Mini Project Report

HOSPITAL MANAGEMENT SYSTEM

Submitted by

Mayur Pingale	1032180083	PB07
Rohit Patil	1032180216	PB11
Karan Patil	1032180232	PB13
Hrishikesh Neymade	1032181180	PB39

Under the Guidance of

Prof. Snehal P. Umare

At

School of Computer Engineering and Technology

Abstract

This project Hospital Management system includes registration of patients, storing their details into the system, and also has computerized billing. The software has the facility to give a unique id for every patient and stores the details of every patient and hospital tests done automatically. It includes a search facility to know the current status of each patient. User can search details of a patient using the id. The Hospital Management System can be entered using a username and password. It is accessible either by an administrator or a doctor. Only they can add data into the database. The data can be retrieved easily. The interface is very user-friendly. The data are well protected for personal use and makes the data processing very fast.

Table of Contents

List of Abbreviations

DBMS - Database management system

JDBC - Java Database Connectivity

ER - Entity Relationship

DDL - Data Definition Language

DML - Data Manipulation Language

DCL - Data Control Language

GUI - Graphical User Interface

IEEE - Institute of Electrical and Electronics Engineers

List of Figures

Introduction (Motivation and Objectives)

Database is an organized collection of data. The data is typically organized to model aspects of reality in a way that supports requiring information. A DBMS makes it possible for end users to read, create, update, delete in a database. The DBMS essentially serves as an interface between the end users or application programs, ensuring that data is consistently organized and remains easily accessible. The DBMS manages three things : the data, the database engine to be accessed, locked and modified and the database schema, which defines the database's logical structure. These 3 foundational elements help provide concurrency, security and data integrity. The DBMS can offer both logical and physical independence. That means it can protect users and applications from needing to know where the data is located or having to be concerned about the changes to the physical structure of data.

The main purpose for having a database system for Hospital is to reduce the manual errors involved in various details involved and make it convenient to maintain details about the patients and the doctors. Due to automation, many loopholes that exist in manual maintenance can be removed. The speed of obtaining and processing the details will be fast.

Problem Definition

The project deal with storing of database for a Hospital. It comprises information about the details of the patients, doctors and other relevant information. The hospital management system database project will help to reduce the pen paper work in the hospitals if not completely. There will be many patients admitted in the hospitals. So to keep track about the information like personal details, treatment that is going on etc. There will be doctors who treat these patients. One doctor can treat many patients. So the doctor and the patient are related in the database. This database can also contains the information regarding the doctors, rooms, labs, etc.

The features that can be included in the hospital management system are as follows:

- **Patients database management:** The details related to the patients like name, address, contact number, disease suffered from, treatment given and so on.
- **Staff database management:** The details of the staff like the doctors, rooms, labs and other details can be stored using this application.
- **Information at one stretch:** The information of the staff and the patients can be obtained at just one stretch.

Tools and Technologies Used

- Java NetBean IDE 8.2
- Java JDK 8
- MySQL Workbench 8.0
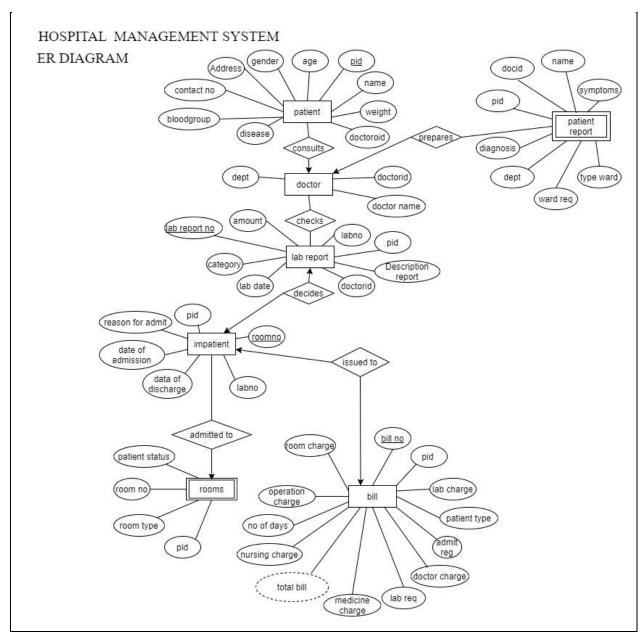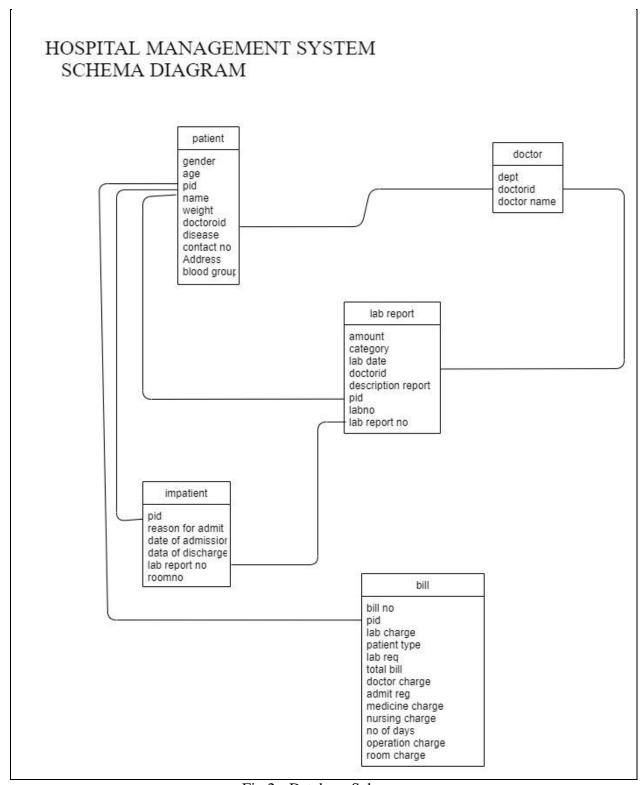- JDBC Connectivity
- Ubuntu

Database Design (ER Diagram)

HOSPITAL MANAGEMENT SYSTEM
ER DIAGRAM

Fig 1 - ER Diagram

Database Schema

HOSPITAL MANAGEMENT SYSTEM
SCHEMA DIAGRAM

patient
gender
age
pid
name
weight
doctoroid
disease
contact no
Address
blood group

doctor
dept
doctorid
doctor name

lab report
amount
category
lab date
doctorid
description report
pid
labno
lab report no

impatient
pid
reason for admit
date of admission
data of discharge
lab report no
roomno

bill
bill no
pid
lab charge
patient type
lab req
total bill
doctor charge
admit reg
medicine charge
nursing charge
no of days
operation charge
room charge

Fig 2 - Database Schema

DDL

```
CREATE TABLE doctor(
doc_id VARCHAR(2) PRIMARY KEY,
dept VARCHAR(20),
full_name VARCHAR(20)
);

CREATE TABLE patient(
patient_id VARCHAR(2) PRIMARY KEY,
full_name VARCHAR(20),
contact VARCHAR(10),
age VARCHAR(2),
gender VARCHAR(10),
weight VARCHAR(3),
bloodgrp VARCHAR(10),
address VARCHAR(300),
disease VARCHAR(20),
doc_id VARCHAR(2),
FOREIGN KEY(doc_id) REFERENCES doctor(doc_id)
);

CREATE TABLE patient_report(
patient_id VARCHAR(2),
dept VARCHAR(20),
doc_id VARCHAR(2),
full_name VARCHAR(20),
symptoms VARCHAR(30),
diagnosis VARCHAR(30),
```

```
medicine VARCHAR(30),

wardreq varchar(5),

typeward varchar(20),

FOREIGN KEY(patient_id) REFERENCES patient(patient_id),

FOREIGN KEY(doc_id) REFERENCES doctor(doc_id)
);


CREATE TABLE labreport(

patient_id VARCHAR(2),

lab_report_no VARCHAR(2) PRIMARY KEY,

lab_id VARCHAR(2),

doc_id VARCHAR(2),

category VARCHAR(20),

amount VARCHAR(10),

descriptionreport VARCHAR(30),

lab_date VARCHAR(10),

FOREIGN KEY(patient_id) REFERENCES patient(patient_id),

FOREIGN KEY(doc_id) REFERENCES doctor(doc_id)
);


create table room(

patient_id VARCHAR(2),

patient_status VARCHAR(20),

room_no VARCHAR(4) PRIMARY KEY,

room_type VARCHAR(10),

FOREIGN KEY(patient_id) REFERENCES patient(patient_id) );


create table impatient(
```

```
patient_id VARCHAR(2),

lab_report_no VARCHAR(2),

date_of_admission VARCHAR(10),

date_of_discharge VARCHAR(10),

room_no VARCHAR(4),

reason_for_admit varchar(30),

FOREIGN KEY(room_no) REFERENCES room(room_no),

FOREIGN KEY(patient_id) REFERENCES patient(patient_id),

FOREIGN KEY(lab_report_no) REFERENCES labreport(lab_report_no)
);


create table bill(

patient_id VARCHAR(2),

bill_no VARCHAR(3) PRIMARY KEY,

patient_type VARCHAR(20),

doctor_charge VARCHAR(10),

medicine_charge VARCHAR(10),

nursing_charge VARCHAR(10),

labreq VARCHAR(5),

lab_charge VARCHAR(10),

admitreq varchar(5),

no_of_days VARCHAR(3),

room_charge VARCHAR(10),

operationcharge VARCHAR(10),

total_bill VARCHAR(10),

FOREIGN KEY(patient_id) REFERENCES patient(patient_id)
 );
```

DML

UPDATE patient SET weight = 80 WHERE pid = 1;

UPDATE patient SET age = 20 WHERE pid = 1;

UPDATE patient SET Address = "" WHERE pid = 1;

UPDATE patient SET contact_no = 9988665544 WHERE pid = 1;

UPDATE patient SET disease = "" WHERE pid = 1;

UPDATE patient SET doctor_id = 21 WHERE pid = 1;

DCL

GRANT ALL PRIVILEDGES ON hspl_mgmt_sys TO admin;

PLSQL Procedure/Function

Delimeter //

Create Procedure Sum()

BEGIN

SET total_bill = nursing_charge+doctor_charge+room_charge+lab_charge+operation_charge+medicine_charge

FROM bill WHERE pid = 1

END //

DELIMETER ;

Frontend GUI Screenshots

Fig 3 - Home Page

Fig 4 - Admin Login Page

Fig 5 - Admin Home Page
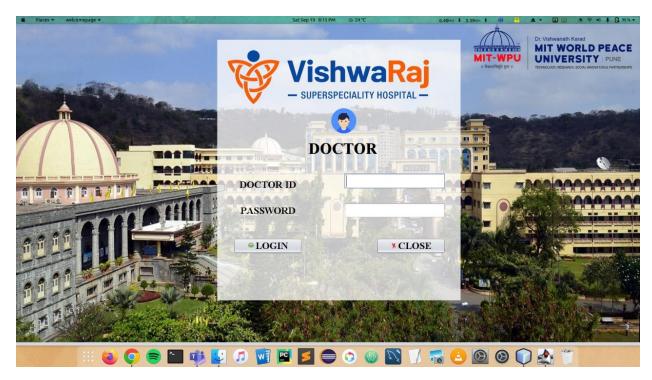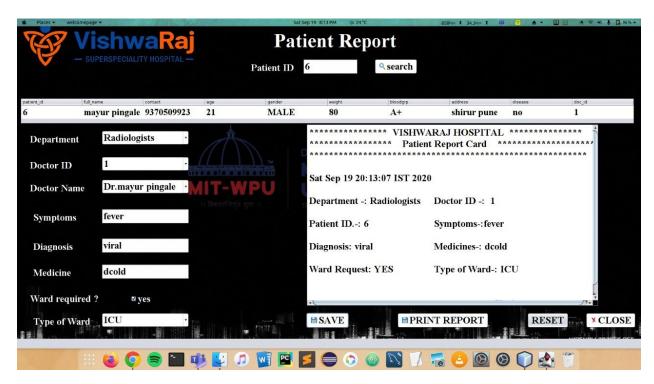

Fig 6 - Patient Information

Fig 7 - Doctor Login Page

Fig 8 - Patient Report Card
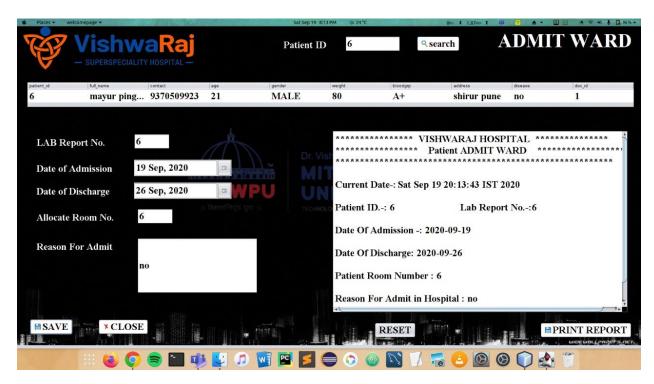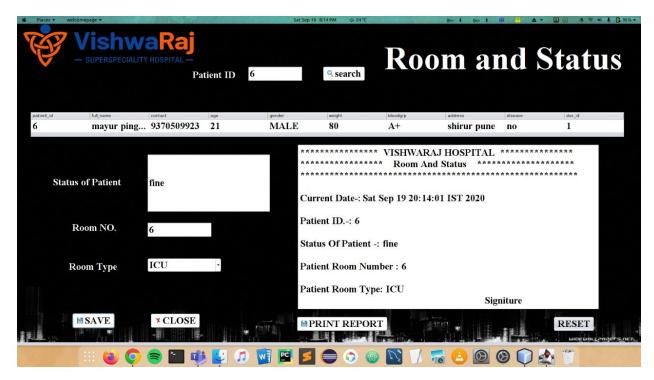
Fig 9 - Lab Report

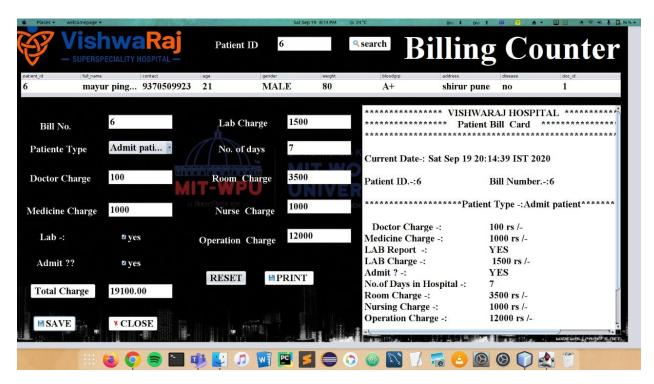Fig 10 - Admit Ward

Fig 11 - Room and Status
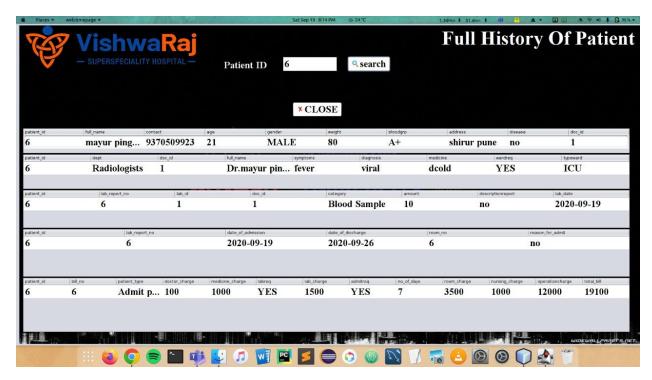

Fig 12 - Billing Counter

Fig 13 - Full History of Patient

Fig 14 - Hospital Information

Conclusion

Taking into account all the mentioned details, we can make the conclusion that the hospital management system is the inevitable part of the lifecycle of the modern medical institution. It automates numerous daily operations and enables smooth interactions of the users. Developing the hospital system software was a great opportunity to create the distinct, efficient and fast delivering healthcare model.