

Synopsis

TITLE

HOSPITAL MANAGEMENT SYSTEM

(USING CLIENT –SERVER DATA BASE)

A “**HOSPITAL MANAGEMENT SYSTEM**” is a computerized management system. This management system has been developed to form whole management system including Employees, Doctors (consultants), Nurses, Patients, Bills, and Complains etc. This system also keeps the records of hardware assets besides software of this organization. The proposed system will keep a track of Employees, Doctors, Patients, Accounts and generation of report regarding the present status. This project has GUI based software that will help in storing, updating and retrieving the information through various user-friendly menu-driven modules.

INTRODUCTION

Health of citizen is the wealth of Nation. India has contributed the most ancient Medical science “AYURVEDA” to the world besides other Medical sciences. This field had witnessed a rapid metamorphosis in all of its sections. With immensely increasing advancement in the field of technology, ambit of medical sciences has enlarged making it more sophisticated, diagnosis and cure-oriented. With the advent of 21st century, Health care has become an industry having tremendous potential.

This century witnessed a giant leap in information technology. Computers are not only used to diagnose the illness or for doing surgery with one hundred percent accuracy, but also they are used to increase the efficiency in all fields ranging from fixing the appointment with the Doctor to keeping the record of the Patient.

Software application can provide solution and services for the global health care industry. By using the cutting edge technologies, Hospital Management can be improved with efficient work flow and communication. Any time any where facilities of the INTERNET have helped the Medical fields to integrate into a single unit. Various Hospitals across the globe can be connected together. They can share information and even services. Details of the Patients, their previous visits etc. are totally not perceptible without a computer. Relevant Informations are always stored in the computer and are available instantly in front of the user.

Medical Transcription, one of the latest technologies in health world provided with Internet facilities helps patient’s interaction with different Health Experts of the world for an astute clinical analysis as if treatment is going on at his home and doorstep.

OBJECTIVES OF THE PROJECT

The project “Hospital Management System” is aimed to develop to maintain the day-to-day state of admission/discharge of Patients, List of Doctors (Consultants), List of medicines, Bills etc.

There are following main objectives of the Hospital:

- Keeping records of admission of patient.
- Keeping patient-care as utmost priority.
- Scheduling the appointment of Patient with Doctor (Consultant) to make it convenient for both.
- Scheduling the services of specialized Doctor (Consultant) and emergency properly so that facilities provided by Hospital are fully utilized in effective and efficient manner.
- Keeping records of Medicine department in a meticulously arranged order so that the treatment of Patient becomes quick and satisfactory.
- Keeping details about the consultants, their Prescriptions and treatments, surgery reports etc.
- Keeping the best laboratory facilities and diagnostic tools for early and clear diagnosis of the disease and early cure and disposal of the patient.
- Keeping explicit details about the patient’s diseases, diagnosis and management for comprehensive research.
- Keeping the records of salary structure of the employees of Hospital by billing approach.

PROJECT CATEGORY

OVERVIEW OF RDBMS (Project Category)

The project work is entitled "Hospital Management System" category "RDBMS". Hence before discussing anything about the project, a brief discussion of the related basic concepts is necessary.

As a software developer or as a programmer, we are expected to design and develop every program that works correctly, efficiently and at the same time is easy to be used by any person who may or may not be well versed with the computer and its capabilities. The application programs written in any language must be such that the user of that program should find it extremely friendly in the sense that not much effort is/should be required on the part of the user to understand and use its application. The package should be user-friendly.

BASIC DATABASE CONCEPTS:

A database is a collection of related information stored so that it is available to many users for different purposes. It consists of a collection of interrelated data and a set of programs to access those data. It is a coherent collection of data with some inherent meaning and design built and populated with data for a specific purpose. A database stores data that is useful to us. This data is only a part of the entire data available in the world around us.

To be able to successfully design and maintain database, we have to do the following:

- Identify which part of the world's data is of interest to us.
- Identify what specific objects in that part of the world's data are of interest to us.
- Identify a relationship between the objects.
- Hence, the objects, their attributes and the relationship between them

That are of interest to us are in the database which is designed, built and populated with data for a specific purpose.

Characteristics of a Database Management System

- Provides creation of a file, addition & deletion & modification of data, creation, addition and deletion of entire files.
- Provides retrieval of data collectively or selectively by Database.
- Provides storing and indexing the data stored at the user's discretion and direction.
- Provides performance of Mathematical function and manipulation of the data stored in the database to perform the desired calculation.
- Provides maintenance of data integrity and database use.
- Provides form-based interface for easy accessibility and data entry.
- Provides complex relationships between data.
- Provides keeping a tight control over data redundancy.
- Provides enforcement of user-defined rules to ensure the integrity of table data.
- Provides a centralized data dictionary for the storage of information pertaining to data and its manipulation.
- Provides ensuring that data can be shared across applications.
- Provides automatic intelligent backup and recovery procedures for data.
- Provides different interfaces via which users can manipulate data.

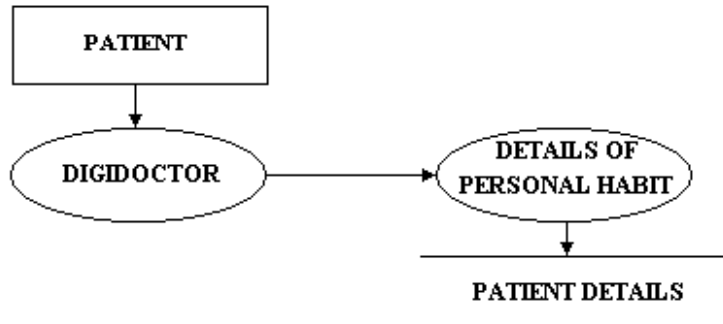
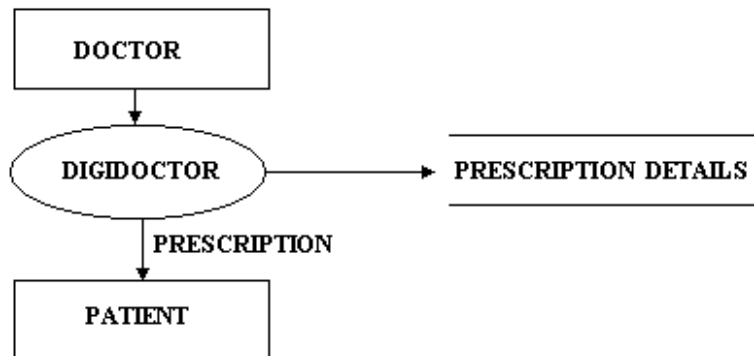
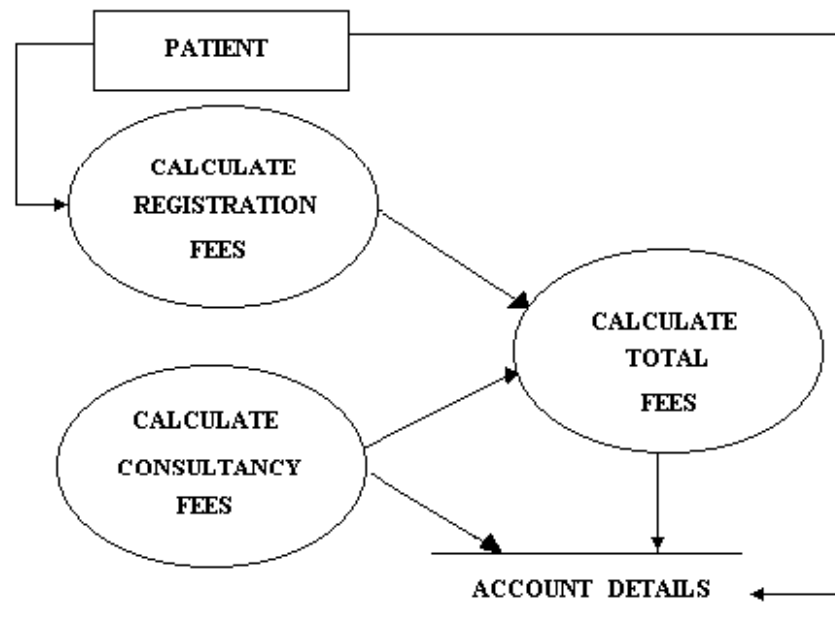
In the early days of computing the DBM System, used to manage data, were of the Hierarchic or Network model. When these were placed into network operating System and multiple users began to access table data concurrently, the DBM system responded to these user requests very sluggishly and was not totally stable when the number of users exceeded four or five.

Oracle 8.0 implements around seven of Codd's laws, Ingress nine, Sybase ten and a half. However, research and development is constantly going on at all these vendor's sites. Each vendor is striving to implement all of Codd's laws in their products; this constantly leads to products upgrades being brought out by the product vendors. This is really good news for programmers. Currently **Oracle 8.0**

implements all rules fully or partially. The programming world is constantly getting easier and easier.

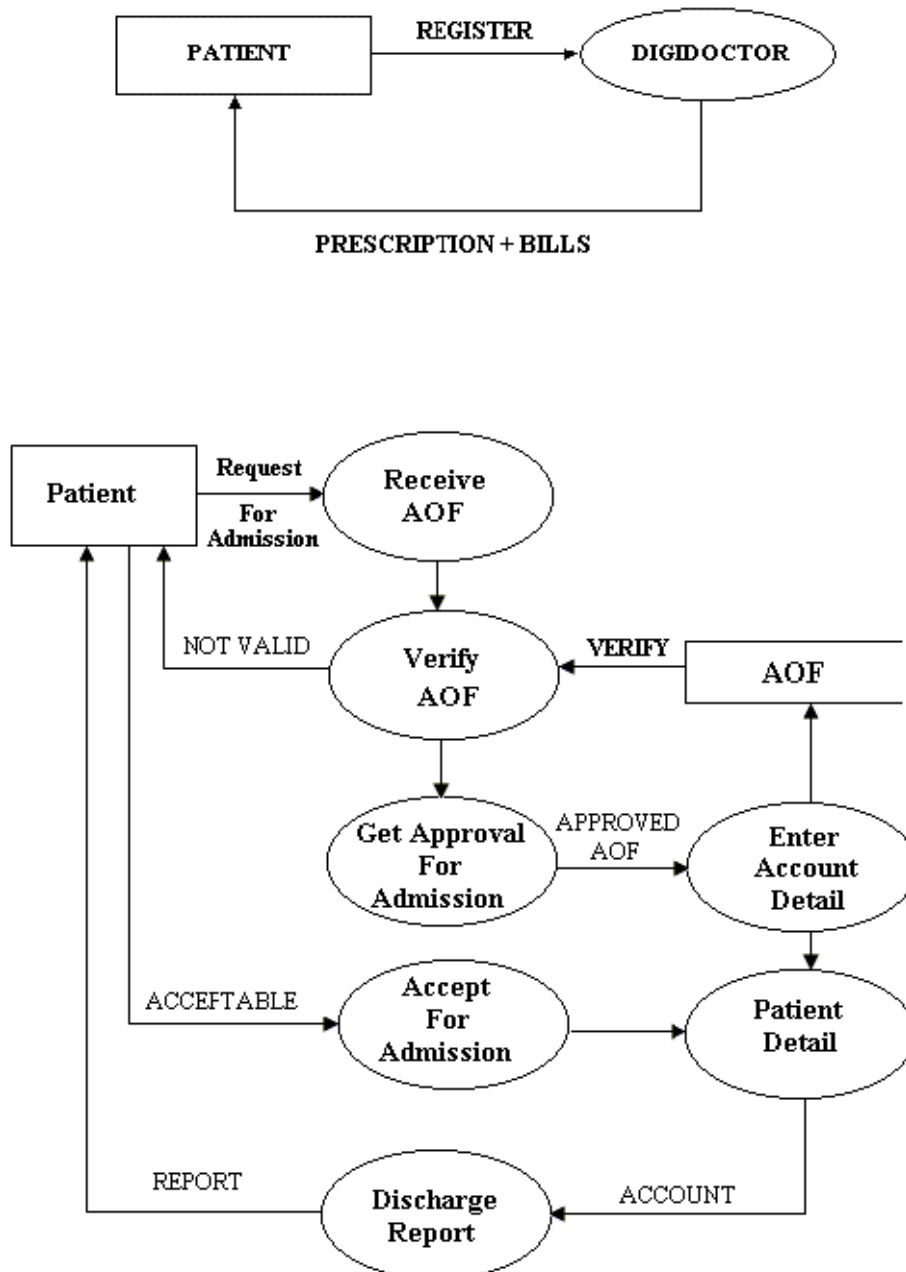
Characteristics of a Relational DBMS Model

- The relational database management model eliminates all parent-child relationships and instead represents all data in the database as simple row/column tables of data values.
- A relation is similar to a table with row/column similar to its data values. The rows of a table are referred to as **Tuples** and the columns are referred as **Attributes**. Several tuples of equal length placed one below the other create a table.
- Each table is an independent entity and there is no physical relationship between tables.
- Most data management system based on the relational model has a built-in support for query language like ANSI SQL or QBE (Query By Example). These queries are simple English constructs that allow adhoc data manipulation from a table.
- Relational model of data management is based on set theory. Built-in query language is designed in the RDBMS, so that it can manipulate sets of data (one or more tuples). The user interface used with relational models is non-procedural because only what needs to be done is specified and not how it has to be done.
- In a RDBMS, all user requests to insert, Update or Delete in a table must be routed through the RDBMS engine only. Direct calls for data cannot be made to the tables themselves.
- If the RDBMS engine is not loaded and running in a computer memory, user security and request for a table data are simply not entertained.

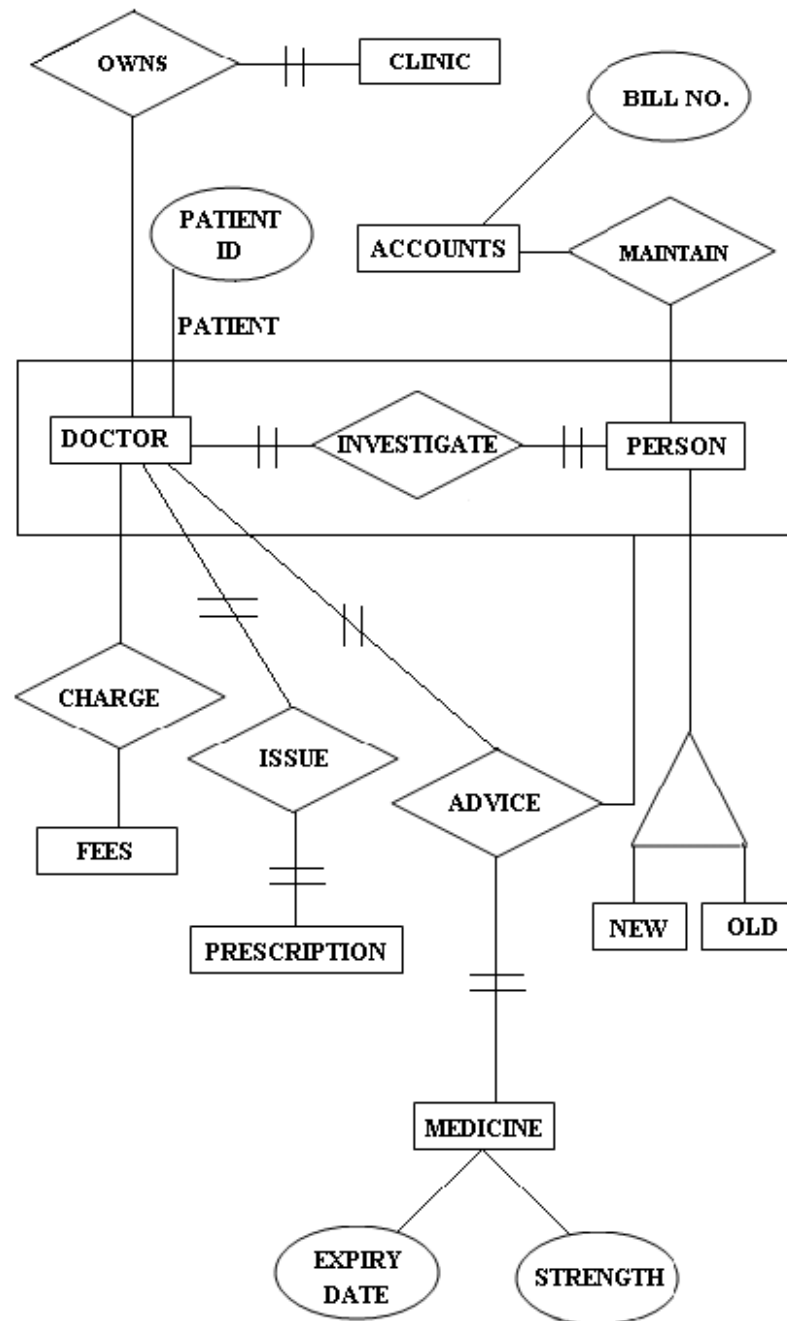
**Analysis
DFD****Module wise Data Flow Diagram****1. REGISTRATION****2. PRESCRIPTION****3. BILLING MODULE**

BILLING

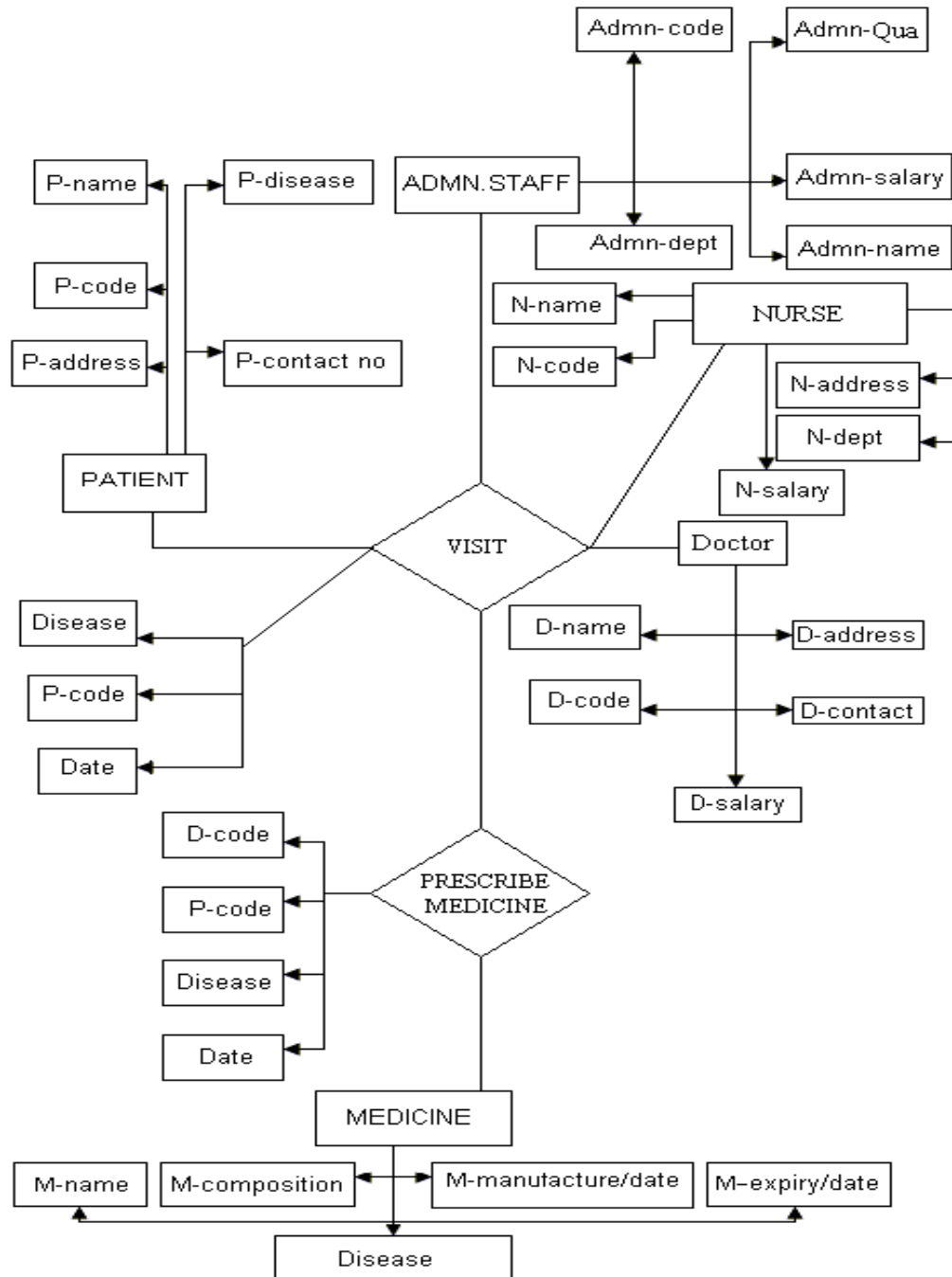
CONTEXT LEVEL DFD



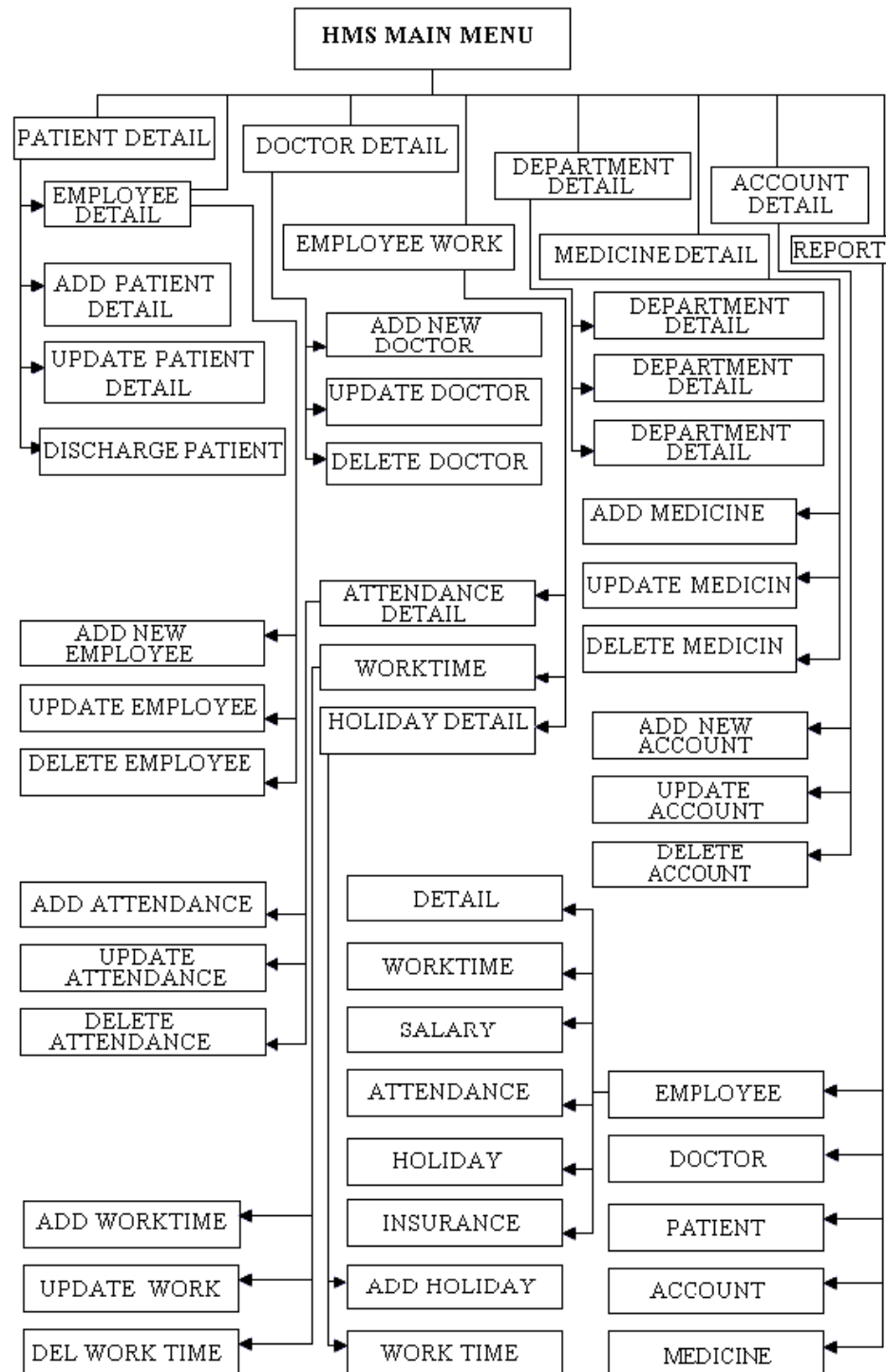
ENTITY RELATIONSHIP DIAGRAM OF DIGITAL NURSING



E-R DIAGRAM



MODULES AND ITS DESCRIPTION



DATA STRUCTURE (TABLE)**Employee Table****Doctor**

Entity Name: doc

Entity Type : Independent

Primary Key : doc_id

Foreign Key : doc_id reference doc.

Description : Contains the information of the Doctors.

Field name	Data type	Length	Constraint	Description
Doctor_id	Number	10	Not Null	Doctor Identification Number
Doct_name	Varchar	12	Not Null	Doctor name.
Edu_Qualification	Varchar2	15	Not Null	Educational Qualification
D_O_B	Varchar2	08	Not Null	Date of Birth
Doc_department	Varchar2	15	Not Null	Department
Doc_Salary	Varchar2	08	Not Null	Doctor Salary
Father's name	Varchar2	20	Not Null	Doctor's Father name
Phone_num	Number	11	Not Null	Doctor Phone number
Email_id	Varchar	12	Not Null	Doctor

				Electronic mail
--	--	--	--	--------------------

Patient Table**Patient**

Entity Name: pat
Entity Type : Independent
Primary Key: pat_id
Foreign Key: pat_id reference pat.
Description : Contains the information of the Patients

Field name	Data type	Length	Constraint	Description
Patient_id	Number	10	Not Null	Patient Identification Number
Pat_name	Varchar	12	Not Null	Patient Name.
Edu_Qualification	Varchar2	12	Not Null	Educational Qualification
D_O_B	Varchar2	08	Not Null	Date of Birth
pat_department	Varchar2	15	Not Null	Department
Pat_fee	Varchar2	08	Not Null	Patient Fee
Father's name	Varchar2	20	Not Null	Patient Father Name
Phone_num	Number	11	Not Null	Patient Phone number

Nurse Table**Nurse**

Entity Name: nur

Entity Type : Independent

Primary Key : nur_id

Foreign Key: nur_id reference nur.

Description : Contains the information of the Nurse.

Field name	Data type	Length	Constraint	Description
Nur_id	Number	10	Not Null	Nurse Identification Number
Nur_name	Varchar	12	Not Null	Nurse Name.
Edu_Qualification	Varchar2	15	Not Null	Educational Qualification
D_O_B	Varchar2	08	Not Null	Date of birth
Nur_department	Varchar2	15	Not Null	Department
Nur_Salary	Varchar2	08	Not Null	Nurse Salary
Father's name	Varchar2	20	Not Null	Nurse Father Name

Admn.Staff Table**Staff**

Entity Name: staff

Entity Type : Independent

Primary Key : staff_id

Foreign Key: staff_id reference staff.

Description : Contains the information of the staff

Field name	Data type	Length	Constraint	Description
Staff_id	Number	14	Not Null	Staff Identification Number
Staff_name	Varchar	16	Not Null	Staff name.
Edu_Qualification	Varchar2	10	Not Null	Educational Qualification
D_O_B	Varchar2	10	Not Null	Date of birth
Staff_department	Varchar2	15	Not Null	Department
Staff_Salary	Varchar2	08	Not Null	Staff Salary
Father's name	Varchar2	18	Not Null	Staff Father name
Phone_num	Number	11	Not Null	Staff Phone number
Email_id	Varchar	12	Not Null	Staff Electronic mail

Employee work Table**Employee work**

Entity Name: work

Entity Type : Independent

Primary Key : Emp_id

Foreign Key : Emp_id reference Emp.

Description : Contains the information of the Employees work.

Field name	Data type	Length	Constraint	Description
Emp_id	Number	13	Not Null	Employee Identification Number
Emp_name	Varchar	12	Not Null	Employee Name.
Edu_Qualification	Varchar2	13	Not Null	Educational Qualification
D_O_B	Varchar2	10	Not Null	Date of birth
Emp_department	Varchar2	15	Not Null	Department
Emp_Salary	Varchar2	08	Not Null	Employee Salary
Father's name	Varchar2	20	Not Null	Employee Father name
Phone_num	Number	11	Not Null	Employee Phone number

Department Table**Department**

Entity Name: dept

Entity Type : Independent

Primary Key : dept_id

Foreign Key : dept_id reference dep.

Description : Contains the information of the Department.

Field name	Data type	Length	Constraint	Description
Dept_id	Number	10	Not Null	Department Identification Number
Dept_name	Varchar	12	Not Null	Department Name
Dep_account	Varchar2	16	Not Null	Department Account
Dep_Doctor	Varchar2	08	Not Null	Department Doctor
Dep_head	Varchar2	15	Not Null	Department Head
Dept_Salary	Varchar2	10	Not Null	Department Employee Salary
Dept_num	Varchar	12	Not Null	Department Contact Number

Account Table

Account

Entity Name: a/c

Entity Type : Independent

Primary Key : a/c_id

Foreign Key : a/c_id reference a/c.

Description : Contains the information of the Accounts.

Field name	Data type	Length	Constraint	Description
Doctor_id	Number	10	Not Null	Doctor Identification Number
Doct_name	Varchar	12	Not Null	Doctor name.
Edu_Qualification	Varchar2	15	Not Null	Educational Qualification
D_O_B	Varchar2	08	Not Null	Date of birth
Doc_department	Varchar2	15	Not Null	Department
Doc_Salary	Varchar2	08	Not Null	Doctor Salary
Father's name	Varchar2	20	Not Null	Doctor's Father name
Phone_num	Number	11	Not Null	Doctor Phone Number
Email_id	Varchar	12	Not Null	Doctor Electronic Mail

PROCESS LOGIC

As all the software program is divided into different modules to know its details of whole processing and they will interact with each other to maintain the data and information module structure is designed with the concept of considering the case of maintaining the logic for modification and maintenance.

Since the software program has to interact with database. It is basically based on Visual Basic as Front end tool and Oracle as Back end tool for data base architecture each the module is interacting with data base and gives the design output as follows

- Patient details
- Doctor details
- Employee details
- Medicine Details
- Department details
- Report details
- Account details
- Employee work details

PATIENT DETAILS:

This module is the most important module of the “Hospital”. Unless there is patient there is no means of having hospital. In this module we know the details of patient. Patient which come from any part take treatment in form of admitted basis or out patient door basis. Patient is an important factor for Hospital.

DOCTORS DETAIL:

This module is equally as important as patient. In this module Doctors check the patient in CASE of O.P.D and visit the ward in case of admission of patient and do treatment accordingly.

EMPLOYEE DETAIL:

The employee detail module of Hospital through which the treatment are done. In this module the main works are attendance of Employee work time, salary of the employee are shown. In every hospital whether it is private sector or public sector the work are important part off any hospital. It is the survival of any hospital worker.

MEDICINE DETAILS:

The module medicines are common modules of any hospital. In this module details of medicine are kept as records. The stock of medicine, date of manufacturing, date of expiry, composition of medicine is kept in this module. Medicine composition is also adjusted so that required medicines reach to required ward so that treatment can have in proper way.

REPORT DETAILS:

This module is very important for each part of the hospital like employee, doctors, patients, etc. This module will help in showing the progress the hospital. In this module report of the patient ,condition of the patient, status of the patient, recovery of the patient, detection of the patient disease all are explain. Doctors report, Employee reports, Employee working reports, Hospital department record ,Medicine prescribe to the Patient, accounts and Billing of all employee including Doctor, Patient, Nurse and administrative employee are kept and do work as and where demand.

ACCOUNT DETAILS:

This module of Hospital management system has an impact of whole hospital management performance. This module provides account details of a hospital, his employee, staff etc. This module describes the inflow of amounts and out flow of amounts of hospital by different mode like patient, test, salary, electricity bills etc. This Account details about the budget and balance sheet of Hospital and also explains the Hospitals present status and how rich is hospital.

OUTPUT / REPORT GENERATION

The out put and report generation for Hospital management system will include the following details

- Doctors detail
- Visiting doctors detail
- patient detail
- outpatient door detail
- staff detail
- nurse detail
- ward boy detail
- employee detail
- employees work time detail
- emp_work detail
- emp attendance detail
- medicine detail
- medicine prescribed detail
- holiday detail
- accounts detail

VALIDATIONS CHECK

Validation for any organization especially for hospital is necessary checks to make the entire projects more reliable, strong, effective, and more efficient. When the user passes the value it would be checked to fall within the range of software provided for hospital management system. If values are incorrect or any data type error then the appropriate message would be flash and the user come to know the exact position fault for error in software. Due to the validation check software becomes more effective as it does not accept the wrong entries to the data.

In my project the validations are performed through the following fields in form of keys and values:-

- **Primary key**
- **Foreign key**
- **Limit values**

PLATFORM

WINDOWS 98 :-

Recently Windows has become a powerful platform for development of Graphics User Interface (GUI) based software.

The feature supported by Window98 to make it one of the most powerful operating system for both general user and developers. This powerful operating system controls the over all activities of the computer.

- Windows 98 includes tools that help your computer run faster.
- Windows 98 includes a suit of program designed to optimize your computer efficiency.
- Windows 98 improves computer reliability by introducing new wizards, utilizes and resources that keep your system running smoothly.
- Windows 98 makes you r computer easier to use with new and enhance features.
- Window 98 is friendly operating system to connect to the web to get software updates

TOOLS

FRONT-END : VISUAL BASIC 6.0

Visual Basic is an ideal Window programming language for developing sophisticated applications that has been developed at Microsoft Corporation. It makes use of Graphical User Interface (GUI) for creating Robust and powerful applications and it includes all the necessary extension required to produce Windows program.

It is an object oriented event driven programming language.

It means all activities are dependent upon the objectives like Command button, Text box, label and also dependent upon the events like Click, Change, Key press, Key down etc.

There are two types of events:

- i. **User** events
- ii. **System** event
- i. The action or event which is taken or performs by the user.
- ii. The action or event which is taken or performs by the system.

Visual Basic applications are very popular as front-end to many client server database systems like SQL server, Oracle etc. The basic idea behind using Visual Basic as front-end it is easy to understand, easy to learn. The application we built in visual basic are not limited the tools provided in the development environment

There are external tools that can enrich the application. Microsoft Visual Basic for Windows comes in working model, a learning edition, a professional edition and an enterprise edition.

The basic version learning edition is built primarily to creation to stand alone application and basic programming.

The professional edition provides developer with fullest of tools for developing solution including **ACTIVE X** controls, the internet information, server application, designer, integrated visual data base tools and data environment.

ACTIVE X data object (**ADO**) and dynamic **HTML** page designer, the Enterprise Edition includes all the source safe, **SNA** server etc.

The significant functionality's of Visual Basic 6.0 is outline as follows.

- Data Access
- Internet Features
- New and updated Controls
- Component Creation
- Language Features
- Wizards

TOOLS / PLATFORMS/H/W & S/W SPECIFICATION**TOOLS/PLATFORM & LANGUAGES TO BE USED**

- **TOOLS**

FRONT END	:	VISUAL BASIC 6.0
BACK END	:	ORACLE 8.0

- **PLATFORM**

OPERATING SYSTEM	:	WINDOWS 98
------------------	---	------------

- **HARDWARE SPECIFICATION**

PROCESSOR	:	P III
RAM	:	128 MB
STORAGE CAPACITY (HDD)	:	20 GB
DRIVE CD	:	52 X CD
DRIVE FLOPPY	:	1.44 MB

- **SOFTWARE SPECIFICATION**

OPERATING SYSTEM	:	WINDOWS 98
RDBMS	:	ORACLE 8.0

SCOPE OF THE FUTURE APPLICATION

There is always a scope of betterment and the candidate system is not against this perception. At present the system satisfy most of the functions of hospital management system. This project is especially designed for HOSPITAL to take appropriate step to improve the working standard and documentation through computerization. Since our system is going to be implemented as the application of “Hospital management information system” on Visual Basic as Front end and Oracle as Back end tool, so it will help better for both user as well as developer.

This project has been developed in keeping in view of requirement of the Hospital.

As far as my projects cover all important details of information of Hospital, therefore in future it will help to gather such information about the information of Hospital.

The main scope of futures may be:

- Generates the important report on day wise or month wise.
- Storing large amount of data for future point of view.
- Reducing manual efforts for maintaining the system.
- Reducing process time.
- Emphasis on accuracy data.
- Assures security and validity. Provision for enhancement without disturbing the developed modules.
- Object oriented design.
- Fully automated, need for human intervention
- The project being developed with friendly user interface can be used by any user.

INTRODUCTION

Health of citizen is the wealth of Nation. India has contributed the most ancient Medical science “AYURVEDA” to the world besides other Medical sciences. This field had witnessed a rapid metamorphosis in all of its sections. With immensely increasing advancement in the field of technology, ambit of medical sciences has enlarged making it more sophisticated, diagnosis and cure-oriented. With the advent of 21st century, Health care has become an industry having tremendous potential.

This century witnessed a giant leap in information technology. Computers are not only used to diagnose the illness or for doing surgery with one hundred percent accuracy, but also they are used to increase the efficiency in all fields ranging from fixing the appointment with the Doctor to keeping the record of the Patient.

Software application can provide solution and services for the global health care industry. By using the cutting edge technologies, Hospital Management can be improved with efficient work flow and communication. Any time any where facilities of the INTERNET have helped the Medical fields to integrate into a single unit. Various Hospitals across the globe can be connected together. They can share information and even services. Details of the Patients, their previous visits etc. are totally not perceptible without a computer. Relevant Informations are always stored in the computer and are available instantly in front of the user.

Medical Transcription, one of the latest technologies in health world provided with Internet facilities helps patient's interaction with different Health Experts of the world for an astute clinical analysis as if treatment is going on at his home and doorstep.

System Analysis And Feasibility Study

Preliminary Investigation

This is the first phase and consists of a brief survey of the areas involved and will result in taking the project into the next phase, postponing development for a period or recommending that no further action be taken. Sometimes it is subdivided into a preliminary investigation (initial study) followed by a more detailed feasibility study. The phase is initiated by management, who perceive the need because of changes or expected changes in the Hospital environment, limitations or failure of existing systems, or the awareness of technological advances relating to the particular area involved in particular systems which competitors are developing.

Information systems projects' originate from many reasons: to achieve greater speed in processing data, better accuracy and improved consistency, faster information retrieval, integration of business areas, reduced cost and better security. The sources also vary project proposals originate with department managers, senior executives and systems analysis.

Sometimes the real origin is an outside source, such as a government agency which stipulates a systems requirements the organisation must meet. When the request is made, the first systems activity, the preliminary investigation, begins. The activity has three parts: request clarification, feasibility study and request approval.

Request Clarification

Many requests from employees and users in organisations are not clearly stated. Therefore, before any systems investigation can be considered, the project request must be examined to determine precisely what the originator wants. A simple telephone call may suffice if the requester has a clear idea but does not know how to state it. On the other hand, the requester may merely be asking for help without knowing what is wrong or why there is a problem. Problem clarification in this case is much more difficult. In either case, before any further steps can be taken, the project requests must be clearly states.

This pahse (initial study) involves estimating whether or not a development project is worthwhile. Problems with the current automated or manual system are identified, as well as the benefits and costs of an alternative system. If the benefits seem to outweigh the costs (especially when compared with competin projects), a green signal may be given to continue the project, and detailed plans and schedules are dtafted for making the system a reality.

The propsed solution to the user's problem may involve something between dramatic change (completely new system) and slight change to the present system. If the present system is manual and a computer system is proposed, the development project will probably be very large. At the other exttreme are small development project that represent slight changes to existing systems, such as sorting information in a different way or inserting subtoals or adding new columns to a report.

The objectives of this phase are:

1. To determine the feasibility of computerisation of a particular system or area of operation.
2. To define clearly the objectives, scope and limitations of the project.
3. To establish a good working relationship between the user department and the data processing (DP) department.
4. To acquaint user management with the approach and method of work in systems development.
5. To estimate the resources required for system development , live running and maintenance.
6. To identify the likely benefits which should accrue from the introduction of the system.

During this phase, which should be as short as possible, the systems analyst will be concerned with:

1. How the present system works?
 2. The staffing levels involved their grades and duties.
 3. The volume of work: statistics on the various types of transaction, level Of overtime working, employment of casual staff, etc. Any current backlog of work and any seasonal influences on the workload.
- The time taken to process data through the system, delays in issuing Management reports, etc.

6. Lists of all documents, files and reports associated with the system.
7. Interfaces with other systems.

An initial study calls for learning as much as possible about the proposed project. At this early point in the project, rough estimates are made of the following:

1. Desired outputs of the system.
2. Available inputs for the system.
3. Time required completing the changes.
4. Feasibility of such changes.

During the preparation of the initial study, information must be gathered from many sources.

Within the organisation, information is gathered from users, managers, and system developers.

Users and managers must be interviewed to find out what they need from the system.

System developers have a wealth of knowledge of the organisation's current capabilities and operations in terms of providing inputs to the system. External information is also critical for determining what new possibilities are offered by vendors in terms of systems available and programs for providing specific types of output. New technologies and developments may allow faster processing, more storage, and lower costs than ever before.

At this point in the project, it is usually very difficult and expensive to specify system characteristics accurately. Often it is impossible to be precise without actually doing much of the proposed work -- including extensive interviewing of users and very detailed studying of procedures. A quick guess about the system's characteristics is all that is needed at this point.

To develop of rough plan of the outputs of the proposed system, brief interviews are held with the intended users. These interviews result in short descriptions of how the outputs will be displayed or printed on video displays or printed in reports. The inputs required to produce the required outputs must be listed and the sources of these inputs determined. It is important to discover if new files will have to be created or sources of information developed, or if the inputs are already gathered and stored in the current system.

A tentative, general schedule for developing the system should be described. It is determined by the users and the developers and must take into account the urgency of the business need and the limited scheduling resources of the developers.

FEASIBILITY STUDY

Technical Feasibility:

In this project, I can say that this project will be technical feasible. This is successfully satisfying the users of the basis requirement. The tools and application software are used in this project are very popular and easily available across the world.

My project is demo project completely applicable the ground level of department desktop computer this can easily enhanced by a team of experts on the basis of my project and its data collection report.

For the setup of the project minimum software and hardware requirements are given above “package collection”.

Operational Feasibility:

There wouldn't be any of problem in this type of project, because HOSPITAL MANAGEMENT SYSTEM taking every possible decision to implement this type of system. After small training of office assistants they will be able to work on it.

Economic Feasibility:

Economic feasibility requires one to make the cost benefit analysis, if I say it in technical term. In a layman's language it is a comparison of the cost of installing system and benefits according from it.

At this level the project may be called a demo project at desktop level. Therefore, it is not possible to measure the benefits accruing and cost of installation. Such study can be made only the project advances to a higher level.

Management Feasibility:

The senior level officers are supportive in implement this project for motivation of clerical or no-technical employee.

Legal Feasibility:

This project is not expected to violate any statute made 3 under the Constitution of India and addition it is approved by IGNOU to do this project.

Time Feasibility:

Implementation and installation time takes only 30 to 45 minutes.

Technical Feasibility:

In this project, I can say that this project will be technical feasible. This is successfully satisfying the users for the basis requirements. The tools and application software are used in this project are very popular and easily available across the world.

My project is a demo project completely applicable the groun level of department on desktop computers. This can easily enhanced by a team of experts on the basis of my project and its data collection report.

For the setup of the project minimum Software and Hardware requirements are given above in “Package Selection”.

Operational Feasibility:

There wouldn't be any of the problems in this type of project, becaue *HOSPITAL MANAGEMENT SYSTEM* taking every possible decision to implement this type of system. After small traning of office assistants they will able to work on it.

Analysis Document:-

Determining systems requirements requires analysis of the facts in hand. Systems analysis is fact-finding followed by analysis of the facts. The analyst first defines and documents, then analyzes. Descriptions and documentation developed as a result of the fact-finding effort are studied to evaluate current system performance and establish requirements to be met in a new design. The conclusions drawn during this activity form the basis for the transition to design and to all over other development activities that follow.

I perform requirements analysis for the development of the project “Hospital management system”. For this I visit the hospital. In hospital the existing system is not computerized. They use manual work on paper for its management works. Since all the processing is done manually, it takes a lot of time. To collect all required data for good management is done on some specific format.

Management of a Hospital is being time critical in nature; smooth running of the management software has been a nightmare for both the management department as well as the IT department. Constant updating is required to the management software due to the ever-changing statutory and internal requirement, adding to woes.

For developing “Hospital management system”, the work is carried out by me. The work is carried out in accordance with specific requirements of the Hospital.

The forms and the tables required for monthly salary processing is designed and developed with the help of the respected Guide sir. After this stage proper coding is carried out and finally checked and modified by the project guide time-to-time and finally testing and debugging is carried out thoroughly and minutely. Reports are prepared in the last. The analysis process assists me in the development of good design for the system.

⇒ **Design Specification:-**

The topic provides idea regarding general structure of application keeping system constraints and functionality, in view. The design means to plan or sketch out the form and method of a solution. The design represents the major characteristic of the final system and determines the upper bound in quality for the system. System design emphasizes on two aspects of a system:

- Dividing the system into components.
- Defining the interrelationship between the components.

A fundamental objective in the design of an information system is that computer and communication technology specified in the design should always be secondary to the results, the system is intended to produce. System design consists of two steps:

- **Logical Design:-** This steps describes the features, the inputs, the outputs, tables, databases and procedures to meet the project requirements. This gives the detailed specification for the new system.
- **Physical Design:-** This steps involve production of software. Programs are written to accept user input, process the data, produces output or reports and store data in database.

We divide the project design into four fragments:-

1.) Output Design:- For many end-user, output is the main reason for developing the system and the basis on which they will evaluate the usefulness of the application. Output design involves....,

- Determine what information is present?
- Decide whether to display or print the information.
- Presentation in an acceptable format.

2.) Input Design:- Input design specifies how data are accepted for processing. This involves...,

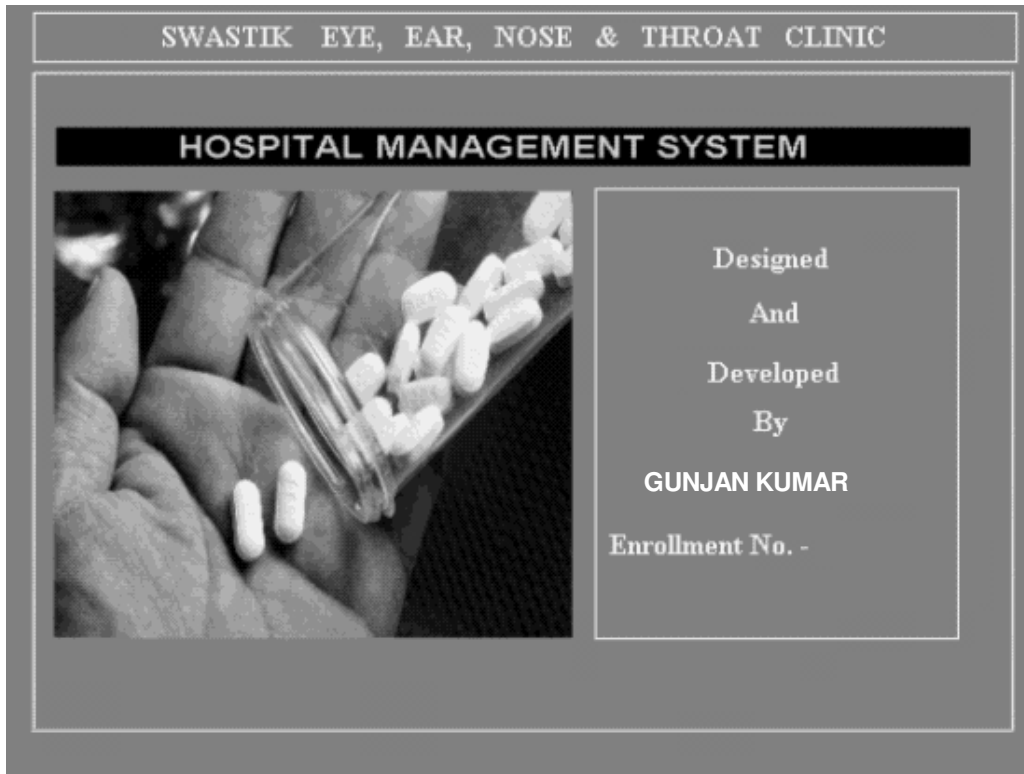
- What data to input?
- What medium to use?
- How data should be arranged or coded?
- The dialog to guide user in providing input?
- Method for performing input validation and steps to follow when error occurs.

3.) Control Design:- While entering data or in requesting the performance of certain functions there may occur errors which could result in improper use of the system. Controls provide ways to:

- Ensure that only authorized users access the system.
- Guarantee that transactions are acceptable.
- Validate the data for accuracy.
- Determine whether any necessary data have been omitted.

Database Design:- The collection data is usually referred as the database. The database contains information about the particulars of an enterprise. The management of data involves both the definitions of structures for the storage of information and provision of mechanism for the manipulation of information. In addition, the database system must provide for the safety of information stored in the database despite system crashes or attempt to unauthorized access.

User interface Design



Form - 1

Splash Form

Splash Form contains only information about the project.

Hospital Management System - [First_Page]

Time : 3:45:57 PM Dat 26/8/2006

WELCOME TO HOSPITAL MANAGEMENT SYSTEM

Please enter your identification (User Id) : -

Please enter your password :-

Control Button's

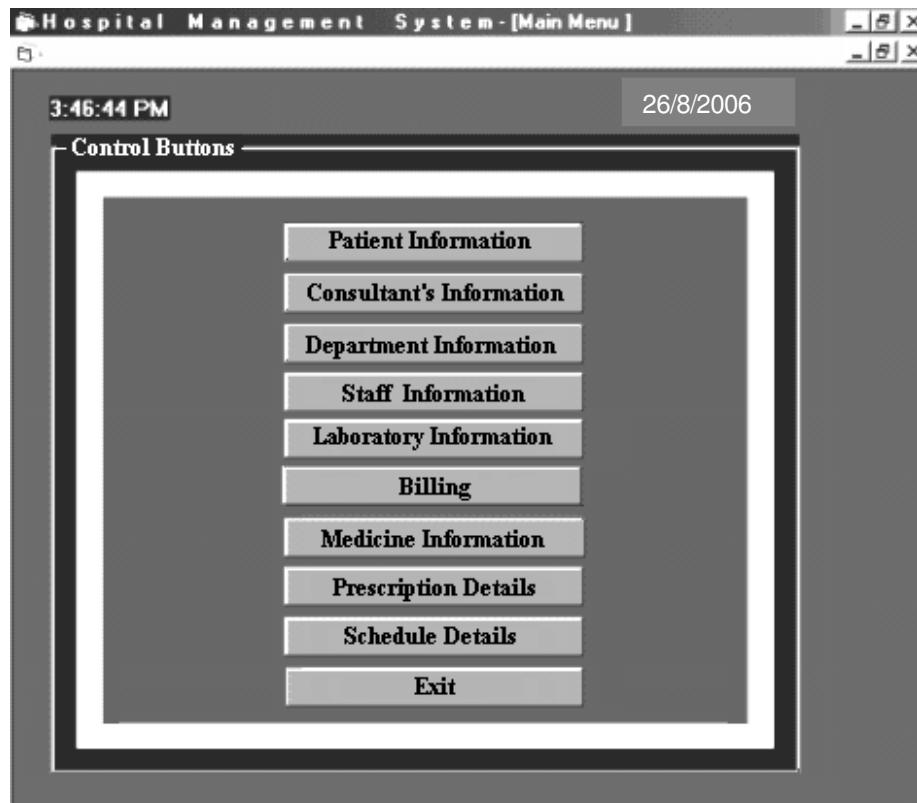
Login Cancel Exit

Form - 2

Login Form

This interface is used for checking the unauthorized user to enter the system/software. It contains the following fields:

- a. User name/Id
- b. Password



Form - 3

Main menu /option form

This form actually interfaces the user and from here they can access the whole system. It contains the option list to move one form to another as well as searches form that is as follows:

1. Patient Information
2. Consultant Information
3. Department Information
4. Staff Information
5. Laboratory information
6. Billing information
7. Medicine information
8. Prescription details
9. Schedule details

Hospital Management System - [Search_for_patient]

Please Enter Patient Identification Number :

Control Button's

Form - 4

SEARCH FOR PATIENT

This form actually searches patients.

Patient Information

Patient No : 101 Date of birth : 09/09/79

Date : 10/10/04 Age : 24

First Name : Govindo Blood Group : B+

Last Name : Ghosh Address : Kankerbagh, Doctor's Colony, Patna

Occupation : Student

Gender : ☒ Male ☐ Female

Pin code : 800020

Marital Status : ☐ Married ☒ Unmarried

Ward : Gh

Room type : AC

Control Button's

Add Save First Next Last Previous Modify Find Delete Exit

Form – 5

Patient Information

This form captures the details of issued account by the add button through the Hospital Management system and also checks all the validity conditions of the given fields. It contains following fields –

1. Patient Id
2. Ward Name Holder name
3. Registration Type
4. Registration date
5. First Name
6. Occupation
7. Reference person
8. Contact number
9. Gender

Consultant's Information

Personal Information

ID No. Registration No.

First Name Specilization

Last Name Nationality

Schedule Day Schedule Time

Consultant's Personal Details

Adress Date of Birth

Telephone No.

Pin Code Mobile No.

Control Button's

Add Save First Next Last Previous Modify Find Exit

Form - 6

Consultant Information

This form captures the details of patient who is having a registration with the hospital. It contains following fields –

1. Consultant Id No
2. First name
3. Last name
4. Schedule Day
5. Registration number
6. Nationality
7. Schedule Time
8. Address

The screenshot shows a window titled "Hospital Management System - [Department_information]". Inside the window, there is a form titled "Department Information". The form contains the following fields: "Department No." (a single-line text box), "Contact No." (a single-line text box), "Department Name" (a single-line text box), "Description" (a single-line text box), "Head Name" (a single-line text box), and "Address" (a multi-line text box with two rows). Below the form fields, there is a section titled "Control Button's" containing a row of buttons: "Add", "Save", "First", "Next", "Last", "Previous", "Modify", "Find", and "Exit". The "Modify" button is highlighted with a grey border.

Form - 7

Department information

This form captures the information about the department of the hospital. It contains following fields –

1. Department number
2. Department Name
3. Description
4. Head name
5. Address
6. Contact Number

Staff Information / Entry Form

Code	<input type="text"/>	Sex	<input type="text"/>
Category	<input type="text"/>	Basic Salary	<input type="text"/>
First Name	<input type="text"/>	Date of Birth	<input type="text"/>
Last Name	<input type="text"/>	Date of Joining	<input type="text"/>
Address	<input type="text"/>	Date of Retirement	<input type="text"/>
Designation	<input type="text"/>		
Contact No.	<input type="text"/>		

Salary Statement

Travelling Allowance	<input type="text"/>	Medical Allowance	<input type="text"/>
Dearness Allowance	<input type="text"/>	Gross Salary	<input type="text"/>
Provident Fund	<input type="text"/>	Net Salary	<input type="text"/>
House Rent Allowance	<input type="text"/>		

Control Button

Main	Back	Calculate
------	------	-----------

Navigation Button

Insert	Save	First	Next	Last	Prior	Delete
--------	------	-------	------	------	-------	--------

Form - 8

Staff Information

This form captures the information about the staff details of the hospital. It contains following fields –

1. Staff code
2. Category
3. First Name
4. Last Name
5. Address
6. Designation
7. Salary
8. Date of joining
9. Date of Retirement

The screenshot shows a software window titled "Hospital Management System - [Transaction_Details]". The form is organized into several sections:

- Transaction Section:** Contains "Transaction Type" with radio buttons for "Debit" and "Credit", a "Date" text field, and "Transaction Mode" with radio buttons for "Cash" and "Checque/Draft".
- Input Fields:** Includes "Debit / Credit" (a dropdown menu), "Please Enter Amount [Rs.]" (a text field), "Please Enter Voucher No." (a text field), "Verified [Y/N]" (a checkbox), and "Verified By" (a text field).
- Patient Information Section:** Includes "Patient Id" (text field), "Reg. No." (text field), "Ward Name" (text field), "Patient Name" (text field), "Last Name" (text field), "Address" (text field), and "Pin Code" (text field).
- Data Control Section:** Contains "Navigate Buttons" with a "Main Menu" button, and "Save / Exit" buttons with "Save" and "Exit" buttons.

Form – 9

TRANSACTION DETAILS

This form depicts the transaction details.

Hospital Management System - [Medicine]

Medicine Information

Medicine Code

Medicine Name Manufacturer Name

Type of Medicine Batch No.

Manufacturer Date

Date of Expiry

Control Button's

Add Save First Next Last Previous Modify Find Delete Exit

Form – 10

Medicine Information

This form captures the detail information of report of all drugs being used here. These are following –

1. Medicine code
2. Medicine Name
3. Type of Medicine
4. Type of Manufacturer
5. Date of expiry.

Hospital Management System - [Prescription Details]

Consultant's Details

Name	<input type="text"/>	Name of Disease	<input type="text"/>
Date	<input type="text"/>	Medicine Prescribed	<input type="text"/>
Next Appointment	<input type="text"/>		<input type="text"/>

Patient's Details

Id	<input type="text"/>	Age	<input type="text"/>
Name	<input type="text"/>	Blood Group	<input type="text"/>
	<input type="text"/>		<input type="text"/>

Control Button's

Add	Save	First	Next	Last	Previous	Modify	Back	Delete	Exit
-----	------	-------	------	------	----------	--------	------	--------	------

Form – 11

PRESCRIPTION DETAILS

Employee Work Schedule

Code

Name

Designation

Time

Day

Time Schedule

☐ 6 AM - 2 PM ☐ 2 PM - 10 PM ☐ 10 PM - 6 AM

Control Button's

Form – 12

EMPLOYEE WORK SCHEDULE

This form depicts the duty-routine with reference to the employees.

FORM – 1 FRMSPLASH

Option Explicit

Private Sub Form_KeyPress(KeyAscii As Integer)

 first.Show

 Unload Me

End Sub

Private Sub Form_Load()

 Version.Caption = "Version " & App.Major & "." & App.Minor & "." & App.Revision

End Sub

Private Sub Frame1_Click()

 first.Show

 Unload Me

End Sub

FORM – 2 FIRSTPAGE

```
Dim a As String
```

```
Private Sub cancel_Click()
```

```
    Text1.Text = ""
```

```
    Text2.Text = ""
```

```
End Sub
```

```
Private Sub cmdexit_Click()
```

```
    Unload Me
```

```
End Sub
```

```
Private Sub Form_Activate()
```

```
    first.WindowState = 2
```

```
    Text1.SetFocus
```

```
    Frame1.Caption = "Control Button's"
```

```
    Frame1.ForeColor = vbYellow
```

```
    login.BackColor = vbYellow
```

```
    Label8.AutoSize = True
```

```
    Label8.FontSize = 15
```

```
    Label8.BackColor = vbBlue
```

```
    Label8.Caption = "Designed and Developed by: IGNOU,
```

```
    Enrollment No. - 013108187 , MCA (Final Year)"
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
    Text1.Text = ""
```

```
    Text2.Text = ""
```

```
    Text1.FontSize = 14
```

```
    Text1.FontBold = True
```

```
Text1.Alignment = 2
Text2.FontSize = 14
Text2.FontBold = True
Text2.Alignment = 2
Timer2.Interval = 50
Timer2.Enabled = True
Label8.Left = Picture2.Width + 10
Label8.Visible = True
Timer1.Interval = 50
Timer1.Enabled = True
Label1.Left = Picture1.Width + 10
Label1.Visible = True
End Sub

Private Sub login_Click()
    If Text1.Text = "x" or Text2.Text = "x" Then
        Main_menu.Show
        Main_menu.WindowState = 2
    Else
        MsgBox "Your Password is invalid / You are not authorize
            user..... Try Again",
            vbQuestion,
        End If
    End Sub

Private Sub Timer1_Timer()
    Label5.Caption = Time()
    Label7.Caption = Date
    Label1.Move (Label1.Left) - 50
    If (Label1.Left + Label1.Width) < 0 Then
```

Label1.Left = Picture1.Width + 10

End If

End Sub

Private Sub Timer2_Timer()

Label8.Move (Label8.Left) - 50

If (Label8.Left + Label8.Width) < 0 Then

Label8.Left = Picture2.Width + 10

End If

End Sub

Private Sub logging_lostfocus()

Text1.Text = ""

Text2.Text = ""

End Sub

FORM-3 MAIN MENU

Private Sub cmdbill_Click()

search_Transaction.Show

End Sub

Private Sub cmdconsult_Click()

Unload Me

Consultant_Details.Show

Consultant_Details.WindowState = 2

End Sub

Private Sub cmddept_Click()

Unload Me

```
        Department_information.Show
        Department_information.WindowState = 2
End Sub

Private Sub cmdlab_Click()
    Unload Me
    Laboratory_Test_Information.Show
    Laboratory_Test_Information.WindowState = 2
End Sub

Private Sub cmdmed_Click()
    Unload Me
    Medicine.Show
    Medicine.WindowState = 2
End Sub

Private Sub cmdpatient_Click()
    Unload Me
    Patient_details.Show
    Patient_details.WindowState = 2
End Sub

Private Sub cmdprep_Click()
    Unload Me
    Search_prescription.Show
    Search_prescription.WindowState = 2
End Sub

Private Sub cmdshedule_Click()
    Unload Me
```

```
frmSchedule.Show
frmSchedule.WindowState = 2
End Sub

Private Sub cmdstaff_Click()
Unload Me
Staff_Information.Show
Staff_Information.WindowState = 2
End Sub
```

FORM- 4 PATIENT INFORMATION

```
Dim con As New ADODB.Connection
Dim Recordsource As New ADODB.Recordset
Private Sub cmdadd_Click()
On Error GoTo adderr
cmdadd.Enabled = False
cmddelete.Enabled = False
cmdmodify.Enabled = False
cmdexit.Enabled = False
cmdnext.Enabled = False
cmdlast.Enabled = False
cmdpre.Enabled = False
cmdFind.Enabled = False
cmdfirst.Enabled = False
cmdsave.Enabled = True
Clear_text
If Not Recordsource.BOF = True Then
```

```

        Recordsource.MoveLast
        MsgBox "TXT" & Recordsource.Fields(0)
        txtpatid.Text = Val(Recordsource.Fields(0)) + 1
    Else
        txtpatid.Text = 101
    End If
    If Recordsource.BOF <> True Or Recordsource.EOF <> True Then
        Recordsource.MoveLast
        Recordsource.MoveNext
    End If
    Recordsource.AddNew
    cmdsave.Enabled = True
Exit Sub
adderr:
MsgBox Err.Description
End Sub

Private Sub cmdback_Click()
    Patient_details.Hide
    Main_menu.Show
    Main_menu.WindowState = 2
End Sub

Private Function Clear_text()
    txtpatid = ""
    txtregis = ""
    txtfirstname = ""
    txtlastname = ""
    txtoccu = ""
    txtrefper = ""
    txtcont = ""

```

```
txtdob = ""  
txtage = ""  
txtblood = ""  
txtadd1 = ""  
txtadd2 = ""  
txtpin = ""  
txtref_cont = ""  
txtward = ""  
txtreg = ""
```

End Function

```
Private Sub cmdfind_Click()  
    Dim id, i, a, current As Integer  
    i = 0  
    id = InputBox("Please enter a Patient id:- ")  
    Recordsource.MoveFirst  
    Do While Not Recordsource.EOF  
        i = i + 1  
        Recordsource.MoveNext  
    Loop  
    Recordsource.MoveFirst  
    For a = 1 To i  
        current = Recordsource.Fields(0)  
        If current = id Then  
            Show_Record  
        Exit Sub  
        End If  
        Recordsource.MoveNext  
    Next  
    MsgBox ("Recourd not found")
```

End Sub

Private Sub cmdmodify_Click()

On Error GoTo down

cmdadd.Enabled = False

cmddelete.Enabled = False

cmdexit.Enabled = False

cmdnext.Enabled = False

cmdlast.Enabled = False

cmdpre.Enabled = False

cmdFind.Enabled = False

cmdfirst.Enabled = False

cmdsave.Enabled = True

Recordsource.UpdateBatch adAffectAllChapters

Recordsource.UpdateBatch adAffectCurrent

Exit Sub

down:

MsgBox Err.Description

End Sub

Private Sub cmdpre_Click()

Recordsource.MoveFirst

Show_Record

End Sub

Private Sub Form_Load()

con.Open "dsn=udsn;user id=scott;Password=tiger"

Recordsource.Open "select * from patient", con, adOpenDynamic,

adLockOptimistic

End Sub


```
Private Sub cmd_Last_Click()  
    Recordsource.MoveLast  
    Show_Record  
End Sub
```

```
Private Sub cmd_Next_Click()  
    Recordsource.MoveNext  
    Show_Record  
End Sub
```

```
Private Sub cmdexit_Click()  
    Patient_details.Hide  
    Clear_text  
    Main_menu.Show  
    Main_menu.WindowState = 2  
End Sub
```

```
Private Sub cmdfirst_Click()  
    Recordsource.CancelUpdate  
    Recordsource.MoveFirst  
    Show_Record  
End Sub
```

```
Private Sub cmdlast_Click()  
    Recordsource.CancelUpdate  
    Recordsource.MoveLast  
    Show_Record  
End Sub
```

```
Private Sub cmdnext_Click()
```

```
Recordsource.CancelUpdate
Recordsource.MoveNext
If Recordsource.EOF Then
    Recordsource.MoveLast
End If
    Show_Record
End Sub
```

```
Private Sub CmdPrior_Click()
    Recordsource.CancelUpdate
    Recordsource.MovePrevious
    If Recordsource.BOF Then
        Recordsource.MoveFirst
    End If
    Show_Record
End Sub
```

```
Private Sub CmdSave_Click()
    On Error GoTo down
    cmdadd.Enabled = True
    cmddelete.Enabled = True
    cmdmodify.Enabled = False
    cmdexit.Enabled = True
    cmdnext.Enabled = True
    cmdlast.Enabled = True
    cmdpre.Enabled = True
    cmdFind.Enabled = True
    cmdfirst.Enabled = True
    cmdsave.Enabled = True
    Recordsource.Fields(0) = txtpatid
```

```
Recordsource.Fields(1) = txtregis
Recordsource.Fields(2) = txtfirstname
Recordsource.Fields(3) = txtlastname
Recordsource.Fields(4) = txtoccu
If txtrefper.Text <> "" Then
    Recordsource.Fields(5) = txtrefper
Else
    Recordsource.Fields(5) = " "
End If
```

```
If txtcont.Text <> "" Then
    Recordsource.Fields(6) = txtcont
Else
    Recordsource.Fields(6) = " "
End If
```

```
If Opt_male.Value = True Then
    Recordsource.Fields(7) = "M"
ElseIf
    Opt_female.Value = True Then
    Recordsource.Fields(7) = "F"
End If
```

```
If Opt_Married.Value = True Then
    Recordsource.Fields(8) = "M"
ElseIf
    Opt_unMarried.Value = True Then
    Recordsource.Fields(8) = "U"
End If
Recordsource.Fields(9) = txtdob
```

```

Recordsource.Fields(10) = txtage
Recordsource.Fields(11) = txtblood
Recordsource.Fields(12) = txtadd1
Recordsource.Fields(13) = txtadd2
Recordsource.Fields(14) = txtpin
Recordsource.Fields(15) = txtcont
Recordsource.Fields(16) = txtward
Recordsource.Fields(17) = txtreg
Recordsource.Update
Recordsource.MoveLast
MsgBox "Record Has Been Saved Successfully", vbInformation + vbOKOnly, "Save"
Clear_text
Exit Sub
down:
MsgBox Err.Description
End Sub

Private Function Show_Record()
    Dim abc As String
    txtpatid = Recordsource.Fields(0)
    txtregis = Recordsource.Fields(1)
    txtfirstname = Recordsource.Fields(2)
    txtlastname = Recordsource.Fields(3)
    txtoccu = Recordsource.Fields(4)
    If Recordsource.Fields(5) <> "" Then
        txtrefper.Text = Recordsource.Fields(5)
    Else
        txtAddress2.Text = ""
    End If
    If Recordsource.Fields(6) <> "" Then

```

```
txtref_cont.Text = Recordsource.Fields(6)

Else

    txtref_cont.Text = ""

End If

If Recordsource.Fields(7) = "M" Then
    Opt_male.Value = True
ElseIf
    Recordsource.Fields(7) = "F" Then
    Opt_female.Value = True
End If

If Recordsource.Fields(8) = "M" Then
    Opt_Married.Value = True
ElseIf
    Recordsource.Fields(8) = "U" Then
    Opt_unMarried.Value = True
End If

txtdob.Text = Recordsource.Fields(9)
txtage = Recordsource.Fields(10)
txtblood = Recordsource.Fields(11)
txtadd1 = Recordsource.Fields(12)
txtadd2.Text = Recordsource.Fields(13)
txtpin = Recordsource.Fields(14)
txtcont = Recordsource.Fields(15)
txtward = Recordsource.Fields(16)
txtreg = Recordsource.Fields(17)

End Function
```

FORM-5 CONSULTANT INFORMATION

Dim con As New ADODB.Connection

Dim Recordsource As New ADODB.Recordset

Private Sub cmdadd_Click()

On Error GoTo adderr

cmdadd.Enabled = False

cmddelete.Enabled = False

cmdmodify.Enabled = False

cmdexit.Enabled = False

cmdnext.Enabled = False

cmdlast.Enabled = False

cmdpre.Enabled = False

cmdFind.Enabled = False

cmdfirst.Enabled = False

cmdsave.Enabled = True

Clear_text

If Not Recordsource.BOF = True Then

Recordsource.MoveLast

MsgBox "TXT" & Recordsource.Fields(0)

txtpatid.Text = Val(Recordsource.Fields(0)) + 1

Else

txtpatid.Text = 101

End If

If Recordsource.BOF <> True or Recordsource.EOF <> True Then

Recordsource.MoveLast

Recordsource.MoveNext

End If

Recordsource.AddNew

cmdsave.Enabled = True

```
Exit Sub
adderr:
MsgBox Err.Description
End Sub

Private Sub cmdback_Click()
Patient_details.Hide
Main_menu.Show
Main_menu.WindowState = 2
End Sub

Private Function Clear_text()
txtpatid = ""
txtrejis = ""
txtfirstname = ""
txtlastname = ""
txtoccu = ""
txtrefper = ""
txtcont = ""
txtdob = ""
txtage = ""
txtblood = ""
txtadd1 = ""
txtadd2 = ""
txtpin = ""
txtref_cont = ""
txtward = ""
txtreg = ""
End Function
```

```
Private Sub cmdfind_Click()
    Dim id, i, a, current As Integer
    i = 0
    id = InputBox("Please enter a Patient id:- ")
    Recordsource.MoveFirst
Do While Not Recordsource.EOF
    i = i + 1
    Recordsource.MoveNext
Loop
    Recordsource.MoveFirst
    For a = 1 To i
        current = Recordsource.Fields(0)
        If current = id Then
            Show_Record
        Exit Sub
        End If
        Recordsource.MoveNext
    Next
    MsgBox ("Recourd not found")
End Sub
```

```
Private Sub cmdmodify_Click()
    On Error GoTo down
    cmdadd.Enabled = False
    cmddelete.Enabled = False
    cmdmodify.Enabled = False
    cmdexit.Enabled = False
    cmdnext.Enabled = False
    cmdlast.Enabled = False
    cmdpre.Enabled = False
```



```
cmdFind.Enabled = False
cmdfirst.Enabled = False
cmdsave.Enabled = True
Recordsource.UpdateBatch adAffectAllChapters
Recordsource.UpdateBatch adAffectCurrent
Exit Sub
down:
MsgBox Err.Description
End Sub

Private Sub cmdpre_Click()
    Recordsource.MoveFirst
    Show_Record
End Sub

Private Sub Form_Load()
    con.Open "dsn=udsn;user id=scott;Password=tiger"
    Recordsource.Open "select * from patient", con, adOpenDynamic, adLockOptimistic
End Sub

Private Sub cmd_Last_Click()
    Recordsource.MoveLast
    Show_Record
End Sub

Private Sub cmd_Next_Click()
    Recordsource.MoveNext
    Show_Record
End Sub
```

```
Private Sub cmdexit_Click()
    Patient_details.Hide
    Clear_text
    Main_menu.Show
    Main_menu.WindowState = 2
End Sub

Private Sub cmdfirst_Click()
    Recordsource.CancelUpdate
    Recordsource.MoveFirst
    Show_Record
End Sub

Private Sub cmdlast_Click()
    Recordsource.CancelUpdate
    Recordsource.MoveLast
    Show_Record
End Sub

Private Sub cmdnext_Click()
    Recordsource.CancelUpdate
    Recordsource.MoveNext
    If Recordsource.EOF Then
        Recordsource.MoveLast
    End If
    Show_Record
End Sub

Private Sub CmdPrior_Click()
    Recordsource.CancelUpdate
```

```
Recordsource.MovePrevious
If Recordsource.BOF Then
    Recordsource.MoveFirst
End If

Show_Record

End Sub

Private Sub CmdSave_Click()
    On Error GoTo down
    cmdadd.Enabled = True
    cmddelete.Enabled = True
    cmdmodify.Enabled = False
    cmdexit.Enabled = True
    cmdnext.Enabled = True
    cmdlast.Enabled = True
    cmdpre.Enabled = True
    cmdFind.Enabled = True
    cmdfirst.Enabled = True
    cmdsave.Enabled = True
    Recordsource.Fields(0) = txtpatid
    Recordsource.Fields(1) = txtregis
    Recordsource.Fields(2) = txtfirstname
    Recordsource.Fields(3) = txtlastname
    Recordsource.Fields(4) = txtoccu
    If txtrefper.Text <> "" Then
        Recordsource.Fields(5) = txtrefper
    Else
        Recordsource.Fields(5) = " "
    End If
    If txtcont.Text <> "" Then
        Recordsource.Fields(6) = txtcont
```

```
Else
    Recordsource.Fields(6) = " "
End If
If Opt_male.Value = True Then
    Recordsource.Fields(7) = "M"
ElseIf Opt_female.Value = True Then
    Recordsource.Fields(7) = "F"
End If

If Opt_Married.Value = True Then
    Recordsource.Fields(8) = "M"
ElseIf
    Opt_unMarried.Value = True Then
    Recordsource.Fields(8) = "U"
End If

Recordsource.Fields(9) = txtdob
Recordsource.Fields(10) = txtage
Recordsource.Fields(11) = txtblood
Recordsource.Fields(12) = txtadd1
Recordsource.Fields(13) = txtadd2
Recordsource.Fields(14) = txtpin
Recordsource.Fields(15) = txtcont
Recordsource.Fields(16) = txtward
Recordsource.Fields(17) = txtreg
Recordsource.Update
Recordsource.MoveLast
MsgBox "Record Has Been Saved Successfully", vbInformation +
vbOKOnly, "Save"
Clear_text
Exit Sub
```

down:

MsgBox Err.Description

End Sub

Private Function Show_Record()

Dim abc As String

txtpatid = Recordsource.Fields(0)

txtregis = Recordsource.Fields(1)

txtfirstname = Recordsource.Fields(2)

txtlastname = Recordsource.Fields(3)

txtoccu = Recordsource.Fields(4)

If Recordsource.Fields(5) <> "" Then

txtrefper.Text = Recordsource.Fields(5)

Else

txtAddress2.Text = ""

End If

If Recordsource.Fields(6) <> "" Then

txtref_cont.Text = Recordsource.Fields(6)

Else

txtref_cont.Text = ""

End If

If Recordsource.Fields(7) = "M" Then

Opt_male.Value = True

ElseIf

Recordsource.Fields(7) = "F" Then

Opt_female.Value = True

End If

If Recordsource.Fields(8) = "M" Then

```
Opt_Married.Value = True
ElseIf
    Recordsource.Fields(8) = "U" Then
    Opt_unMarried.Value = True
End If

txtdob.Text = Recordsource.Fields(9)
txtage = Recordsource.Fields(10)
txtblood = Recordsource.Fields(11)
txtadd1 = Recordsource.Fields(12)
txtadd2.Text = Recordsource.Fields(13)
txtpin = Recordsource.Fields(14)
txtcont = Recordsource.Fields(15)
txtward = Recordsource.Fields(16)
txtreg = Recordsource.Fields(17)
End Function
```

FORM-6 DEPARTMENT INFORMATION

```
Dim con As New ADODB.Connection
Dim Recordsource As New ADODB.Recordset
Private Sub cmdadd_Click()
    On Error GoTo adderr
    cmdadd.Enabled = False
    cmdmodify.Enabled = False
    cmdexit.Enabled = False
    cmdnext.Enabled = False
    cmdlast.Enabled = False
    cmdpre.Enabled = False
    cmdFind.Enabled = False
    cmdfirst.Enabled = False
    cmdsave.Enabled = True
    Clear_text
    If Not Recordsource.BOF = True Then
        Recordsource.MoveLast
        MsgBox "TXT" & Recordsource.Fields(0)
        txtno.Text = Val(Recordsource.Fields(0)) + 1
    Else
        txtno.Text = 101
    End If
    If Recordsource.BOF <> True Or Recordsource.EOF <> True Then
        MsgBox "ADD" & Recordsource.Fields(0)
        Recordsource.MoveLast
        MsgBox "LAST" & Recordsource.Fields(0)
        Recordsource.MoveNext
    End If
```

```
        Recordsource.AddNew
        cmdsave.Enabled = True
    Exit Sub
adderr:
    MsgBox Err.Description
End Sub

Private Sub cmdexit_Click()
    Recordsource.Close
    Unload Me
    Main_menu.Show
    Main_menu.WindowState = 2
End Sub

Private Sub cmdmodify_Click()
    On Error GoTo down
    cmdadd.Enabled = False
    cmdmodify.Enabled = False
    cmdexit.Enabled = False
    cmdnext.Enabled = False
    cmdlast.Enabled = False
    cmdpre.Enabled = False
    cmdFind.Enabled = False
    cmdfirst.Enabled = False
    cmdsave.Enabled = True
    Recordsource.UpdateBatch adAffectAllChapters
    Recordsource.UpdateBatch adAffectCurrent
    Exit Sub
down:
    MsgBox Err.Description
```


End Sub

Private Sub cmdpre_Click()

Recordsource.CancelUpdate

Recordsource.MovePrevious

If Recordsource.BOF Then

MsgBox "You are viewing First Record", vbCritical, "Employee
Information Report"

Recordsource.MoveFirst

End If

Show_Record

End Sub

Private Sub CmdSave_Click()

On Error GoTo down

cmdadd.Enabled = True

cmdexit.Enabled = True

cmdnext.Enabled = True

cmdlast.Enabled = True

cmdmodify.Enabled = True

cmdpre.Enabled = True

cmdFind.Enabled = True

cmdfirst.Enabled = True

cmdsave.Enabled = True

Recordsource.Fields(0) = txtno

Recordsource.Fields(1) = txtlname

Recordsource.Fields(2) = txtdesc

Recordsource.Fields(3) = txtthead

Recordsource.Fields(4) = txtadd1

Recordsource.Fields(5) = txtadd2

```

Recordsource.Fields(6) = txttel
Recordsource.Update
MsgBox "Record Has Been Saved Successfully", vbInformation + vbOKOnly, "Save"
Clear_text
Exit Sub
down:
MsgBox Err.Description
End Sub

Private Sub Form_Load()
    con.Open "dsn=udsn;user id=scott;Password=tiger"
    Recordsource.Open "select * from dept", con, adOpenDynamic, adLockOptimistic
End Sub

Private Sub cmddelete_Click()
    On Error GoTo down
    If MsgBox("Do you Want To Delete Cureent Record", vbYesNo + vbInformation,
        "Delete") = vbYes Then
        If Recordsource.EOF = True Or Recordsource.BOF = True Then
            MsgBox "No current record", vbCritical, "Delete"
        End If
        Exit Sub
    End If
    Recordsource.Delete adAffectCurrent
    Clear_text
    MsgBox "Record deleted", vbCritical, "Delete Hospital."
    Recordsource.Update
    Recordsource.MoveNext
    cmddelete.Enabled = False
    Exit Sub
End If

```

```
Show_Record  
Exit Sub  
down:  
MsgBox Err.Description, vbOKOnly + vbCritical  
End Sub
```

```
Private Sub cmdfind_Click()  
    Dim id, i, a, current As Integer  
    i = 0  
    id = InputBox("Please enter Department id:- ")  
    Recordsource.MoveFirst  
    Do While Not Recordsource.EOF  
        i = i + 1  
        Recordsource.MoveNext  
    Loop  
    Recordsource.MoveFirst  
    For a = 1 To i  
        current = Recordsource.Fields(0)  
        If current = id Then  
            Show_Record  
        End If  
        Recordsource.MoveNext  
    Next  
    MsgBox ("Department Record Not Found.....")  
End Sub
```

```
Private Sub cmdfirst_Click()  
    Recordsource.MoveFirst  
    Show_Record
```

```
If Recordsource.BOF = True Then
    MsgBox "You are viwing first record", vbCritical, "Employee
    Information Report"
End If
End Sub

Private Sub cmdlast_Click()
    Recordsource.CancelUpdate
    Recordsource.MoveLast
    If Recordsource.EOF Then
        MsgBox "You are viewing last Record", vbCritical, "Employee Information
        Report"
        Recordsource.MoveLast
    End If
    Show_Record
End Sub

Private Sub cmdnext_Click()
    Recordsource.CancelUpdate
    Recordsource.MoveNext
    If Recordsource.EOF Then
        MsgBox "You are viewing last Record", vbCritical, "Employee Information
        Report"
        Recordsource.MoveLast
    End If
    Show_Record
End Sub

Private Function Clear_text()
    txtno = ""
    txtlname = ""
```

```
txtdesc = ""  
txthead = ""  
txtadd1 = ""  
txtadd2 = ""  
txttel = ""
```

End Function

Private Function Show_Record()

```
txtno = Recordsource.Fields(0)  
txtdname = Recordsource.Fields(1)  
txtdesc = Recordsource.Fields(2)  
txthead = Recordsource.Fields(3)  
txtadd1 = Recordsource.Fields(4)  
txtadd2 = Recordsource.Fields(5)  
txttel = Recordsource.Fields(6)
```

End Function

FROM-7 STAFF INFORMATION

```
Dim con As New ADODB.Connection
```

```
Dim Recordsource As New ADODB.Recordset
```

```
Private Sub cmdback_Click()
```

```
    Unload Me
```

```
    Main_menu.Show
```

```
    Main_menu.WindowState = 2
```

```
End Sub
```

```
Private Sub cmdcalculate_Click()
```

```
    If txtbasic_sal.Text <= 3000 Then
```

```
        txtta.Text = txtbasic_sal * 0.05
```

```
        txtta.Text = txtbasic_sal * 0.8
```

```
        txthra.Text = txtbasic_sal * 0.03
```

```
        txtma.Text = txtbasic_sal * 0.03
```

```
        txtpf.Text = txtbasic_sal * 0.02
```

```
        txtgs.Text = Val(txtta) + Val(txtta) + Val(txthra) + Val(txtma) +
```

```
        Val(txtpf) + Val(txtbasic_sal)
```

```
        txtnet.Text = Val(txtta) + Val(txtta) + Val(txthra) + Val(txtma) +
```

```
        Val(txtbasic_sal) - Val(txtpf)
```

```
    End If
```

```
    If txtbasic_sal.Text > 3000 Or txtbasic_sal.Text <= 6000 Then
```

```
        txtta.Text = txtbasic_sal * 0.1
```

```
        txtta.Text = txtbasic_sal * 0.9
```

```
        txthra.Text = txtbasic_sal * 0.06
```

```
        txtma.Text = txtbasic_sal * 0.06
```

```
        txtpf.Text = txtbasic_sal * 0.04
```

```

txtgs.Text = Val(txtta) + Val(txttda) + Val(txthra) + Val(txtma) + Val(txtpf) +
Val(txtbasic_sal)
txtnet.Text = Val(txtta) + Val(txttda) + Val(txthra) + Val(txtma) + Val(txtbasic_sal) -
Val(txtpf)

```

```
End If
```

```
End Sub
```

```
Private Sub cmddelete_Click()
```

```
    On Error GoTo down
```

```
    If MsgBox("Do you Want To Delete Cureent Record", vbYesNo +
vbInformation, "Delete") = vbYes Then
```

```
        If Recordsource.EOF = True Or Recordsource.BOF = True Then
            MsgBox "No current record", vbCritical, "Delete"

```

```
Exit Sub
```

```
    End If
```

```
    Recordsource.Delete adAffectCurrent
```

```
    Clear_text
```

```
    MsgBox "Record deleted", vbCritical, "Delete Hospital."
```

```
    Recordsource.Update
```

```
    Recordsource.MoveNext
```

```
    cmddelete.Enabled = False
```

```
Exit Sub
```

```
End If
```

```
    Show_Record
```

```
Exit Sub
```

```
down:
```

```
MsgBox Err.Description, vbOKOnly + vbCritical
```

```
End Sub
```

```
Private Sub cmdfind_Click()
```

```
Dim emp, i, a, current As Integer
i = 0
emp = InputBox("Please enter Depositer_no:- ")
Recordsource.MoveFirst
Do While Not Recordsource.EOF
    i = i + 1
    Recordsource.MoveNext
Loop
Recordsource.MoveFirst
For a = 1 To i
    current = Recordsource.Fields(0)
    If current = emp Then
        Show_Record
    End If
    Recordsource.MoveNext
Next
MsgBox ("Employee Record Not Found.....")
End Sub

Private Sub cmdfirst_Click()
    Recordsource.MoveFirst
    Show_Record
    If Recordsource.BOF = True Then
        MsgBox "You are viwing first record", vbCritical, "Employee Information Report"
    End If
End Sub

Private Sub cmdinsert_Click()
```



```
On Error GoTo adderr
Clear_text
If Not Recordsource.BOF = True Then
    Recordsource.MoveLast
    MsgBox "TXT" & Recordsource.Fields(0)
    txtcode.Text = Val(Recordsource.Fields(0)) + 1
Else
    txtcode.Text = 101
End If
If Recordsource.BOF <> True Or Recordsource.EOF <> True Then
    Recordsource.MoveLast
    Recordsource.MoveNext
End If
Recordsource.AddNew
cmdsave.Enabled = True
Exit Sub
adderr:
MsgBox Err.Description
End Sub

Private Sub cmdlast_Click()
    Recordsource.CancelUpdate
    Recordsource.MoveLast
    If Recordsource.EOF Then
        MsgBox "You are viewing last Record", vbCritical, "Employee Information Report"
        Recordsource.MoveLast
    End If
    Show_Record
End Sub
```

```
Private Sub cmdmain_Click()  
    Unload Me  
    Main_menu.Show  
    Main_menu.WindowState = 2  
End Sub
```

```
Private Sub cmdnext_Click()  
    Recordsource.CancelUpdate  
    Recordsource.MoveNext  
If Recordsource.EOF Then  
    MsgBox "You are viewing last Record", vbCritical, "Employee Information Report"  
    Recordsource.MoveLast  
End If  
Show_Record  
End Sub
```

```
Private Sub cmdprevious_Click()  
    Recordsource.CancelUpdate  
    Recordsource.MovePrevious  
If Recordsource.BOF Then  
    MsgBox "You are viewing First Record", vbCritical, "Employee Information  
        Report"  
    Recordsource.MoveFirst  
End If  
    Show_Record  
End Sub
```

```
Private Sub CmdSave_Click()  
    On Error GoTo down
```

```
Recordsource.Fields(0) = txtcode
Recordsource.Fields(1) = txttype
Recordsource.Fields(2) = txtfirstname
Recordsource.Fields(3) = txtlastname
Recordsource.Fields(4) = txtadd
Recordsource.Fields(5) = txtdesignation
Recordsource.Fields(6) = txtph
Recordsource.Fields(7) = txtsex
Recordsource.Fields(8) = txtbasic_sal
Recordsource.Fields(9) = txtdob
Recordsource.Fields(10) = txt DOJ
Recordsource.Fields(11) = txt_dor
Recordsource.Update
MsgBox "Record Has Been Saved Successfully", vbInformation + vbOKOnly, "Save"
Clear_text

Exit Sub
down:
MsgBox Err.Description
End Sub

Private Sub Form_Load()
    con.Open "dsn=udsn;user id=scott;Password=tiger"
    Recordsource.Open "select * from emp", con, adOpenDynamic, adLockOptimistic
End Sub

Private Function Clear_text()
    txtcode = ""
    txttype = ""
    txtfirstname = ""
    txtlastname = ""
```

```
txtdesignation = ""  
txtbasic_sal = ""  
txtdob = ""  
txt doj = ""  
txt_dor = ""  
txt da = ""  
txt ta = ""  
tx thra = ""  
txt ma = ""  
txt pf = ""  
txt gs = ""  
txt net = ""  
txt add = ""  
txt ph = ""  
txt sex = ""
```

End Function

Private Function Show_Record()

```
txtcode = Recordsource.Fields(0)  
txttype = Recordsource.Fields(1)  
txtfirstname = Recordsource.Fields(2)  
txtlastname = Recordsource.Fields(3)  
txtadd = Recordsource.Fields(4)  
txtdesignation = Recordsource.Fields(5)  
txtph = Recordsource.Fields(6)  
txtsex = Recordsource.Fields(7)  
txtbasic_sal = Recordsource.Fields(8)  
txtdob = Recordsource.Fields(9)  
txt doj = Recordsource.Fields(10)  
txt_dor = Recordsource.Fields(11)
```

End Function

FORM – 8 LABORATORY INFORMATION

Dim con As New ADODB.Connection

Dim Recordsource As New ADODB.Recordset

Private Sub cmdadd_Click()

 On Error GoTo adderr

 cmdadd.Enabled = False

 cmddelete.Enabled = False

 cmdmodify.Enabled = False

 cmdexit.Enabled = False

 cmdnext.Enabled = False

 cmdlast.Enabled = False

 cmdpre.Enabled = False

 cmdFind.Enabled = False

 cmdfirst.Enabled = False

 cmdsave.Enabled = True

 If Not Recordsource.BOF = True Then

 Recordsource.MoveLast

 MsgBox "TXT" & Recordsource.Fields(0)

 txtpid.Text = Val(Recordsource.Fields(0)) + 1

 Else

 txtpid.Text = 201

 End If

 If Recordsource.BOF <> True Or Recordsource.EOF <> True Then

 MsgBox "ADD" & Recordsource.Fields(0)

 Recordsource.MoveLast

 MsgBo x "LAST" & Recordsource.Fields(0)

```
        Recordsource.MoveNext
    End If
        Recordsource.AddNew
        cmdsave.Enabled = True
    Exit Sub
adderr:
    MsgBox Err.Description
End Sub

Private Sub cmdexit_Click()
    Consultant_Details
    Unload Me
    Main_menu.Show
    Main_menu.WindowState = 2
End Sub

Private Sub cmdmodify_Click()
    On Error GoTo down
    cmdadd.Enabled = False
    cmddelete.Enabled = False
    cmdexit.Enabled = False
    cmdnext.Enabled = False
    cmdlast.Enabled = False
    cmdpre.Enabled = False
    cmdFind.Enabled = False
    cmdfirst.Enabled = False
    cmdsave.Enabled = True
    Recordsource.UpdateBatch adAffectAllChapters
    Recordsource.UpdateBatch adAffectCurrent
    Exit Sub
down:
```

```
        down:
        MsgBox Err.Description
End Sub

Private Sub cmdpre_Click()
    Recordsource.CancelUpdate
    Recordsource.MovePrevious
    If Recordsource.BOF Then
        MsgBox "You are viewing First Record", vbCritical, "Employee Information
        Report"
        Recordsource.MoveFirst
    End If
    Show_Record
End Sub

Private Sub CmdSave_Click()
    On Error GoTo down
    cmdadd.Enabled = True
    cmddelete.Enabled = True
    cmdexit.Enabled = True
    cmdnext.Enabled = True
    cmdlast.Enabled = True
    cmdpre.Enabled = True
    cmdFind.Enabled = True
    cmdfirst.Enabled = True
    cmdsave.Enabled = True
    Recordsource.Fields(0) = txtpid
    Recordsource.Fields(1) = txtpname
    Recordsource.Fields(2) = txtrefby
    Recordsource.Fields(3) = txtexam
```

```

Recordsource.Fields(4) = txtdate
Recordsource.Fields(5) = txtby
Recordsource.Fields(6) = txtrepo1
Recordsource.Fields(7) = txtrepo2
Recordsource.Update
MsgBox "Record Has Been Saved Successfully", vbInformation + vbOKOnly, "Save"
Clear_text
Exit Sub
down:
MsgBox Err.Description
End Sub

Private Sub Form_Load()
    con.Open "dsn=udsn;user id= scott;Password=tiger "
    Recordsource.Open "select * from report", con, adOpenDynamic, adLockOptimistic
End Sub

Private Sub cmddelete_Click()
    On Error GoTo down
    If MsgBox("Do you Want To Delete Cureent Record", vbYesNo + vbInformation,
        "Delete") = vbYes Then
        If Recordsource.EOF = True Or Recordsource.BOF = True Then
            MsgBox "No current record", vbCritical, "Delete"
        End If
        Exit Sub
    End If
    Recordsource.Delete adAffectCurrent
    Clear_text
    MsgBox "Record deleted", vbCritical, "Delete L.I.C."
    Recordsource.Update
    Recordsource.MoveNext

```



```
        cmddelete.Enabled = False
    Exit Sub
End If
Show_Record
Exit Sub
down:
MsgBox Err.Description, vbOKOnly + vbCritical
End Sub

Private Sub cmdfind_Click()
    Dim id, i, a, current As Integer
    i = 0
    id = InputBox("Please enter Patient Code:- ")
    Recordsource.MoveFirst
    Do While Not Recordsource.EOF
        i = i + 1
        Recordsource.MoveNext
    Loop
    Recordsource.MoveFirst
    For a = 1 To i
        current = Recordsource.Fields(0)
        If current = id Then
            Show_Record
        Exit Sub
    End If
    Recordsource.MoveNext
Next
MsgBox ("Medicine Not Found.....")
End Sub
```

```
Private Sub cmdfirst_Click()  
    Recordsource.MoveFirst  
    Show_Record  
    If Recordsource.BOF = True Then  
        MsgBox "You are viwing first record", vbCritical, "Employee Information  
        Report"  
    End If  
End Sub
```

```
Private Sub cmdlast_Click()  
    Recordsource.CancelUpdate  
    Recordsource.MoveLast  
    If Recordsource.EOF Then  
        MsgBox "You are viewing last Record", vbCritical, "Employee Information  
        Report"  
        Recordsource.MoveLast  
    End If  
    Show_Record  
End Sub
```

```
Private Sub cmdmain_Click()  
    Unload Me  
    Main_menu.Show  
    Main_menu.WindowState = 2  
End Sub
```

```
Private Sub cmdnext_Click()  
    Recordsource.CancelUpdate  
    Recordsource.MoveNext  
    If Recordsource.EOF Then
```

```
MsgBox "You are viewing last Record", vbCritical, "Employee Information  
Report"  
Recordsource.MoveLast  
End If  
Show_Record  
End Sub
```

```
Private Function Clear_text()
```

```
txtpid = ""  
txtpname = ""  
txtrefby = ""  
txtexam = ""  
txtdate = ""  
txtby = ""  
txtrepo1 = ""  
txtrepo2 = ""
```

```
End Function
```

```
Private Function Show_Record()
```

```
txtpid = Recordsource.Fields(0)  
txtpname = Recordsource.Fields(1)  
txtrefby = Recordsource.Fields(2)  
txtexam = Recordsource.Fields(3)  
txtdate = Recordsource.Fields(4)  
txtby = Recordsource.Fields(5)  
txtrepo1 = Recordsource.Fields(6)  
txtrepo2 = Recordsource.Fields(7)
```

```
End Function
```

FORM-9 SEARCH BILLING INFORMATION

```
Dim con As New ADODB.Connection
Dim Recordsource As New ADODB.Recordset
Private Sub cmdback_Click()
    Unload Me
    Main_menu.Show
    Main_menu.WindowState = 2
End Sub
```

```
Private Sub cmdnew_Click()
    Unload Me
    Transaction_Details.Show
    Transaction_Details.WindowState = 2
End Sub
```

```
Private Sub cmdRefr_Click()
    Text1.Text = ""
    Text1.SetFocus
End Sub
```

```
Private Sub cmdsea_Click()
    Dim id, i, a, current As Integer
    i = 0
    id = Text1.Text
    Recordsource.MoveFirst
    Do While Not Recordsource.EOF
        i = i + 1
        Recordsource.MoveNext
```

```
Loop
    Recordsource.MoveFirst
    For a = 1 To i
        current = Recordsource.Fields(0)
        If current = id Then
            Transaction_Details.Show
            With Transaction_Details
                .txtid = Recordsource.Fields(0)
                .txtfirstname = Recordsource.Fields(2)
                .txtlastname = Recordsource.Fields(3)
                .txtAddress1 = Recordsource.Fields(12)
                .txtAddress2 = Recordsource.Fields(13)
                .txtPinCode = Recordsource.Fields(15)
                .txtward = Recordsource.Fields(16)
                .txtreg = Recordsource.Fields(17)
            End With
        End If
        Exit Sub
    Next
    Recordsource.MoveNext
    'End If
    MsgBox ("Recourd not found")
End Sub

Private Sub Form_Load()
    con.Open "dsn=udsn;user id= scott;Password=tiger "
    Recordsource.Open "select * from patient", con, adOpenDynamic, adLockOptimistic
End Sub
```

FORM-10 BILLING INFORMATION

```
Dim con As New ADODB.Connection
Dim Recordsource As New ADODB.Recordset
Private Sub cmdexit_Click()
    Unload Me
    search_Transaction.Show
    search_Transaction.WindowState = 2
End Sub
```

```
Private Sub cmdmain_Click()
    Unload Me
    Main_menu.WindowState = 2
    Main_menu.Show
End Sub
```

```
Private Sub CmdSave_Click()
    On Error GoTo down
    Recordsource.AddNew
    Recordsource.Fields(0) = txtdate
    Recordsource.Fields(1) = txtdc
    Recordsource.Fields(2) = txt_tra_mode
    Recordsource.Fields(3) = txttra_amount
    Recordsource.Fields(4) = txtverification
    Recordsource.Fields(5) = txtverified_by
    Recordsource.Fields(6) = txtid
    Recordsource.Fields(7) = txtreg
    Recordsource.Fields(8) = txtward
    Recordsource.Fields(9) = txtFirstName
```

```

Recordsource.Fields(10) = txtLastName
Recordsource.Fields(11) = txtAddress1
Recordsource.Fields(12) = txtAddress2
Recordsource.Fields(13) = txtPinCode
Recordsource.Update
MsgBox "Record Has Been Saved Successfully", vbInformation + vbOKOnly,
"Save"
Clear_text
Exit Sub
down:
MsgBox Err.Description
End Sub

Private Sub Form_Activate()
txtdate = Date
txtdate.Locked = True
txtFirstName.Locked = True
txtLastName.Locked = True
txtAddress1.Locked = True
txtAddress2.Locked = True
txtPinCode.Locked = True
End Sub

Private Sub Form_Load()
con.Open "dsn=udsn;user id= scott;Password=tiger "
Recordsource.Open "select * from trans", con, adOpenDynamic, adLockOptimistic
End Sub

Private Function Clear_text()
txtdate = ""

```

```
txt_tra_mode = ""  
txttra_amount = ""  
txtverification = ""  
txtverified_by = ""  
txtid = ""  
txtFirstName = ""  
txtLastName = ""  
txtAddress1 = ""  
txtAddress2 = ""  
txtPinCode = ""  
txtward = ""  
txtreg = ""
```

End Function

```
Private Sub opt_cash_Click()  
    Label8.Visible = True  
    Label8.ForeColor = vbBlue  
    Label8.Caption = "Please Enter Voucher No:"
```

End Sub

```
Private Sub opt_checque_Click()  
    Label8.Visible = True  
    Label8.ForeColor = vbBlue  
    Label8.Caption = "Please Enter Cheque/Draft No:"
```

End Sub

```
Private Sub Opt_credit_Click()  
    txtdc.ForeColor = vbBlue  
    txtdc.FontBold = True  
    txtdc.Alignment = 2
```



```
txtdc.Text = "CREDIT"
```

```
End Sub
```

```
Private Sub Opt_debit_Click()
```

```
txtdc.ForeColor = vbBlue
```

```
txtdc.FontBold = True
```

```
txtdc.Alignment = 2
```

```
txtdc.Text = "DEBIT"
```

```
End Sub
```

FORM – 11 MEDICINE INFORMATION

Dim con As New ADODB.Connection

Dim Recordsource As New ADODB.Recordset

Private Sub cmdadd_Click()

On Error GoTo adderr

cmdadd.Enabled = False

cmddelete.Enabled = False

cmdmodify.Enabled = False

cmdexit.Enabled = False

cmdnext.Enabled = False

cmdlast.Enabled = False

cmdpre.Enabled = False

cmdFind.Enabled = False

cmdfirst.Enabled = False

cmdsave.Enabled = True

Clear_text

If Not Recordsource.BOF = True Then

Recordsource.MoveLast

MsgBox "TXT" & Recordsource.Fields(0)

txtmcode.Text = Val(Recordsource.Fields(0)) + 1

Else

txtmcode.Text = 101

End If

If Recordsource.BOF <> True Or Recordsource.EOF <> True then

Recordsource.MoveLast

Recordsource.MoveNext

End If

Recordsource.AddNew

cmdsave.Enabled = True

```
Exit Sub
adderr:
MsgBox Err.Description
End Sub

Private Sub cmdexit_Click()
Unload Me
Main_menu.Show
Main_menu.WindowState = 2
End Sub

Private Sub cmdmodify_Click()
On Error GoTo down
cmdadd.Enabled = False
cmddelete.Enabled = False
cmdexit.Enabled = False
cmdnext.Enabled = False
cmdlast.Enabled = False
cmdpre.Enabled = False
cmdFind.Enabled = False
cmdfirst.Enabled = False
cmdsave.Enabled = True
Recordsource.UpdateBatch adAffectAllChapters
Recordsource.UpdateBatch adAffectCurrent
Exit Sub
down:
MsgBox Err.Description
End Sub

Private Sub cmdpre_Click()
```

```
Recordsource.CancelUpdate
Recordsource.MovePrevious
If Recordsource.BOF Then
    MsgBox "You are viewing First Record", vbCritical, "Employee
    Information Report"
    Recordsource.MoveFirst
End If
Show_Record
End Sub

Private Sub CmdSave_Click()
    On Error GoTo down
    cmdadd.Enabled = True
    cmddelete.Enabled = True
    cmdexit.Enabled = True
    cmdnext.Enabled = True
    cmdlast.Enabled = True
    cmdpre.Enabled = True
    cmdFind.Enabled = True
    cmdfirst.Enabled = True
    cmdsave.Enabled = True
    Recordsource.Fields(0) = txtmcode
    Recordsource.Fields(1) = txtmname
    Recordsource.Fields(2) = txttype
    Recordsource.Fields(3) = txtmdate
    Recordsource.Fields(4) = txtxdate
    Recordsource.Fields(5) = txtmanu
    Recordsource.Fields(6) = txtbno
    Recordsource.Update
    Recordsource.MoveLast
```

```

    MsgBox "Record Has Been Saved Successfully", vbInformation + vbOKOnly, "Save"
    Clear_text
    Exit Sub
down:
    MsgBox Err.Description
End Sub

Private Sub Form_Load()
    con.Open "dsn=udsn;user id= scott;Password=tiger "
    Recordsource.Open "select * from medicine", con, adOpenDynamic,
    adLockOptimistic
End Sub

Private Sub cmddelete_Click()
    On Error GoTo down
    If MsgBox("Do you Want To Delete Cureent Record", vbYesNo + vbInformation,
        "Delete") = vbYes Then
        If Recordsource.EOF = True Or Recordsource.BOF = True Then
            MsgBox "No current record", vbCritical, "Delete"
        End If
    End If
    Recordsource.Delete adAffectCurrent
    Clear_text
    MsgBox "Record deleted", vbCritical, "Delete L.I.C."
    Recordsource.Update
    Recordsource.MoveNext
    cmddelete.Enabled = False
    Exit Sub
End If
Show_Record

```

```
Exit Sub
down:
MsgBox Err.Description, vbOKOnly + vbCritical
End Sub
```

```
Private Sub cmdfind_Click()
    Dim id, i, a, current As Integer
    i = 0
    id = InputBox("Please enter Medicine Code:- ")
    Recordsource.MoveFirst
    Do While Not Recordsource.EOF
        i = i + 1
        Recordsource.MoveNext
    Loop
    Recordsource.MoveFirst
    For a = 1 To i
        current = Recordsource.Fields(0)
        If current = id Then
            Show_Record
            Exit Sub
        End If
        Recordsource.MoveNext
    Next
    MsgBox ("Medicine Not Found.....")
End Sub
```

```
Private Sub cmdfirst_Click()
    Recordsource.MoveFirst
    Show_Record
    If Recordsource.BOF = True Then
```

```
MsgBox "You are viwing first record", vbCritical, "Employee Information Report"
End If
End Sub

Private Sub cmdlast_Click()
    Recordsource.CancelUpdate
    Recordsource.MoveLast
    If Recordsource.EOF Then
        MsgBox "You are viewing last Record", vbCritical, "Employee Information
            Report"
        Recordsource.MoveLast
    End If
    Show_Record
End Sub

Private Sub cmdmain_Click()
    Unload Me
    Main_menu.Show
    Main_menu.WindowState = 2
End Sub

Private Sub cmdnext_Click()
    Recordsource.CancelUpdate
    Recordsource.MoveNext
    If Recordsource.EOF Then
        MsgBox "You are viewing last Record", vbCritical, "Employee Information
            Report"
        Recordsource.MoveLast
    End If
    Show_Record
```

End Sub

Private Function Clear_text()

txtmcode = ""

txtmname = ""

txttype = ""

txtmdate = ""

txtxdate = ""

txtmanu = ""

txtbno = ""

End Function

Private Function Show_Record()

txtmcode = Recordsource.Fields(0)

txtmname = Recordsource.Fields(1)

txttype = Recordsource.Fields(2)

txtmdate = Recordsource.Fields(3)

txtxdate = Recordsource.Fields(4)

txtmanu = Recordsource.Fields(5)

txtbno = Recordsource.Fields(6)

End Function

FORM – 12 PERCEPTION DETAILS

```
Dim con As New ADODB.Connection
Dim Recordsource As New ADODB.Recordset
Private Sub cmdadd_Click()
    On Error GoTo adderr
    cmdadd.Enabled = False
    cmddelete.Enabled = False
    cmdmodify.Enabled = False
    cmdexit.Enabled = False
    cmdnext.Enabled = False
    cmdlast.Enabled = False
    cmdpre.Enabled = False
    cmdfirst.Enabled = False
    cmdsave.Enabled = True
    If Recordsource.BOF <> True Or Recordsource.EOF <> True Then
        Recordsource.MoveLast
        Recordsource.MoveNext
    End If
    Recordsource.AddNew
    cmdsave.Enabled = True
Exit Sub
adderr:
    MsgBox Err.Description
End Sub

Private Sub cmdback_Click()
    Recordsource.Close
    Search_prescription.Show
```

```
Search_prescription.WindowState = 2
End Sub

Private Sub cmdexit_Click()
    Unload Me
    Main_menu.Show
    Main_menu.WindowState = 2
End Sub

Private Sub cmdmodify_Click()
    On Error GoTo down
    cmdadd.Enabled = False
    cmddelete.Enabled = False
    cmdexit.Enabled = False
    cmdnext.Enabled = False
    cmdlast.Enabled = False
    cmdpre.Enabled = False
    cmdFind.Enabled = False
    cmdfirst.Enabled = False
    cmdsave.Enabled = True
    Recordsource.UpdateBatch adAffectAllChapters
    Recordsource.UpdateBatch adAffectCurrent
    Exit Sub
down:
    MsgBox Err.Description
End Sub

Private Sub cmdpre_Click()
    Recordsource.CancelUpdate
    Recordsource.MovePrevious
```

```
If Recordsource.BOF Then
    MsgBox "You are viewing First Record", vbCritical, "Employee Information
        Report"
    Recordsource.MoveFirst
End If
    Show_Record
End Sub

Private Sub CmdSave_Click()
    On Error GoTo down
    cmdadd.Enabled = True
    cmddelete.Enabled = True
    cmdexit.Enabled = True
    cmdnext.Enabled = True
    cmdlast.Enabled = True
    cmdpre.Enabled = True
    cmdFind.Enabled = True
    cmdfirst.Enabled = True
    cmdsave.Enabled = True
    Recordsource.Fields(0) = txtlname
    Recordsource.Fields(1) = txtsdate
    Recordsource.Fields(2) = txtname
    Recordsource.Fields(3) = txtaddress
    Recordsource.Fields(4) = txtm1
    Recordsource.Fields(5) = txtm2
    Recordsource.Fields(6) = txtid
    Recordsource.Fields(7) = txtfirstname
    Recordsource.Fields(8) = txtlastname
    Recordsource.Fields(9) = txtage
    Recordsource.Fields(10) = txtblood
```

```

Recordsource.Update
MsgBox "Record Has Been Saved Successfully", vbInformation + vbOKOnly, "Save"
Clear_text
Exit Sub
down:
MsgBox Err.Description
End Sub

Private Sub Form_Activate()
    Consultant_Details.WindowState = 2
End Sub

Private Sub Form_Load()
    con.Open "dsn=udsn;user id= scott;Password=tiger "
    Recordsource.Open "select * from pres", con, adOpenDynamic, adLockOptimistic
End Sub

Private Sub cmddelete_Click()
    On Error GoTo down
    If MsgBox("Do you Want To Delete Cureent Record", vbYesNo + vbInformation,
        "Delete") = vbYes Then
        If Recordsource.EOF = True Or Recordsource.BOF = True Then
            MsgBox "No current record", vbCritical, "Delete"
        End If
        Exit Sub
    End If
    Recordsource.Delete adAffectCurrent
    Clear_text
    MsgBox "Record deleted", vbCritical, "Delete L.I.C."
    Recordsource.Update
    Recordsource.MoveNext

```

```
cmddelete.Enabled = False
Exit Sub
End If
Show_Record
Exit Sub
down:
MsgBox Err.Description, vbOKOnly + vbCritical
End Sub

Private Sub cmdfind_Click()
    Dim id, i, a, current As Integer
    i = 0
    id = InputBox("Please enter Patient Code:- ")
    Recordsource.MoveFirst
    Do While Not Recordsource.EOF
        i = i + 1
        Recordsource.MoveNext
    Loop
    Recordsource.MoveFirst
    For a = 1 To i
        current = Recordsource.Fields(0)
        If current = id Then
            Show_Record
        End If
        Recordsource.MoveNext
    Next
    MsgBox ("Medicine Not Found.....")
End Sub
```

```
Private Sub cmdfirst_Click()  
    Recordsource.MoveFirst  
    Show_Record  
    If Recordsource.BOF = True Then  
        MsgBox "You are viwing first record", vbCritical, "Employee Information  
        Report"  
    End If  
End Sub  
  
Private Sub cmdlast_Click()  
    Recordsource.CancelUpdate  
    Recordsource.MoveLast  
    If Recordsource.EOF Then  
        MsgBox "You are viewing last Record", vbCritical, "Employee Information  
        Report"  
        Recordsource.MoveLast  
    End If  
    Show_Record  
End Sub  
  
Private Sub cmdmain_Click()  
    Unload Me  
    Main_menu.Show  
    Main_menu.WindowState = 2  
End Sub  
  
Private Sub cmdnext_Click()  
    Recordsource.CancelUpdate  
    Recordsource.MoveNext  
    If Recordsource.EOF Then  
        MsgBox "You are viewing last Record", vbCritical, "Employee Information
```

```
Report"
Recordsource.MoveLast
End If
Show_Record
End Sub

Private Function Clear_text()
    txtlname = ""
    txtsdate = ""
    txtlna = ""
    txtlde = ""
    txtm1 = ""
    txtm2 = ""
    txtid = ""
    txtfirstname = ""
    txtlastname = ""
    txtage = ""
    txtblood = ""
End Function

Private Function Show_Record()
    txtlname = Recordsource.Fields(0)
    txtsdate = Recordsource.Fields(1)
    txtlna = Recordsource.Fields(2)
    txtlde = Recordsource.Fields(3)
    txtm1 = Recordsource.Fields(4)
    txtm2 = Recordsource.Fields(5)
    txtid = Recordsource.Fields(6)
    txtfirstname = Recordsource.Fields(7)
    txtlastname = Recordsource.Fields(8)
```

```
txtage = Recordsource.Fields(9)
```

```
txtblood = Recordsource.Fields(10)
```

```
End Function
```


FORM – 13 SCHEDULE DETAILS

```
Dim con As New ADODB.Connection
Dim Recordsource As New ADODB.Recordset
Private Sub Combo1_click()
    txtday.Text = Combo1.Text
End Sub
Private Sub Combo2_click()
    txtdesig.Text = Combo2.Text
End Sub

Private Sub Form_Activate()
    Combo1.AddItem "Monday"
    Combo1.AddItem "Tuesday"
    Combo1.AddItem "Wednesday"
    Combo1.AddItem "Thursday"
    Combo1.AddItem "Friday"
    Combo1.AddItem "Saturday"
    Combo1.AddItem "Sunday"
    Combo2.AddItem "Visiting Doctor"
    Combo2.AddItem "Resident Doctor"
    Combo2.AddItem "Nurse"
    Combo2.AddItem "Ward Boy"
    Combo2.AddItem "Dresser"
    Combo2.AddItem "Peon"
    Combo2.AddItem "Cleaner"
    Combo2.AddItem "Attendant"
End Sub
```

```
Private Sub Option1_Click()  
    txttime = "6 Am To 2 PM"  
End Sub
```

```
Private Sub Option2_Click()  
    txttime = "2 PM To 10 PM"  
End Sub
```

```
Private Sub Option3_Click()  
    txttime = "10 PM To 6 AM"  
End Sub
```

```
Private Sub cmdadd_Click()  
    On Error GoTo adderr  
    cmdadd.Enabled = False  
    cmddelete.Enabled = False  
    cmdmodify.Enabled = False  
    cmdexit.Enabled = False  
    cmdnext.Enabled = False  
    cmdlast.Enabled = False  
    cmdpre.Enabled = False  
    cmdFind.Enabled = False  
    cmdfirst.Enabled = False  
    cmdsave.Enabled = True
```

```
    If Recordsource.BOF <> True Or Recordsource.EOF <> True Then  
        Recordsource.MoveLast  
        Recordsource.MoveNext  
    End If  
    Recordsource.AddNew
```

```
        cmdsave.Enabled = True
    Exit Sub
    adderr:
    MsgBox Err.Description
End Sub

Private Sub cmdexit_Click()
    Unload Me
    Main_menu.Show
    Main_menu.WindowState = 2
End Sub

Private Sub cmdmodify_Click()
    On Error GoTo down
    cmdadd.Enabled = False
    cmddelete.Enabled = False
    cmdexit.Enabled = False
    cmdnext.Enabled = False
    cmdlast.Enabled = False
    cmdpre.Enabled = False
    cmdFind.Enabled = False
    cmdfirst.Enabled = False
    cmdsave.Enabled = True
    Recordsource.UpdateBatch adAffectAllChapters
    Recordsource.UpdateBatch adAffectCurrent
    Exit Sub
    down:
    MsgBox Err.Description
End Sub
```

```
Private Sub cmdpre_Click()  
    Recordsource.CancelUpdate  
    Recordsource.MovePrevious  
    If Recordsource.BOF Then  
        MsgBox "You are viewing First Record", vbCritical, "Employee Information  
        Report"  
        Recordsource.MoveFirst  
    End If  
    Show_Record  
End Sub  
  
Private Sub CmdSave_Click()  
    On Error GoTo down  
    cmdadd.Enabled = True  
    cmddelete.Enabled = True  
    cmdexit.Enabled = True  
    cmdnext.Enabled = True  
    cmdlast.Enabled = True  
    cmdpre.Enabled = True  
    cmdFind.Enabled = True  
    cmdfirst.Enabled = True  
    cmdsave.Enabled = True  
    Recordsource.Fields(0) = txtcode  
    Recordsource.Fields(1) = txtname  
    Recordsource.Fields(2) = txtdesig  
    Recordsource.Fields(3) = txtday  
    Recordsource.Fields(4) = txttime  
    Recordsource.Update  
    MsgBox "Record Has Been Saved Successfully", vbInformation + vbOKOnly, "Save"  
    Clear_text
```

```

Exit Sub

down:

MsgBox Err.Description

End Sub

Private Sub Form_Load()

con.Open "dsn=udsn;user id= scott;Password=tiger "

Recordsource.Open "select * from schedule", con, adOpenDynamic,
adLockOptimistic

End Sub

Private Sub cmddelete_Click()

On Error GoTo down

If MsgBox("Do you Want To Delete Cureent Record", vbYesNo + vbInformation,
"Delete") = vbYes Then

If Recordsource.EOF = True Or Recordsource.BOF = True Then

MsgBox "No current record", vbCritical, "Delete"

Exit Sub

End If

Recordsource.Delete adAffectCurrent

Clear_text

MsgBox "Record deleted", vbCritical, "Delete L.I.C."

Recordsource.Update

Recordsource.MoveNext

cmddelete.Enabled = False

Exit Sub

End If

Show_Record

Exit Sub

down:

```

```
MsgBox Err.Description, vbOKOnly + vbCritical  
End Sub
```

```
Private Sub cmdfind_Click()  
    Dim id, i, a, current As String  
    i = 0  
    id = InputBox("Please enter Consultant id:- ")  
    Recordsource.MoveFirst  
    Do While Not Recordsource.EOF  
        i = i + 1  
        Recordsource.MoveNext  
    Loop  
    Recordsource.MoveFirst  
    For a = 1 To i  
        current = Recordsource.Fields(0)  
        If current = id Then  
            Show_Record  
            Exit Sub  
        End If  
        Recordsource.MoveNext  
    Next  
    MsgBox ("Employee Record Not Found.....")  
End Sub
```

```
Private Sub cmdfirst_Click()  
    Recordsource.MoveFirst  
    Show_Record  
    If Recordsource.BOF = True Then  
        MsgBox "You are viwing first record", vbCritical, "Employee Information  
        Report"
```

```
End If
End Sub

Private Sub cmdlast_Click()
    Recordsource.CancelUpdate
    Recordsource.MoveLast
    If Recordsource.EOF Then
        MsgBox "You are viewing last Record", vbCritical, "Employee Information Report"
        Recordsource.MoveLast
    End If
    Show_Record
End Sub

Private Sub cmdmain_Click()
    Unload Me
    Main_menu.Show
    Main_menu.WindowState = 2
End Sub

Private Sub cmdnext_Click()
    Recordsource.CancelUpdate
    Recordsource.MoveNext
    If Recordsource.EOF Then
        MsgBox "You are viewing last Record", vbCritical, "Employee Information Report"
        Recordsource.MoveLast
    End If
    Show_Record
End Sub
```

Private Function Clear_text()

 txtecode = ""

 txtename = ""

 txtdesig = ""

 txttime = ""

 txtday = ""

End Function

Private Function Show_Record()

 txtecode = Recordsource.Fields(0)

 txtename = Recordsource.Fields(1)

 txtdesig = Recordsource.Fields(2)

 txtday = Recordsource.Fields(3)

 txttime = Recordsource.Fields(4)

End Function

.....

CODE EFFICIEENCY

CODE EFFICIEENCY :-

The goal of the coding of programming phase is to translate the design of the system produced during the design phase into code in a given programming language. The coding phase affects both testing and maintenance phase. As we know, sthe time spent in coding is a small percentage of software cost, while testing and maintenance consume the major percentage. So it is clear that the goal of coding is not to reduce the implementation cost, but to reduce the cost of testing and maintenance phase. Efficiency of coded is determined my many factors. These factors are readability of code, size of the program, execution time and require memory. Having readability and understandability as a clear objective of the coding activity can itself help in producing software that is more maintainable.



OPTIMISATION OF CODE

CODE OPTIMIZATION USED IN SOFTWARE

Optimisation of code involves giving life to your code it means the person who is not actually programming this code can also easily understand what is happening over there. This utility is achieved by to effective logical tools:->

1.make title

2.comments

Titling:-> if there is suitable title for each code window one can easily interact with that code and calculate the process for debugging the programme.

Comments:-> using comment as good sign of being a good programmer. A comment line actually a deadline which is not executed by the compiler of the language. This line is used for describing the code behaviours of property so that one can easily trace that what is the code performing the action over there.

Documation:->

Relating to code designing where all logic are define for the login process of that code it also better way of code optimization.

TESTING (TESTING TECHNIQUES AND TESTING STRATEGIES)

TESTING OBJECTIVE

Testing is a process of executing a program with intent of finding an error. A goal test case is one that has a probability of finding an as yet undiscovered error. All test should be traceable to customers requirement. Test should be planned long before testing begins. The testing should begin in the small and progress towards testing in the large. Exhaustive testing is not possible. The most efficient, testing should be conducted by an independent third party.

CHARACTERISTIC OF GOOD TESTING

- * A good testing has a high probability of finding errors.
- * A good test is not redundant.
- * A good test should be of broad.
- * A good test should be neither too simple nor too complicated.

TESTING

TYPE OF TESTING

There are two basic approaches to testing functional and structural. Functional testing is also called Black Box Testing. In Black Box Testing the structure of the program is not considered structural testing is called white Box testing.

TYPE OF TESTING

1. WHITE BOX TESTING

2. BLACK BOX TESTING

BLACK BOX TESTING :- Black box testing to be applied during lateral of testing Black box testing enables the software engineer.

- Black box testing enables the software engineer.
- To derive sets input condition will fully exercise the functional requirement of a program.
- Black box testing enables to find error in the following categories.
- Incorrect or missing functions.
- Interface Errors.
- Errors in data structure or external database access.
- Performance errors
- Initialization and termination errors.

The Black box testing procedure is exhaustive testing. One criterion for generating test cases is to generate them randomly. There are no formal rules for designing test cases for functional testing. However there are a number of methods that can be used to select a number of methods that can be used to select test cases they are.

1. Equivalence class partitioning.
2. Boundary value Analysis.

3. Case-Effectr Graphing.

By applying Black Box testing a set of test cases that satisfy the following criteria.

Test cases that reduce, by a count that is greather than one, the number of additional test case must be designed to achieve rcasonable testing.

Test case that tells as something about the presence associated only with the speeific test at haud.

WHITE BOX TESTING:->

Structural Testing or white Bon texting:- To text the structural testing or white bon texting is used. This test is perforamated ewcly in the testing process. This testing also called glass box testing can drive test cases that.

1. Gwrtantee that all the independent paths within the module have been exeresised at least once.
2. Excercise all logical deesions on their true and flace sides.
3. Execate all loops at their boundries and within their validity.
4. Exercise internal data strctyer to enswe their validity.

These are three different approaches to strtural testing there are:-

1. Control flow based testing.
2. Data how Based testing.
3. Mutation testing.

LEVELS OF TESTING :-

Testing is used to default faults introduced drawing specifition and different leves of testing process . the basic leves of flesting are

***Unit testing**

- **integration testing**
- **system testing**
- **acceptance testing**
- **UNIT TESTING**
- Unit testing is personally for verification of the code produced during the code phase that is goal of testing is to the internal logic at the modules. As focus of this testing level is testing the code structure testing is best suited for this level
- **INTEGRATION TESTING**
- The goal of this testing level is to see if the modules can be integrated properly. In other words the emphasis on testing the interfaces between the modules. This testing activity can be considered testing the design.
- **SYSTEM TESTING**
- In this testing entire software is tested. The goal is to see if the software meets its requirements.
- **ACCEPTANCE TESTING**
- Acceptance testing is preformatted using real data of the client to demonstrate that software is working satisfactorily. Testing here focuses in the external behavior by the system. Intense the fanatical testing is performance at this level.
- **VALIDATION CHECKS**
- Validation refers to different set of activities that ensure that the software that has been built is traceable to customer requirement. Validation is the check that “the product made is what was required or not”
-
- Validation testing provides final assurance performance requirement. Black box testing techniques are used exclusively during validation.
- After each validation test case has been conducted, one or two passable and condition exists. The function or performance characteristics conform to specification and are accepted.
- A deviation from specification is uncovered and a deficiency list is created. Deviation or error discovered at this stage in a project can rarely be corrected prior to scheduled completion. It is often necessary for resolving deficiencies.



SYSTEM SECURITY MEASURES



SYSTEM SECURITY :-

Every system application must provide built-in features for security and integrity of data. Without safeguards against unauthorized access fraud, embezzlement etc a system could be so vulnerable as to threaten the survival of the organization.

System security refers to technical innovation and procedures applied to the operating systems to protect against deliberate or accidental damage from defined threats. On the other hand data security is the protection of data from loss, disclosure, modification, and destruction.

Every software developer has to take care about the security measures required by a system according to the functionality provided by it. Data security is of major concern in this respect. To prevent unauthorized access to the system there are many techniques that can be applied. Some special measures should also be developed for quick recovery after a disaster.

My project provides the user, choices to select the category of user through login facility where each category has its own username and password. Each category has privileges according to the access rights required by it.

Like administrator login have permission to access every menu of MDI FORM, Employee login have same permission denied related to administrator work.

Also there is a login profile, maintained by the system to store the user details who access the system so that the administrator should be able to check the type of users entered whenever he wants.

COST ESTIMATION



COST ESTIMATION OF THE PROJECT

The software project management process begins with a set of activities that are collectively called project planning. The first of these activities is Estimation.

Whenever estimates are made, we look into future and accept some degree of uncertainty as a matter of course. Estimation of resources, cost and schedule of a software development effort requires experience, access to good historical information, and the courage to commit to quantitative measures when qualitative data are all that exist. Estimation carries inherent risk that leads to uncertainty.

Project size is another important factor that can affect the accuracy of estimates. As size increases, the interdependency among various elements of the software grows rapidly. Time also affects cost of the project. If time increases then cost is also increased. Then cost is also increased, this project takes six months for completion. According to 500 Rs. Per day the total amount as cost is 90000 Rs.

Risk is measured by the degree of uncertainty in the quantitative estimates established for resources, cost and schedule. If project requirements are subject to change, uncertainty and risk factors become dangerously high. The software planner should demand completeness of function, performance and interface definitions. The planner and customer, should recognize that variability in software requirements means instability in cost and schedules. A project

manager should not become obsessive about estimation; modern software engineering approaches take an iterative view of the development. In such approaches it is possible to revisit the estimate and revise it when customer makes changes to requirements.

IMPLEMENTATION

The new system is completed replacing an existing manual or automated system or it may be a major modification to an existing system. In either case, proper implementation becomes necessary so that a reliable system based on the requirements of the IED organization can be provided. Successful implementation may not guarantee improvement in the organization using the new system, but improper installation will prevent it. It has been observed that even the best system, cannot show good results if the analysis managing the implementation does not attend to every important detail. This is an area where the systems analysis needs to work with utmost care. In the implementation following takes occur:-

1. Training personnel
2. Conversion procedures
3. Post-implementation review.

Training personnel- In the training personnel both systems operators and user needs training. There are two types of training.

- a. System Operators Training.
- b. Users Training.

THERE ARE TWO METHODS OF TRAINING.

1. Vendor and in-service Training.
2. In-house Training

Conversion Methods- Conversion is the process of changing from the old system to the new one. It must be properly planned and executed. Four methods are common in use, these are:-

1. Parallel System.
2. Direct Conversion.
3. Pilot System.
4. Phase-in method.

Post-Implementation Review- After system is implemented and conversion is complete, a review should be conducted to determine whether the system is meeting expectations and where improvements are needed. A post implementation review measures the systems' performance against predefined requirements. Once request is filed, the user is asked how well the measured benefits have been realized. Suggestions regarding changes and improvements are also asked for.

REVIEW PLAN

The review team prepares a formal review plan around the objectives of the review, the type of evaluation to be carried out and the time schedule required. An overall plan covers the following areas:-

1. Administrative Plan.
2. Personnel Requirements Plan.
3. Hardware Plan.
4. Documents Review Plan.

The review not only assesses how well the current system is designed and implemented ,but also is a valuable sources of information that can be applied to the next systems project.

ON SITE TEST:-

On feed all types of data for the last period for menu system and try generating all kinds of reports. Now these reports have to be mached with manual system Reports. If everything is ok then real life use of package can be started immediately.

MAINTENANCE:-

Maintenance is actually the implementation of the post implementation review plan. Any variation from the prepared package Report desired any improvement would now be suggested by actual users. On as installation and actual use has taken place the feature change desired would actually be called Maintenance of package.

The programmers Analyst spend sufficient time for maintaining programs. The study on the maintenance requirements for the information system revealed that:-

- a. 60-90 percent of the overall cost of software during the life of a system is spent on maintenance.
- b. In documented cases, the cost of maintenance, when measured on the basis of writing each instruction in coding from is more than 50 times the cost of developing a system.
- c. The software demand is increasing at faster rate than supply. Many programmers are devoting more time on systems maintenance than on new software development. There is a backlog of new development work. The maintenance can be classified are as:-

1. Corrective:

2. Adaption or Perfective:

Corrective Maintains means repairing, processing or performance failures of making alterations because of previously ill-defined problems.

Adaption maintance means changing the program functions. Enchanting the performance or modifying the programs according to user's additional or changing needs are included in perfective maintance. More time and the money are spent on perfective than on corrective and adaptive maintenance together.

Maintenance covers a wide range of acitivities including correcting coding and design errors, updating documentation and test data and upgrading user support.

Many activities classified as maintenance actually fall under enhancements.

Maintenance means restoring something to is original position. Unlike hardware, software does not wear out; it is corrected. In contrast, enhancement means adding, modifying or re-developing the code to support changes in the specifications. It is to keep with changing user needs and the operational environment.

The keys to reduce the need for maintenance while making it possible to carry on with essential tasks more efficiently are as follows:-

- A. More accurately defining the user's requirement during systems development.
- B. Preparation of system documentation in a better way.
- C. Using more effective ways for designing processing logic and communicating it to project team members.

- D. Making better use of existing tools and techniques.
- E. Managing the systems engineering process effectively.

An addition factor in the success of the maintenance programmer is the work environment. Maintenance programmers have generally been paid less amount and receives less recognition than other programmers. Little attention has been paid to their training and carrer plans with in the MIS function. Maintenance demands more orientation and training then any other programming activities, especially for entry-level programmers. The environment must recognize the needs of the maintenance programmer for tools, methods and training.

REPORT

The out put and report generation for Hospital management system will include the following details

- Doctors detail
- Visiting doctors detail
- patient detail
- outpatient door detail
- staff detail
- nurse detail
- ward boy detail
- employee detail
- employees work time detail
- emp_work detail
- emp attendance detail
- medicine detail
- medicine prescribed detail
- holiday detail
- accounts detail

PERT & GANTT CHART



GANTT CHART

When creating software schedule, the planner begins with a set of tasks, if amount tools are used, the work breakdown is input as task network or task outline effort, duration and start date are then input for each task. A timeline chart also called a GANTT CHART.

PERT-(PROGRAM EVALUATION AND RESEARCH TECHNIQUE)

PERT Chart is a tool which got a multiford advantage that it deals with various activities and events that forms the part of the entire Project.

Activity:-

Activity is an application of time and resource that is needed to progress on one hand and expands a time period on another.

Event:

A specified objective to the overall plan to be archive at a particular instance of time.

Critical Path:

The path traced though those activity network which constitutes the longest overall time.The access time which can be added to any activity that is not in the critical path without changing the overall completion data of project.

Note:

Activities has few characteristics:-

1. An activity may be independent i.e. the starting and ending of the activity are not dependent on the activities.
2. An activity may be dependent which means that the starting of the activity depends on the completion of other activity.

RULES:-

1. Activities are denoted by circles.
2. Time is denoted by straight line.

OBJECTIVES:--

The basic objective of the PERT chart is to control the project and ensure that the project is completed within estimated time.

ADVANTAGES:-

1. Each job must be examined and defined in great details to create a specified chart.
2. Network shows relationship to aid programming.
3. Help in distribution of resources and finally its allocation.
4. Put a time limit and schedule the project.

Serial No.	Activities	Time Required (in weeks)
In parallel		
1	Submission of project trainee letter	1/7
2	Essurence of project	1
3	Collect Data & Information	1
4	Study and Analysis	2
5	Feasibility report prepration	1
6	Prepare synopsis	1
7	Program specification	1/2
8	Program design	1
9	Coding	3
10	Testing and debugging	1/2
11	Documentation	1
12	Auditing and Review	1/2
13	Installation	1

PERT CHART

PERT CHART

