Department of Computer Engineering & Applications

GLA University

Mathura- 281406, INDIA

<u>SYNOPSIS</u>
<u>on</u>

<u>HOSPITAL MANAGEMENT SYSTEM</u>

Submitted by:
**Piyush kumar singh
(201500469)
Keyshav Sharma
(201500337)
Abhishek Chaudhary
(201500015)**

Submitted to:
Ms. Ruchi Talwar Mam

Declaration:-

We hereby declare that the work which is being presented in the Project Report "HOSPITAL MANAGEMENT SYSTEM"

in partial fulfillment of the requirements for Project is an authentic record of our own work carried under the supervision of **Ms. Ruchi Talwar Mam , GLA University,Mathura**.

Sign _____

Name of Candidates:

Piyush kumar singh (201500469)

Keyshav sharma (201500337)

AbhishekChaudhary (201500018)

Certificate:-

This is to certify that the above statements made by the candidate are correct to the best of my/our knowledge and belief.

_____

Project Mentor

Ms. Ruchi Talwar Mam

<u>INTRODUCTION</u>

Hospital Management System provides the benefits of streamlined operations, enhanced administration & control, superior patient care, strict cost control and improved profitability. HMS is powerful, flexible, and easy to use and is designed and developed to deliver real conceivable benefits to hospitals .In this project, we use PHP and MYSQL database. It has three modules.

1. Admin
2. Doctor
3. Patient

Admin:

1. Dashboard: After clicking on this, the admin can mange users, manage doctors and manage patients, and view queries.
2. Patients: In this section, the admin can manage patients by entering patient's details like name, place and age etc.
3. Doctors session logs: In this section, the admin can view the doctor session logins. For example-doctor's login time,logout time etc.
4. User session logs: In this section, the admin can view the user session logins. For example-user's login time,logout time etc.

Doctor:

1. **Dashboard:** After clicking on this a doctor can do the following activites. For example- view his/her profile (update profile), view his/her appointments.
2. **Appointment history:** In this section, a doctor can view his/her appointment history.
3. **Patients:** In this section, a doctor can add patients by entering patient's details like name, place and age etc. and also manage patients.
4. **Search:** In this section, doctor can manage patients by searching them by their mobile no., name etc.

Patient:

1. **Dashboard:** After clicking on this user can do the following activites. For example- view his/her profile (update profile), view his/her appoinmnets.
2. **Book appointment :** In this section, user can book appointments by entering, doctor specialization, doctors, date and time.
3. **Appoinments history :** In this section, user can view his/her appointment history.
4. **Medical history :** In this section, user can view his/her medical history.

Scope of the Project

The proposed software product is the Hospital Management system (HMS). The system will be used in any hospital, clinic, dispensary or pathology labs. Clinic, dispensary or pathology to get the information from the patients and then storing that data for future usages. The current system in use is a paper based system. It is too slow and cannot provide updated lists of patients within reasonable timeframe. The intention of the system is to reduce over-time pay and increase the number of patients that can be treated accurately. Requirement statements in these documents are both functional and non-functional.

Software & Hardware requirements

✓ Any Version of browser after Mozilla Firefox 4.0, Internet Explorer 6.0,chrome

Hardware requirements:

✓ Any processor after Pentium 4.
✓ Any version of Windows XP or later.
✓ Processor speed: 2.0 GHz
✓ RAM : 1GB
✓ Hard disk: 40GB to 80 GB

Software requirements:

✓ Database : MySQL
✓ Server : Apache
✓ Frontend : HTML
✓ Scripting Language : JavaScript
✓ IDE : Sublime
✓ Technology : PHP

System Design

- The entire system is projected with a physical diagram which specifics the actual storage parameters that are physically necessary for any database to be stored on to the disk. The overall systems existential idea is derived from this diagram.

- The relation upon the system is structure through a conceptual ER- Diagram, which not only specifics the existential entities but also the standard relations through which the system exists and the cardinalities that are necessary for the system state to continue.

- The content level DFD is provided to have an idea of the functional inputs and outputs that are achieved through the system. The system depicts the input and output standards at the high level of the systems existence.

A DFD does not show a sequence of steps. A DFD only shows what the different process in a system is and what data flows between them.
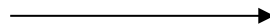
The following are some DFD symbols used in the project

External entities
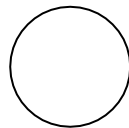
Process: A transaction of information that resides within the bounds of the system to be module.

DATAFLOW

DATASTORE: A repository of data that is to be stored for use by one or more processes, may be as simple as buffer of queue or as a relational database.

RULES FOR DFD:

- Fix the scope of the system by means of context diagrams.
- Organize the DFD so that the main sequence of the actions reads left to right and top to bottom.
- Identify all inputs and outputs.
- Identify and label each process internal to the system with rounded circles.
- A process is required for all the data transformation and transfers. Therefore, never connect a data store to a data source or the destinations or another data store with just a data flow arrow.
- Do not indicate hardware and ignore control information.
- Make sure the names of the processes accurately convey everything the process is done.
- There must not be unnamed process.
- Indicate external sources and destinations of the data, with squares.
- Number each occurrence of repeated external entities.
- Identify all data flows for each process step, except simple Record retrievals.
- Label data flow on each arrow.
- Use details flow on each arrow.

- Use details flow on each arrow.
- Use the details flow arrow to indicate data movements.
- There can't be unnamed data flow.
- A data flow can't connect two external entities.

LEVELS OF DFD:

The complexity of the business system means that it is a responsible to represent the operations of any system of single data flow diagram. At the top level, an Overview of the different systems in an organization is shown by the way of context analysis diagram. When exploded into DFD

They are represented by:

- LEVEL-0 : SYSTEM INPUT/OUTPUT
- LEVEL-1:SUBSYSTEM LEVEL DATAFLOW FUNCTIONAL
- LEVEL-2: FILE LEVEL DETAIL DATA FLOW.

The input and output data shown should be consistent from one level to the next.

LEVEL-0: SYSTEM INPUT/OUTPUT LEVEL

A level-0 DFD describes the system-wide boundaries, dealing inputs to and outputs from the system and major processes. This diagram is similar to the combined user-level context diagram.

LEVEL-1: SUBSYSTEM LEVEL DATA FLOW

A level-1 DFD describes the next level of details within the system, detailing the data flows between subsystems, which makeup the whole

First Level DFD - Hospital Management System
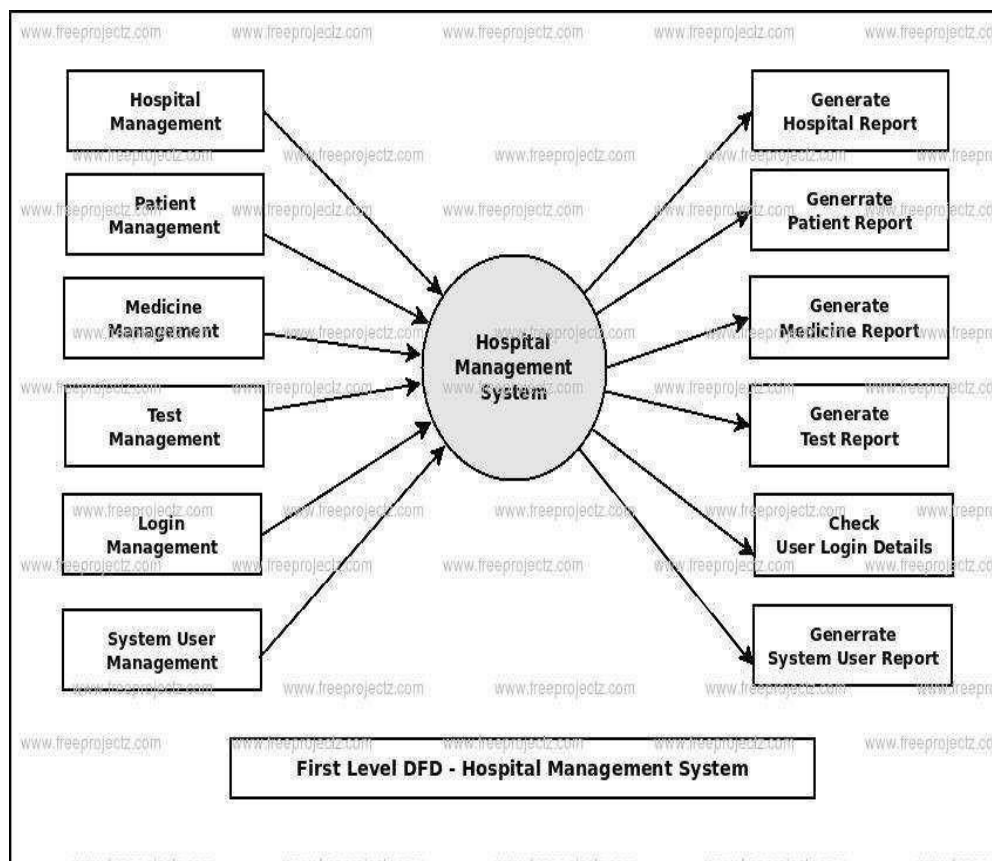
E-R Diagram of Hospital Management System

An entity-relationship diagram (ERD) is an abstract and conceptual representation of data. Entity- relationship modeling is a database modeling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion

Hospital		Patients

Patient information

Admit

Room

Hospital	Doctor record	Doctor

Patient	On Line Appointment	Hospital