large portion of their time at work, they must have a sense of belonging, of being part of a team. When they do, they produce better. That is the Hawthorne Effect.

Occasionally, managers object saying that observation is not a valid test. Of course they will do a good job if you are watching them. Isn't that the Hawthorne Effect? Which is why some people do not count observation as a useful fact-finding technique.

On the other hand, there are many points in favour of observation

The main point in its favour is that the analyst is able to see exactly what is being done. Sometimes, users may find it difficult to explain complex procedures. In such cases, the analyst may get a clear idea if s/he observes the person at work.

- Observation also reveals the physical conditions (like excessive noise, poor lighting, physical layout and place of movement) which may have a bearing on the work being done and which may have been missed or inaccurately described during discussions.

To obtain maximum benefits from observation, you should be fully prepared. Ideally, the analyst should observe the work during normal hours and during peak times. The analyst should prepare proper data collection forms in which he can record the data

observed. Your findings should be documented during or immediately following the observation.

In this project the efforts were made to watch various activities of the students/teacher etc. to determine the environment of working.

3.8 Questionnaires

All of you would have received and even responded to some questionnaire.

So what is a questionnaire?

It is a document prepared for a special purpose that allows the analysts to collect information and opinions from a number of respondents.

It contains a list of questions. The questionnaire is distributed to the selected respondents; the respondents answer the questions in their own time and return the questionnaire with the answers to the analyst. The analyst can then analyze the responses and reach conclusions.

Sample Questionnaires Used by Us:

- What are the services provided by the Study Centre?
- What are the specific data of Student?
- How Schedule is created for a Teacher?
- How are Marks, Fees and Attendance recorded?
- Some of the Sample Layouts of Reports
- What are the various outputs of Systems?

3.9 Personal Interview

There are always 2 roles in the personal interview.

The analyst is the *interviewer* who is responsible for organising and conducting the interview. The other role is that of the *interviewee* who is the end-user or the manager or the decision-maker. The interviewee is asked a number of questions by the interviewer.

In this project we have done interviews of centre incharge, co-ordinator and other main functionaries to ascertain their expectations of the system.

4.0 SYSTEM DESIGNING

It is the process which starts after the completion of Analysis Phase. In this phase the planning of activity starts whereas in the Analysis phase the important data are gathered.

System Designing consists of various activities. The main activities that are being conducted here:

- Modeling Data Flow Diagram of System
- Creating Entity Relationship Diagram of System
- Creation of Database Dictionary Designing
- Database Design
- Input Form Design
- Output Forms or Reports Design.

Finally, after the completion of this phase, the development phase is performed where these designs are used to achieve the proper look and feel of the software.

4.1 Data Flow Diagram for System

It is one of the important tools of System Designing which enables software engineer to develop models of the information domain and functional domain at the same time. It serves two purposes:

1. Provides an indication of how data are transformed as they move through the system
2. Depicts the Functions (and sub-functions) that transform the data flow.

The DFD provides additional information that is used during the analysis of the information domain and serves as a basis for the modelling of function. A description of each function presented in the DFD is contained in a **Process Specification**. As the DFD is refined into greater levels of details, the analyst performs an implicit functional decomposition of the system.

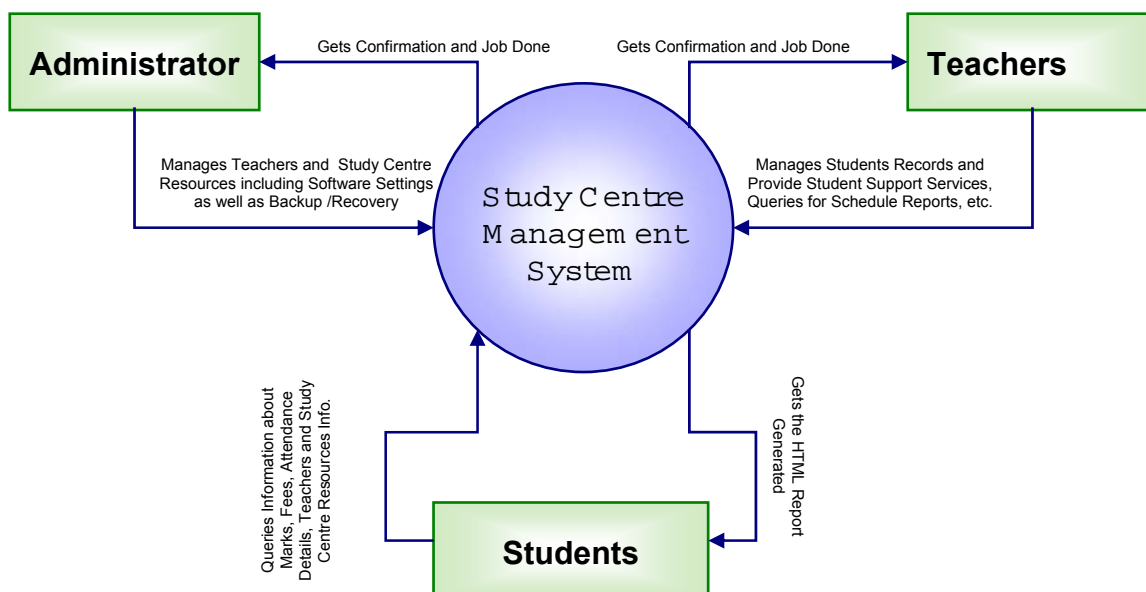
One of the faults in our process while developing the system was that the system was analysed, designed and implemented but the DFDs were developed as an afterthought. Thus, you will find the following problems in the DFDs:

- The dataflow have not been labeled in many cases.
- The processes are shown as linear processes especially in the second level DFDs
- The verification activities are being performed in sequence, although they are not practically done that way in the actual situation.

Exercise

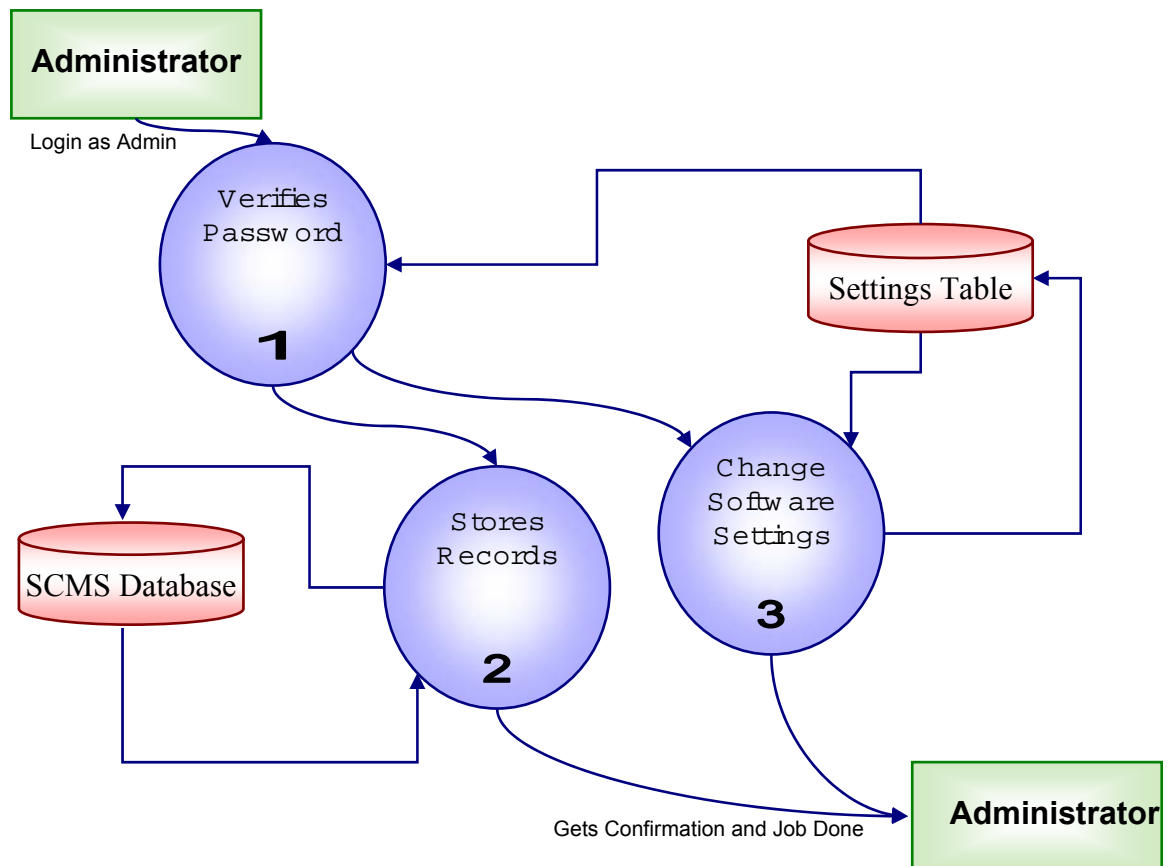
You must discuss the DFDs thoroughly in your sessions and come up with the most optimum DFDs.

0th Level (Context Flow Diagram)

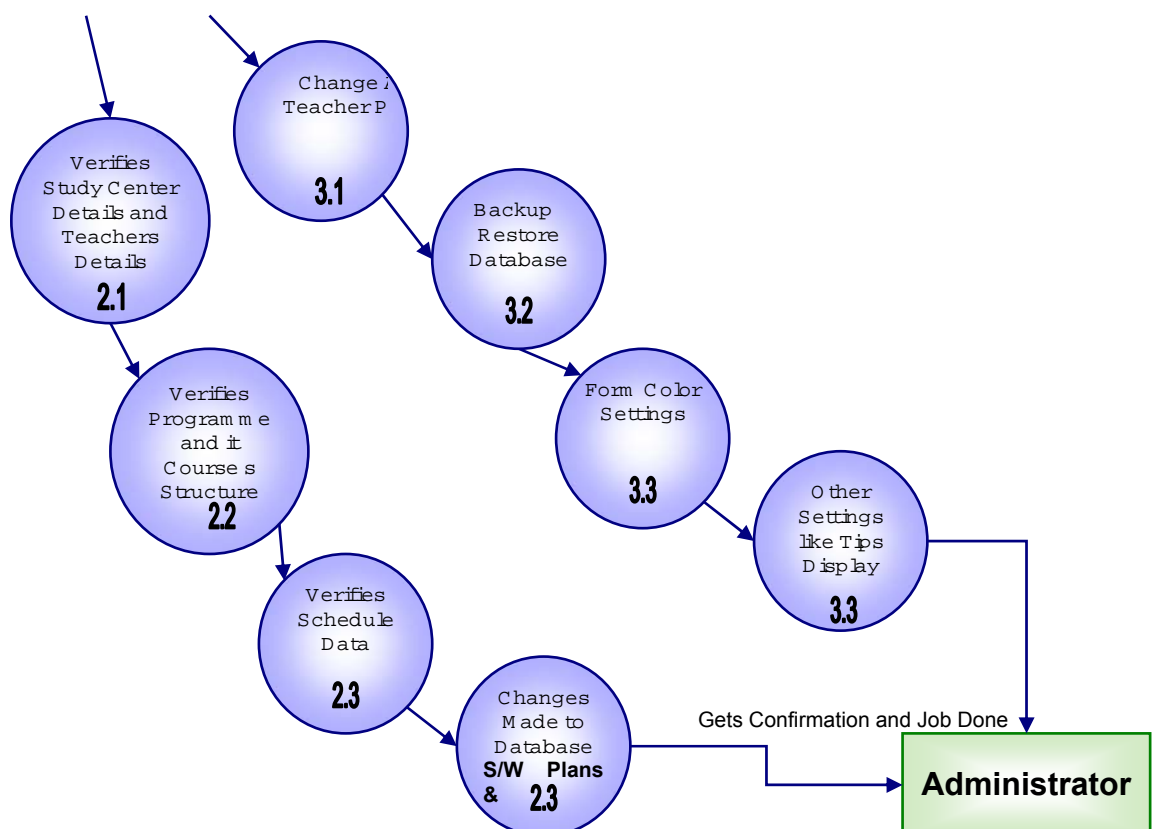


DFD For Administrator

1st Level

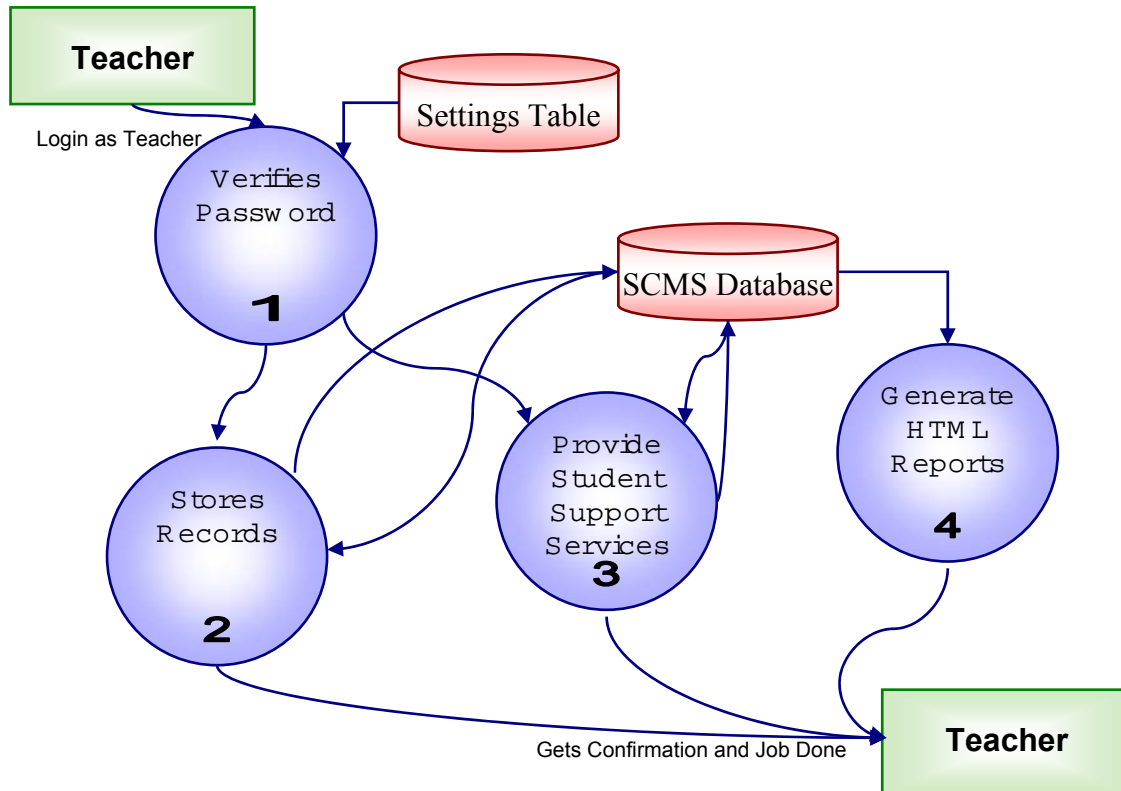


2nd Level

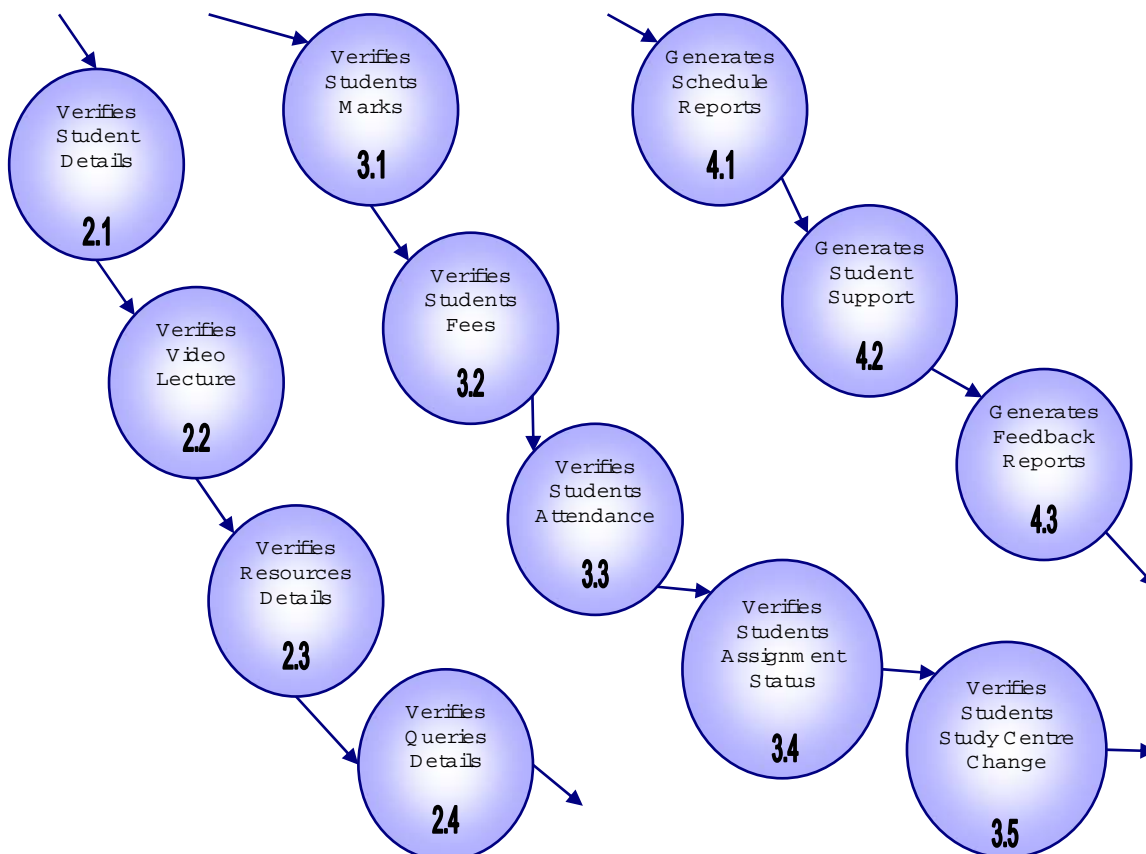


DFD for the Teacher

1st Level

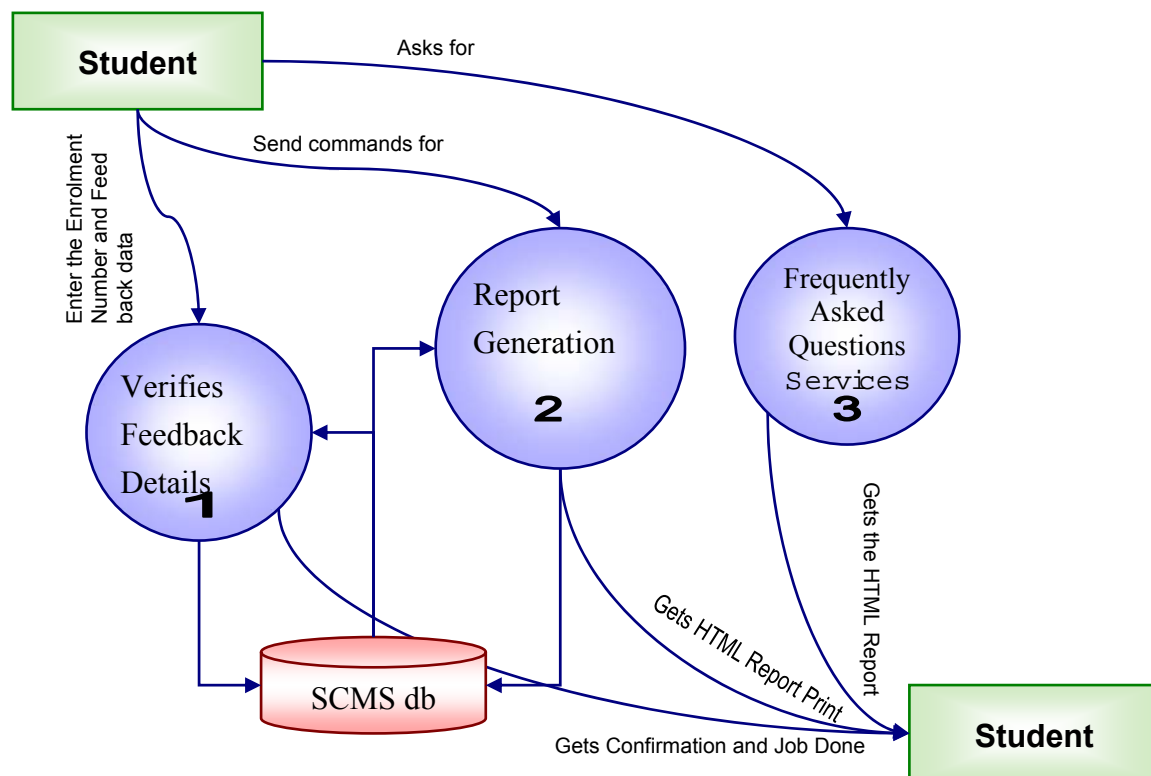


2nd Level



DFD for the Students

1st Level



2nd Level

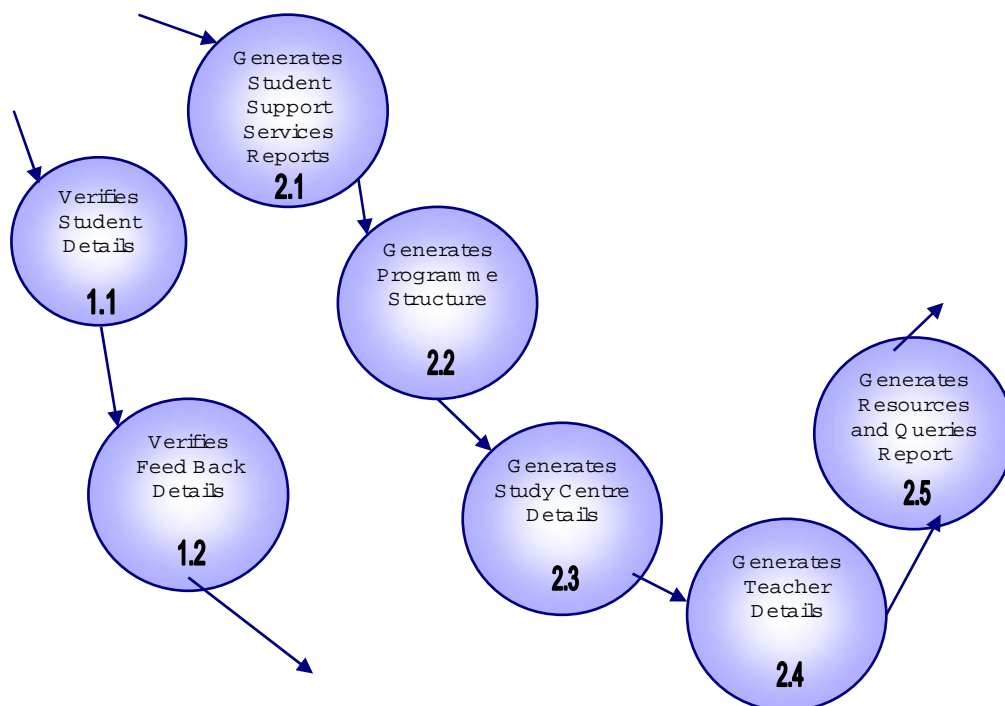


Figure 9: All the DFDs for the System

The **Data Flow Diagrams** as above give a brief idea on how our system is working. It also suggests to us the type of users who are using this system and the processes involved in the system.

After having a brief idea we would like to now explain the relationships among the users and data records available in the system in the next section.

4.2 Entity-Relationship Diagram for the system

The ER Diagram of the system is shown in *Figure 10*.

Figure 10: Entity Relationship Diagram for System

