A Project/Dissertation Report

on

ONLINE HOSPITAL MANAGEMENT SYSTEM

Submitted in partial fulfillment of the requirement for the award of the degree of

B.Tech Computer Science Engineering

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of
Name of Supervisor :- ABHAY KUMAR
Assistant Professor

Submitted By

Name Of Student - PULENDRA KUMAR YADAV
Admission No:-19SCSE1010413
Enrollment No:- 19021011593
Sec - 07
&
Name Of Student - RIKESH KUMAR
Admission No:- 19SCSE1010385
Enrollment No:- 19021011565
Sec - 09

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA
March-2022

Table of contents

Chapter 1: INTRODUCTION

1.1 ABSTRACT..4
1.2 Introduction.. 4
1.3. Problem Statement............................5
1.4 Goals.. 5
1.5 Objective..5
1.6 Scope of the Project..6
1.7 MODULES...6
1.8 Admin module...
1.9 user module...7
1.10 Doctor module...7
1.11 Nurse module...7
1.12 Pharmacist module..7
1.13 Laboratorist module..8
1.14 Accountant module..8

Chapter 2: DESIGN

2.1 SYSTEM DESIGN...10
2.1.1 INTRODUCTION TO UML..10
2.2 UML Approach...11
2.3 Use case diagram of our project...12
2.4 Class Diagram...13
2.5 Sequence diagram..14
2.6 Activity diagram..15
2.7 Deployement diagram..15
2.8 E-R Diagrams.. 16

Chapter 3 : ANALYSIS

3.1 EXISTING SYSTEM..18
3.2 PROPOSED SYSTEM..18
3.3 FEASIBILITY STUDY...18
3.3.1 Economic Feasibility..18
3.3.2 Technical Feasibility..19
3.3.3 Operational Feasibility...19
3.4 SOFTWARE SPECIFICATION..19
3.5 MySQL..20
3.6 PHP...22

Chapter 4 : SYSTEM IMPLEMENTATION

4.Login page...31
4.2Hospital page...32

Chapter-1
Introduction

1.1 ABSTRAC

Hospital Management System is an organized computerized system designed and programmed to deal with day to day operations and management of the hospital activities. The program can look after inpatients, outpatients, records, database treatments, status illness, billings in the pharmacy and labs. It also maintains hospital information such as ward id, doctors in charge and department administering. The major problem for the patient nowadays to get report after consultation , many hospital managing reports in their system but it's not available to the patient when he / she is outside. In this project we are going to provide the extra facility to store the report in the database and make available from anywhere in the world.

1.2 Introduction:

The project Hospital Management system includes registration of patients, storing their details into the system, and also computerized billing in the pharmacy, and labs. The software has the facility to give a unique id for every patient and stores the details of every patient and the staff automatically. It includes a search facility to know the current status of each room. User can search availability of a doctor and the 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 receptionist. 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.

Hospital Management System is powerful, flexible, and easy to use and is designed and developed to deliver real conceivable benefits to hospitals.
Hospital Management System is designed for multispeciality hospitals, to cover a wide range of hospital administration and management processes. It is an integrated end-to-end Hospital Management System that provides relevant information across the hospital to support effective decision making for patient care, hospital administration and critical financial accounting, in a seamless flow.
Hospital Management System is a software product suite designed to improve the quality and management of hospital management in the areas of clinical process analysis and activity-based costing. Hospital Management System enables you to develop your organization and improve its effectiveness and quality of work. Managing the key processes efficiently is critical to the success of the hospital helps you manage your processes

1.3 Problem Introduction:

Lack of immediate retrievals: -
The information is very difficult to retrieve and to find particular information like- E.g. - To find out about the patient's history, the user has to go through various registers. This results in in convenienceand wastage of time.
Lack of immediate information storage: -
The information generated by various transactions takes time and efforts to be stored at right place.
Lack of prompt updating: -
Various changes to information like patient details or immunization details of child are difficult to make as paper work is involved.
Error prone manual calculation: -
Manual calculations are error prone and take a lot of time this may result in incorrect information. For example calculation of patient's bill based on various treatments.
Preparation of accurate and prompt reports: -
This becomes a difficult task as information is difficult to collect from various register.

1.4 Goals

1-User friendly
2-Simple fast
3-Low cost and effective
4-It deals with the collection of patient's information
5- Diagnosis

1.5 Objective:-

1) Define hospital
2) Recording information about the Patients that come.
3) Generating bills.
4) Recording information related to diagnosis given to Patients.
5) Keeping record of the Immunization provided to children/patients.
6) Keeping information about various diseases and medicines available to cure them.

These are the various jobs that need to be done in a Hospital by the operational staff and Doctors. All these works are done on papers.

1.6 Scope of the Project:-

1) Information about Patients is done by just writing the Patients name, age and gender. Whenever the Patient comes up his information is stored freshly.
2) Bills are generated by recording price for each facility provided to Patient on a separate sheet and at last they all are summed up.
3) Diagnosis information to patients is generally recorded on the document, which contains Patient information. It is destroyed after some time period to decrease the paper load in the office.
4) Immunization records of children are maintained in pre-formatted sheets, which are kept in a file.
5) Information about various diseases is not kept as any document. Doctors themselves do this job by remembering various medicines.

All this work is done manually by the receptionist and other operational staff and lot of papers are needed to be handled and taken care of. Doctors have to remember various medicines available for diagnosis and sometimes miss better alternatives as they can’t remember them at that time.

1.7 MODULES:

The entire project mainly consists of 7 modules, which are

- Admin module
- User module (patient)
- Doctor module
- Nurse module
- Pharmacist module
- Laboratorist module
- Accountant module

1.8 Admin module:

- manage department of hospitals, user, doctor, nurse, pharmacist, laboratorist accounts.
- watch appointment of doctors
- watch transaction reports of patient payment
- Bed ,ward, cabin status
- watch blood bank report
- watch medicine status of hospital stock
- watch operation report
- watch birth report

- watch diagnosis report
- watch death report

1.9 user module(patient):

- View appointment list and status with doctors
- View prescription details
- View medication from doctor
- View doctor list
- View blood bank status
- View operation history
- View admit history. like bed, ward icu etc
- Manage own profile

1.10 Doctor module:

- Manage patient. account opening and updating
- Create, manage appointment with patient
- Create prescription for patient
- Provide medication for patients
- Issue for operation of patients and creates operation report
- Manage own profile

1.11 Nurse module:

- Manage patient. account opening and updating
- Allot bed, ward, cabin for patients
- Provide medication according to patient prescription
- Manage blood bank and update status
- Keep record of patient operation, baby born and death of patient
- Manage own profile

1.12 Pharmacist module:

- Maintain medicine
- Keep records of hospitals stock medicines and status
- Manage medicine categories
- Watch prescription of patient
- Provide medication to prescriptions

1.13 Laboratorist module:

- Watch prescription list
- Upload diagnostic report
- Preview of report files. like xray images, ct scan, mri reports
- Manage own profile

1.14 Accountant module:

- Create invoice for payment
- Order invoice to patient
- Take cash payment
- Watch payment history of patients
- Manage own profile

CHAPTER 2
DESIGN

2.1 SYSTEM DESIGN:

2.1.1 INTRODUCTION TO UML:

UML Design

The Unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the software system and its components. It is a graphical language , which provides a vocabulary and set of semantics and rules. The UML focuses on the conceptual and physical representation of the system. It captures the decisions and understandings about systems that must be constructed. It is used to understand, design, configure, maintain, and control information about the systems.

The UML is a language for:

- Visualizing
- Specifying
- Constructing
- Documenting

Visualizing

Through UML we see or visualize an existing system and ultimately we visualize how the system is going to be after implementation. Unless we think, we cannot implement. UML helps to visualize, how the components of the system communicate and interact with each other.

Specifying

Specifying means building, models that are precise, unambiguous and complete UML addresses the specification of all the important analysis design, implementation decisions that must be made in developing and deploying a software system.

Constructing

UML models can be directly connected to a variety of programming language through mapping a model from UML to a programming language like JAVA or C++ or VB. Forward Engineering and Reverse Engineering is possible through UML.

Documenting

The Deliverables of a project apart from coding are some Artifacts, which are critical in controlling, measuring and communicating about a system during its developing requirements, architecture, desire, source code, project plans, tests, prototypes releasers, etc...

2.2 UML Approach

UML Diagram

A diagram is the graphical presentation of a set of elements, most often rendered as a connected graph of vertices and arcs . you draw diagram to visualize a system from different perspective, so a diagram is a projection into a system. For all but most trivial systems, a diagram represents an elided view of the elements that make up a system. The same element may appear in all diagrams, only a few diagrams , or in no diagrams at all. In theory, a diagram may contain any combination of things and relationships. In practice, however, a small number of common combinations arise, which are consistent with the five most useful views that comprise the architecture of a software-intensive system. For this reason, the UML includes nine such diagrams:

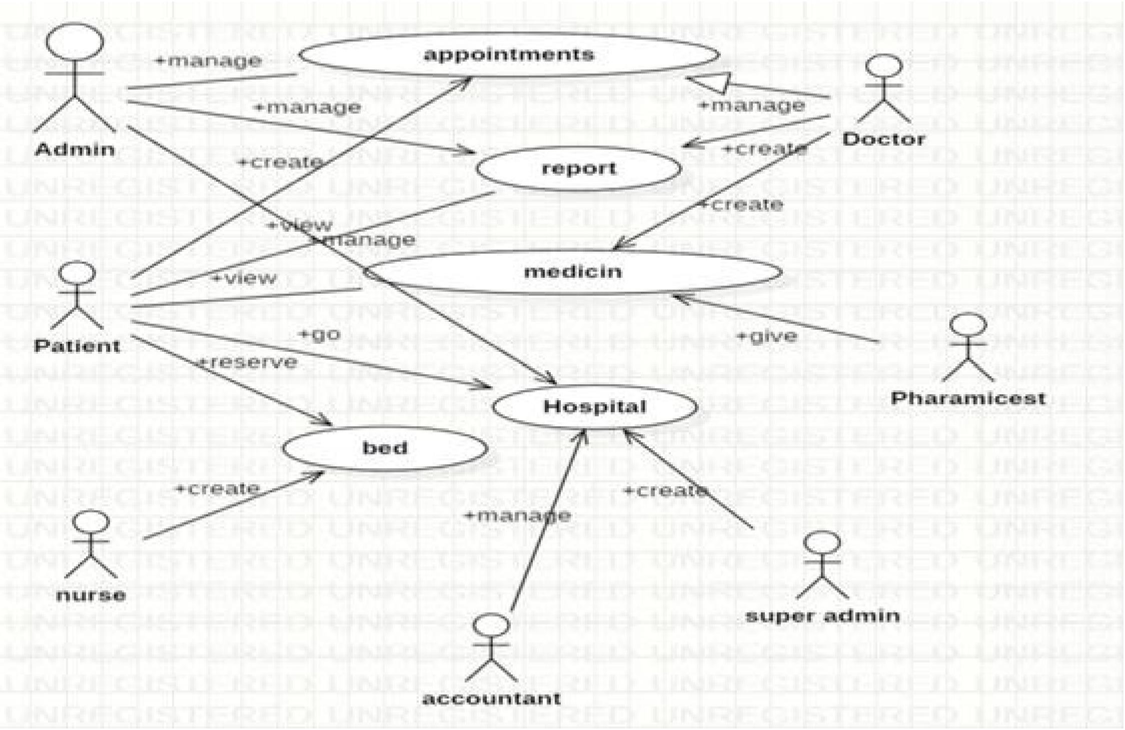
1. Class diagram
2. Object diagram
3. Use case diagram
4. Sequence diagram
5. Collaboration diagram
6. State chart diagram
7. Activity diagram
8. Component diagram
9. Deployment diagram

USE CASE DIAGRAM:

A usecase diagram in the Unified Modeling Language(UML) is atype of behavioral diagram defined by and created from a use-case analysis.its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals(represented as use cases),and any dependencies between those use cases.

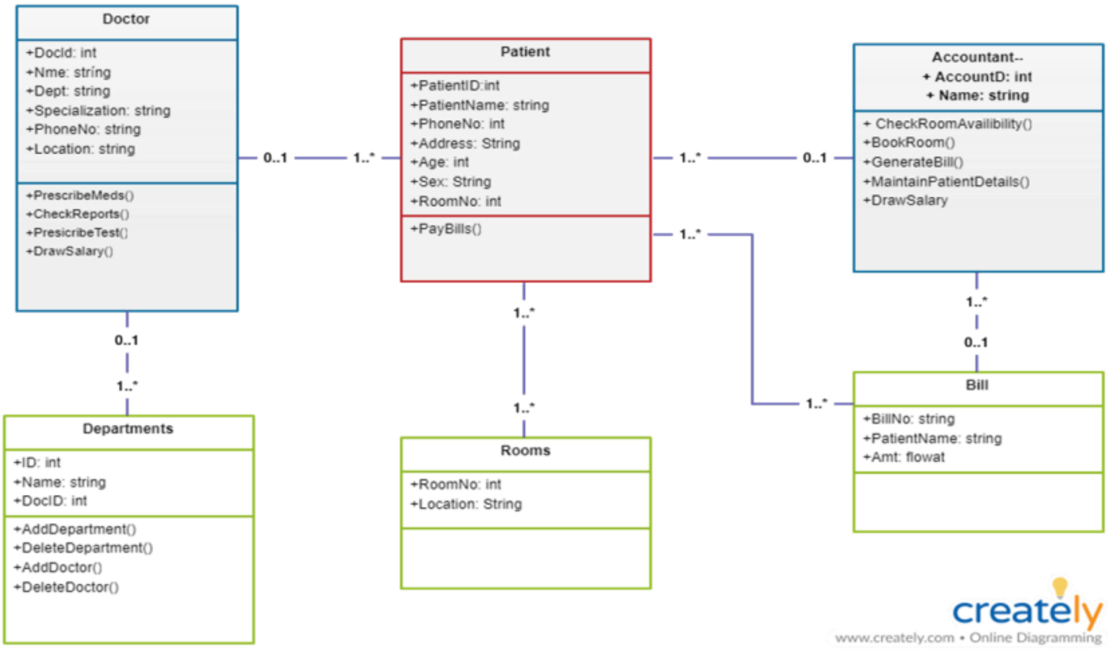
Use case diagrams are formally included in two modeling languages defined by the OMG:theunfied modeling language(UML) and the systems modeling language(sysML)

2.3 Use case diagram of our project:



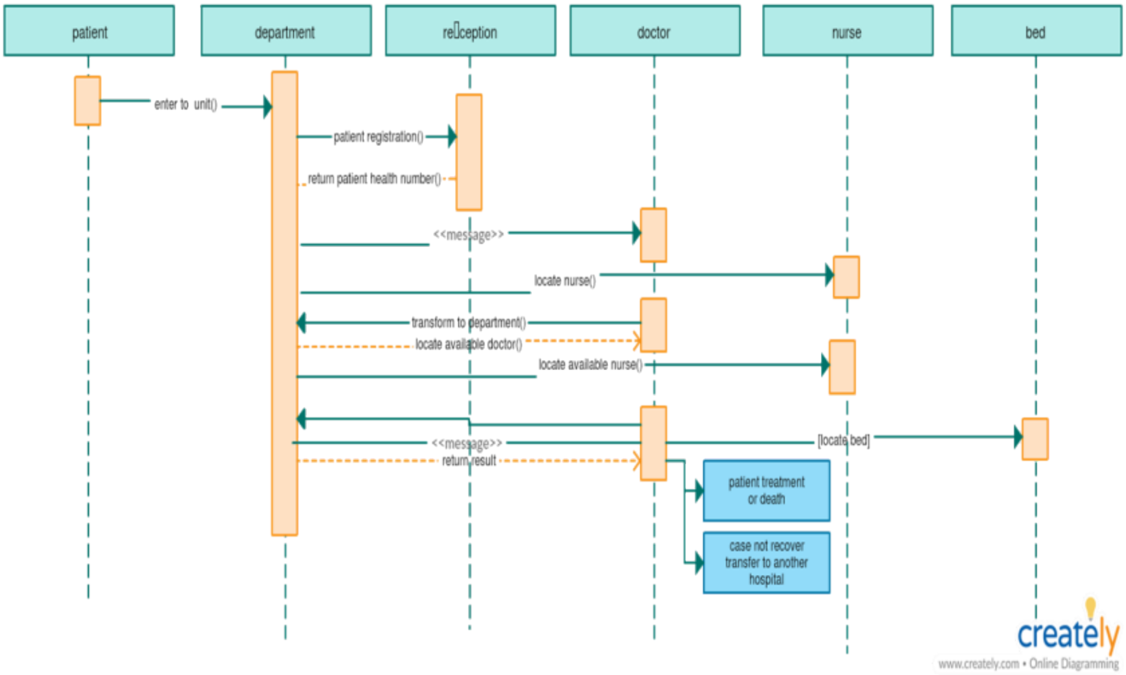
2.4 Class Diagram:

A Class is a category or group of things that has similar attributes and common behavior. A Rectangle is the icon that represents the class it is divided into three areas. The upper most area contains the name, the middle; area contains the attributes and the lowest areas show the operations. Class diagrams provides the representation that developers work from. Class diagrams help on the analysis side, too.



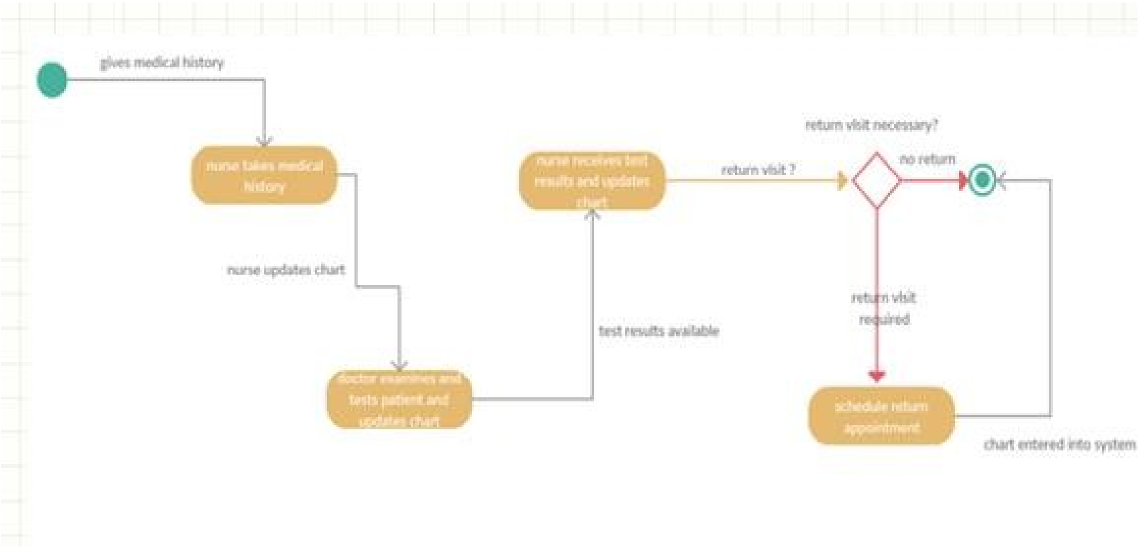
2.5 Sequence diagram:

A **Sequence Diagram** is an interaction diagram that emphasis the time ordering of messages; a collaboration diagram is an interaction diagram that emphasizes the structural organization of the objects that send and receive messages. Sequence diagrams and collaboration diagrams are isomorphic, meaning that you can take one and transform it into the other.


patient
department
reception
doctor
nurse
bed
enter to unit()
patient registration()
return patient health number()
<<message>>
locate nurse()
transform to department()
locate available doctor()
locate available nurse()
<<message>>
return result
[locate bed]
patient treatment or death
case not recover transfer to another hospital
creately
www.creately.com • Online Diagramming

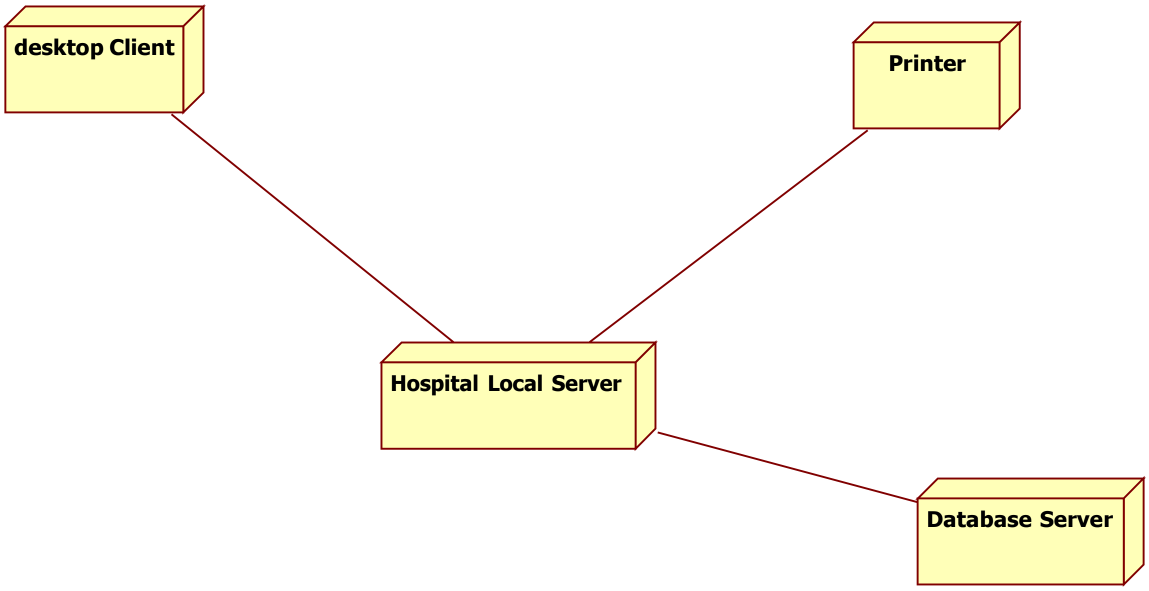
2.6 Activity diagram:

A **Activity diagram** is another important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flow chart that modeling the flow from one activity to another activity.


gives medical history
nurse takes medical history
nurse updates chart
doctor examines and tests patient and updates chart
test results available
nurse receives test results and updates chart
return visit ?
return visit necessary?
no return
return visit required
schedule return appointment
chart entered into system

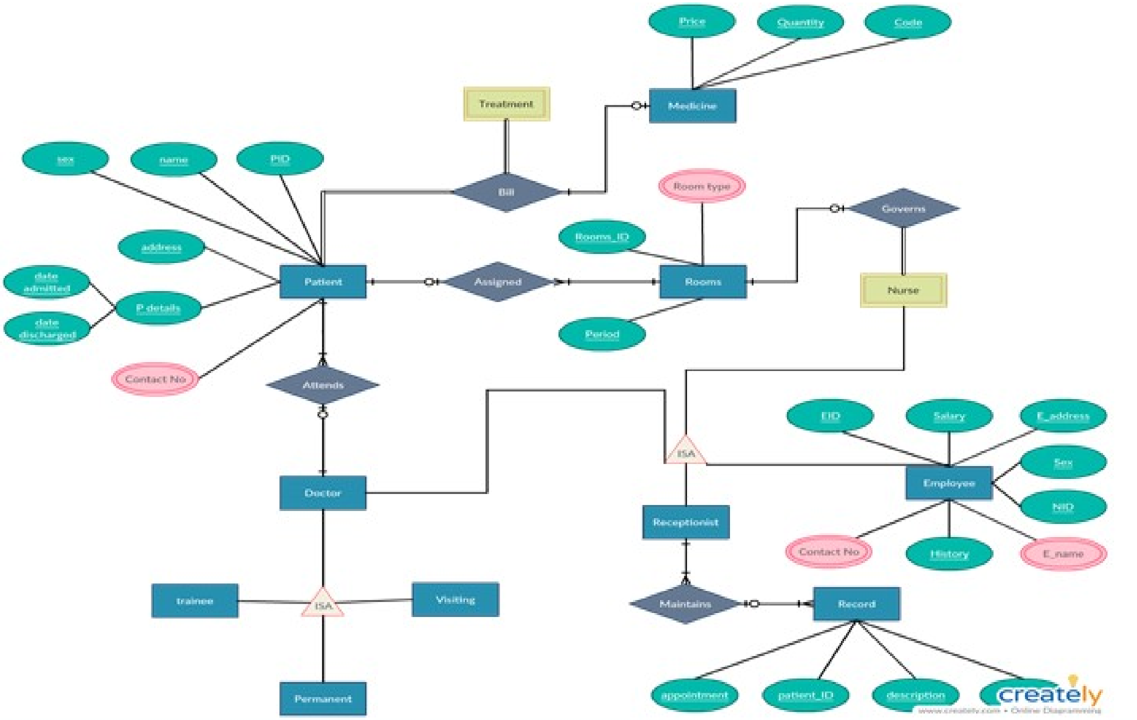
2.7 Deployement diagram:

A **Deployment Diagram** shows the configuration of run-time processing nodes and the components that live on them. Deployment diagrams address the static deployment view of architecture. They are related to component diagrams in that a node typically encloses one or more components.



2.8 E-R Diagrams:

Database is absolutely an integral part of software system. To fully utilize ER Diagram in database engineering guarantee you to produce high quality database design to use in database creation, management and maintenance. An ER model also provides a means for communication.



CHAPTER 3

ANALYSIS

3.1 EXISTING SYSTEM:

Hospitals currently use a manual system for the management and maintainance of critical information. The current system requires numerous paper forms, with data stores spread through out the hospital management infrastructure. Often information is incomplete or does not follow management standards. Forms are often lost in transit between departments requiring a comprehensive auditing process to ensure that no vital information is lost. Multiple copies of the same information exist in the hospital and may lead to inconsistencies in data in various data stores.

3.2 PROPOSED SYSTEM:

The Hospital Management System is designed for any hospital to replace their existing manual paper based system. The new system is to control the information of patients. Room availability, staff and operating room schedules and patient invoices. These services are to be provided in an efficient, cost effective manner, with the goal of reducing the time and resources currently required for such tasks .

3.3 FEASIBILITY STUDY

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are:

3.3.1 Economic Feasibility

This study is carried out to check the economic impact will have on the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customised products have to be purchased.

3.3.2 Technical Feasibility

This study is carried out to check the technical feasibility, that is,the technical requirements of the system. Any system developed must not have a high demand on the available available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes for the implementing this system.

3.3.3 Operational Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3.4 SOFTWARE SPECIFICATION

HTML:

HTML or **Hypertext Markup Language** is the standard markup language used to create web pages.

HTML is written in the form of HTML elements consisting of *tags* enclosed in angle brackets (like <html>). HTML tags most commonly come in pairs like <h1> and </h1>, although some tags represent *empty elements* and so are unpaired, for example <img>. The first tag in a pair is the *start tag*, and the second tag is the *end tag* (they are also called *opening tags* and *closing tags*). Though not always necessary, it is best practice to append a slash to tags which are not paired with a closing tag.

The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presentation, making it a markup language rather than a programming language.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behavior of HTML web pages.

CASCADING STYLE SHEETS (CSS):

It is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to style web pages and interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation.

CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts.[1] This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content .

CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified. However if the author or the reader did not link the document to a specific style sheet the default style of the browser will be applied.

3.5 MySQL:

MySQL is developed, distributed, and supported by Oracle Corporation. MySQL is a database system used on the web it runs on a server. MySQL is ideal for both small and large applications. It is very fast, reliable, and easy to use. It supports standard SQL. MySQL can be compiled on a number of platforms.

The data in MySQL is stored in tables. A table is a collection of related data, and it consists of columns and rows. Databases are useful when storing information categorically.

FEATURES OF MySQL:

Internals and portability:

- Written in C and C++.
- Tested with a broad range of different compilers.
- Works on many different platforms.

- Tested with Purify (a commercial memory leakage detector) as well as with Val grind, a GPL tool.
- Uses multi-layered server design with independent modules.

Security:

- A privilege and password system that is very flexible and secure, and that enables host-based verification.
- Password security by encryption of all password traffic when you connect to a server.

Scalability and Limits:

- Support for large databases. We use MySQL Server with databases that contain 50 million records. We also know of users who use MySQL Server with 200,000 tables and about 5,000,000,000 rows.
- Support for up to 64 indexes per table (32 before MySQL 4.1.2). Each index may consist of 1 to 16 columns or parts of columns. The maximum index width is 767 bytes for **InnoDB** tables, or 1000 for **MyISAM**; before MySQL 4.1.2, the limit is 500 bytes. An index may use a prefix of a column for **CHAR**, **VARCHAR**, **BLOB**, or **TEXT** column types.

CONNECTIVITY:

Clients can connect to MySQL Server using several protocols:

- Clients can connect using TCP/IP sockets on any platform.
- On Windows systems in the NT family (NT, 2000, XP, 2003, or Vista), clients can connect using named pipes if the server is started with the --enable-named-pipe option. In MySQL 4.1 and higher, Windows servers also support shared-memory connections if started with the --shared-memory option. Clients can connect through shared memory by using the --protocol=memory option.
- On UNIX systems, clients can connect using Unix domain socket files.

LOCALIZATION:

- The server can provide error messages to clients in many languages.
- All data is saved in the chosen character set.

CLIENTS AND TOOLS:

- MySQL includes several client and utility programs. These include both command-line programs such as **mysqldump** and **mysqladmin**, and graphical programs such as MySQL Workbench.
- MySQL Server has built-in support for SQL statements to check, optimize, and repair tables. These statements are available from the command line through the **mysqlcheck** client. MySQL also includes **myisamchk**, a very fast command-line utility for performing these operations on **MyISAM** tables.
- MySQL programs can be invoked with the --help or -? option to obtain online assistance.

WHY TO USE MySQL:

- Leading open source RDBMS
- Ease of use – No frills
- Fast
- Robust
- Security
- Multiple OS support
- Free
- Technical support
- Support large database– up to 50 million rows, file size limit up to 8 Million TB

3.6 PHP:

WHAT IS PHP?

- PHP is an acronym for "PHP Hypertext Preprocessor"
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP costs nothing, it is free to download and use

WHAT IS PHP FILE?

- PHP files can contain text, HTML, CSS, JavaScript, and PHP code
- PHP code are executed on the server, and the result is returned to the browser as plain HTML
- PHP files have extension ".php"

WHAT CAN PHP DO?

- PHP can generate dynamic page content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can restrict users to access some pages on your website
- PHP can encrypt data

With PHP you are not limited to output HTML. You can output images, PDF files, and even Flash movies. You can also output any text, such as XHTML and XML.

WHY PHP?

- PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
- PHP is compatible with almost all servers used today (Apache, IIS, etc.)
- PHP supports a wide range of databases
- PHP is free. Download it from the official PHP resource: www.php.net

```
//connect to database
$db['default']['hostname'] = 'localhost';
$db['default']['username'] = 'root';
$db['default']['password'] = ";
$db['default']['database'] = 'multims';
$db['default']['dbdriver'] = 'mysql';
```

Chapter 4 :

SYSTEM IMPLEMENTATION

4.1 Login page

```
<form class="form-signin" method="post" action="auth/login">
            <h2 class="login form-signin-heading">Hospital Management
System</br></br>
<img alt="" src="uploads/favicon.png"></h2>
            <div id="infoMessage"><?php echo $message; ?></div>
            <div class="login-wrap">
               <input type="text" class="form-control" name="identity"
placeholder="User Email" autofocus>
               <input type="password" class="form-control" name="password"
placeholder="Password">
               <p><a data-toggle="modal" href="#myModal"> Forgot
Password?</a></p>
               <!--
                   <label class="checkbox">
                      <input type="checkbox" value="remember-me"> Remember me
                      <span class="pull-right">
                         <a data-toggle="modal" href="#myModal"> Forgot
Password?</a>

                      </span>
                   </label>
               -->
               <button class="btn btn-lg btn-login btn-block" type="submit">Sign
in</button>
            </div>
         </form>
```

4.2 Hospital page

```
<div class="modal-header"> <button type="button" class="close" data-
dismiss="modal" aria-hidden="true">×</button> <h4 class="modal-title"><i
class="fa fa-plus-circle"></i> Add New Hospital</h4> </div> <div
class="modal-body"> <form role="form" action="hospital/addNew"
method="post" enctype="multipart/form-data"> <div class="form-group">
<label for="exampleInputEmail1">Name</label> <input type="text"
class="form-control" name="name" id="exampleInputEmail1" value=" placeholder="">
</div> <div class="form-group"> <label
for="exampleInputEmail1">Email</label> <input type="text"
class="form-control" name="email" id="exampleInputEmail1" value=" placeholder="">
</div> <div class="form-group"> <label
for="exampleInputEmail1">Password</label> <input type="password"
class="form-control" name="password" id="exampleInputEmail1" placeholder="">
</div> <div class="form-group"> <label
for="exampleInputEmail1">Address</label> <input type="text"
class="form-control" name="address" id="exampleInputEmail1" value="
placeholder=""> </div> <div class="form-group">
<label for="exampleInputEmail1">Phone</label> <input type="text"
class="form-control" name="phone" id="exampleInputEmail1" value="
placeholder=""> </div> <input type="hidden"
name="id" value="> <button type="submit" name="submit" class="btn btn-
info">Submit</button>
```