## HOSPITAL MANAGEMENT SYSTEM

*A*

***Major Project Report***

*Submitted*

*In partial fulfillment*

*For the award of the Degree of*

## BACHELOR OF TECHNOLOGY

***In Department of Computer science and Engineering***

### Submitted By: Guided By: Submitted To:

Kavish Lodha

(18ETCCS051)  
 Nupur Thakur

Mr. Aaditya Maheshwari Head of Department

(18ETCCS091)



### Department of Computer Science and Engineering TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY

RAJASTHAN TECHNICAL UNIVERSITY

**May 2022**



### TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY



## CERTIFICATE

This is to certify that this project report **“Hospital Management System”** is the confide work of **“Thakur Nupur - 18ETCCS091 & Kavish Lodha- 18ETCCS051”** who have carried out the project work under my supervision. I approve this project for submission of the Bachelor of Technology in the **Department of Computer Science and Engineering, Techno India NJR Institute of Technology**, affiliated to Rajasthan Technical University, Kota.

##### Mr. Aaditya Maheshwari Mr. Gaurav Kumawat

**Assistant Professor Head of the Department**

Department of Computer Science Department of Computer Science



## ABSTRACT

The purpose of the project entitled as “HOSPITAL MANAGEMENT SYSTEM” is to computerize the Front Office Management of Hospital to develop software which is user friendly simple, fast, and cost – effective. It deals with the collection of patient’s information, diagnosis details, etc.

Traditionally, it was done manually. The main function of the system is register and store patient details and doctor details and retrieve these details as and when required, and also to manipulate these details meaningfully System input contains patient details, diagnosis details, while system output is to get these details on to the screen.

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 data are well protected for personal use and makes the data processing very fast.

Hospital Management System is designed for multi-speciality 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.

## ACKNOWLEDGEMENT

It gives me immense pleasure to express my deepest sense of gratitude and sincere thanks to my highly respected and esteemed guide **Mr. Aaditya Maheshwari (Assistant Professor, Project In charge), TINJRIT** for their valuable guidance, encouragement and help for completing this work. Their useful suggestions for this whole work and co-operative behavior are sincerely acknowledged.

I would like to express my sincere thanks to **faculties , Dept. of CSE TINJRIT** for giving me this opportunity to undertake this project.

I also wish to express my indebtedness to my parents as well as my family member whose blessings and support always helped me to face the challenges ahead.

At the end I would like to express my sincere thanks to all my friends and others who helped me directly or indirectly during this project work.

**Place: Udaipur Date:**

### Nupur Thakur Kavish Lodha



## TABLE OF CONTENTS

**Table of Contents**

[Abstract iv](#_gjdgxs)

[Acknowledgement v](#_30j0zll)

[Table of Contents vi](#_1fob9te)

**CHAPTER – 1 Introduction** 1-5

[Introduction 2](#_3znysh7)

* 1. [Purpose 2](#_2et92p0)
  2. [Overall Description 4](#_tyjcwt)

1.2.1. Modules………………………………………………………………………………………………....4

**CHAPTER – 2 Software Requirement Specification** 6-20

* 1. [Purpose 7](#_3dy6vkm)
  2. [Overall Description 9](#_1t3h5sf)
     1. [Functional Requirements definitions 9](#_4d34og8)
     2. [Use Cases 9](#_2s8eyo1)
  3. Requirement Specification 15
     1. [Functional Requirements 15](#_17dp8vu)
  4. Hardware Specification 19
  5. Software Specification 19
  6. Hardware and Software Requirements in detail 20

**CHAPTER – 3 System Analysis and Design** 21-39

3.1. Study & Weaknesses of current system 22

[3.2 Requirements of new system 23](#_3rdcrjn)

* 1. [Feasibility Study 23](#_26in1rg)
  2. Features of new System 24
  3. Data flow diagram 25
  4. UML Modeling 34
     1. [Activity Diagram 34](#_lnxbz9)
     2. [Class Diagram 35](#_35nkun2)
     3. [Context Diagram 36](#_1ksv4uv)
     4. Sequence Diagram 37

**CHAPTER – 4 Data Dictionary** 40-45

* 1. [Table Details 41](#_44sinio)
  2. [E-R Diagram 45](#_2jxsxqh)

**CHAPTER - 5 Screen Shots** 46-61

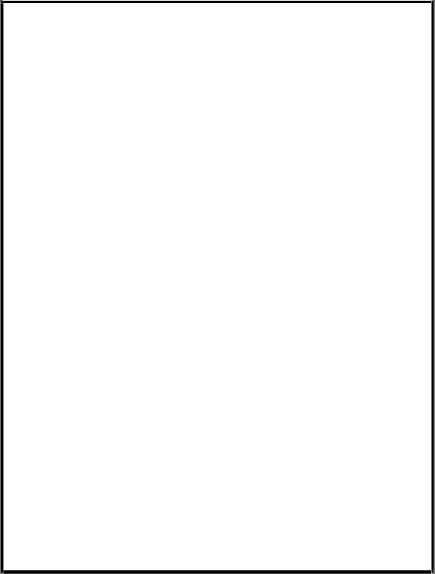
* 1. [Admin Master 47](#_z337ya)
  2. [User Master 54](#_3j2qqm3)

**CHAPTER – 6 Testing** 62-67

**CHAPTER – 7 Conclusion and Future Enhancements** 68-71

* 1. [Limitations 69](#_1y810tw)
  2. [Future Enhancements 70](#_4i7ojhp)
  3. [Conclusion 71](#_2xcytpi)

**CHAPTER – 8 Bibliography** 72







# CHAPTER – I INTRODUCTION



### Introduction

##### Purpose

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 multi-speciality 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





##### Overall Description

This is a hospital management system that is aimed to make it easier to scale up a hospital’s efficiency by easy management for the patients, the doctors and other hospital staff, avoiding wastage of resources and time, and most importantly better management of data which is a vital component because often, crucial details of patients get all disoriented in the paperwork and it affects the quality of care and treatment that the patient is receiving.

Healthcare is a major aspect of our society and many healthcare providers face a lot of problems providing active and efficient solutions to their patients.

Multi-specialty hospitals have numerous people entering and exiting the hospital on daily basis including patients, doctors, visitors etc. and maintaining their records securely is a very difficult task.

This 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.

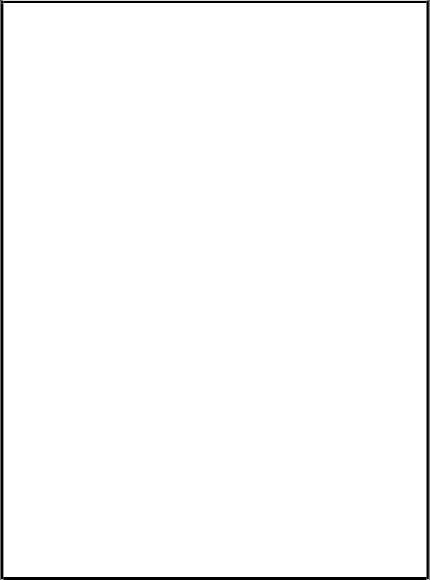
###### Modules

This system will be used in Three User Modules which are Admin, Doctor and Patient, as all of these have different requirements the modules are designed to meet their needs and avoid any type of confusion. The Uses of all three User Modules have been described below.

1. User can do the following functions in the Admin Module:

* View registered patients
* View registered doctors
* View appointments list
* Adding doctors to the module
* View user’s feedback/queries

1. User can do the following functions in the Patient Module:

* Book appointments
* View appointment history
* Bill Generation

1. Users can do the following functions in the Doctor Module:

* View appointments
* View/Write Prescriptions



# CHAPTER – II

***SOFTWARE REQUIREMENT SPECIFICATION***



### Software Requirement Specification

##### Purpose

###### Introduction

###### To be used efficiently, all computer software needs certain hardware components or the other software resources to be present on a computer. These pre-requisites are known as(computer)system requirements and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements: minimum and recommended. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time. Industry analysts suggest that this trend plays a bigger part in driving upgrades to existing computer systems than technological advancements.

###### Software Requirements deal with defining software resource requirements and pre-requisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or pre-requisites are generally not included in the software installation package and need to be installed separately before the software is installed

###### Scope

Information is recorded into patients' files as and when it's necessary, e.g., once in an appointment, Doctors can have a look upon it. This means records are not kept forever, rather they are only retained for the time deemed necessary, so there is a lot less paperwork to go into the trash cans each day or week!

Hospital management system allows us to optimize and digitize all the processes within the institution, which in return helps us improve customer service, efficient of caregiving, reduce process costs, streamline the search of medical records, bills, helps maintain the profiles of doctors and patients etc.

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.





##### Overall description

This is a hospital management system that is aimed to make it easier to scale up a hospital’s efficiency by easy management for the patients, the doctors and other hospital staff, avoiding wastage of resources and time, and most importantly better management of data which is a vital component because often, crucial details of patients get all disoriented in the paperwork and it affects the quality of care and treatment that the patient is receiving. This 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.

Hospital Management system allows us to optimize and digitize all the processes within the institution, which in return helps us improve customer service, efficient of caregiving, reduce process costs, streamline the search of medical records, bills, helps maintain the profiles of doctors and patients etc.

###### Functional requirements definitions

Functional Requirements are those that refer to the functionality of the system, i.e., what services it will provide to the user. Nonfunctional (supplementary) requirements pertain to other information needed to produce the correct system and are detailed separately.

###### Use cases

This system will be used in Three User Modules which are Admin, Doctor and Patient, as all of these have different requirements the modules are designed to meet their needs and avoid any type of confusion. The Uses of all three User Modules have been described below.

1. User can do the following functions in the Admin Module:

* View registered patients
* View registered doctors
* View appointments list
* Adding doctors to the module
* View user’s feedback/queries

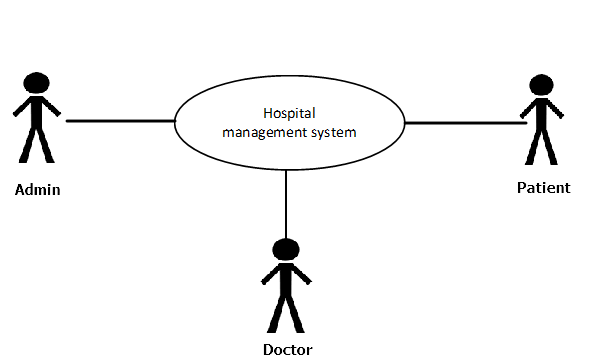
1. User can do the following functions in the Patient Module:

* Book appointments
* View appointment history
* Bill Generation

1. Users can do the following functions in the Doctor Module:

* View appointments
* View/write Prescriptions

2.2.2a Use Case: Access Home Page



##### Fig. 2.1 Access Home Page

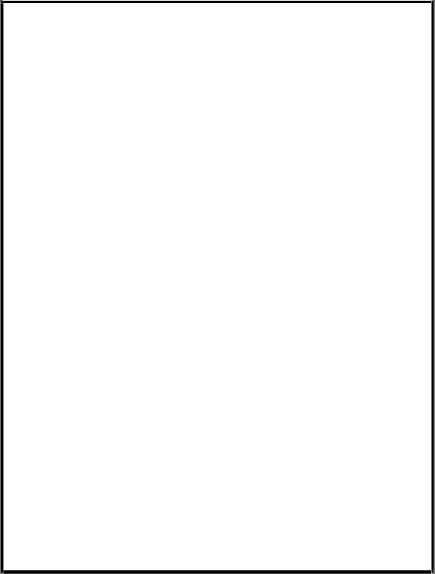
Brief Description:

User uses the hospital management system to access the home page.

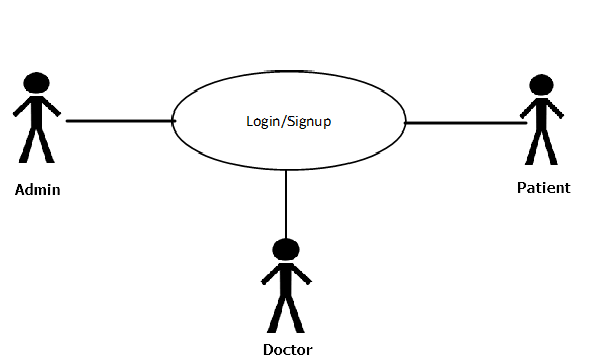
Initial step-by-step description:

For this use case to be initiated, the user (Patient or Admin or Doctor) can use the online hospitable management system.

1. The user connects to the system using a web browser.
2. The user selects the home link on the website home page.
3. The system passes the user to the website Home Page.



2.2.2b. Use Case: User Login or Signup



##### Fig. 2.2 User Login or Signup

Brief Description:

The user either logs in or signs up.

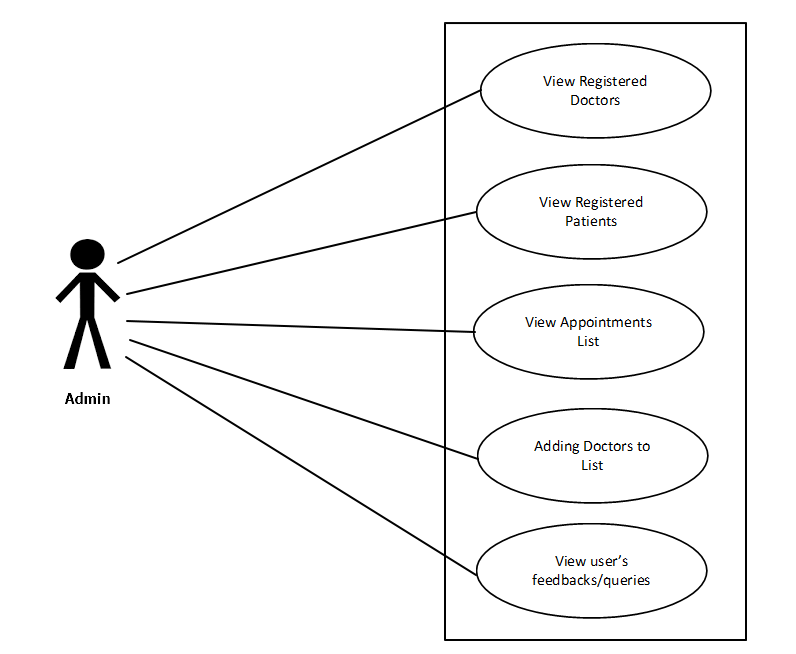
Initial step-by-step description:

For this use case to be initiated the user must be on the website Home Page.

1. The User puts in the credentials.
2. If the user is already registered, he successfully gets logged in.
3. Else user has to sign up or register.
4. The hospital management system returns the signup form.
5. The User fills in the form.
6. The User clicks submit.
7. The System retains information in the database.
8. The System returns the user to the website Home Page.



2.2.2c. Use Case: Create new category/subcategory/subject/topic/question



##### Fig. 2.3 Admin Selects any of the above actions

Brief Description:

The admin chooses to create a new category/subcategory/subject/topic/question.

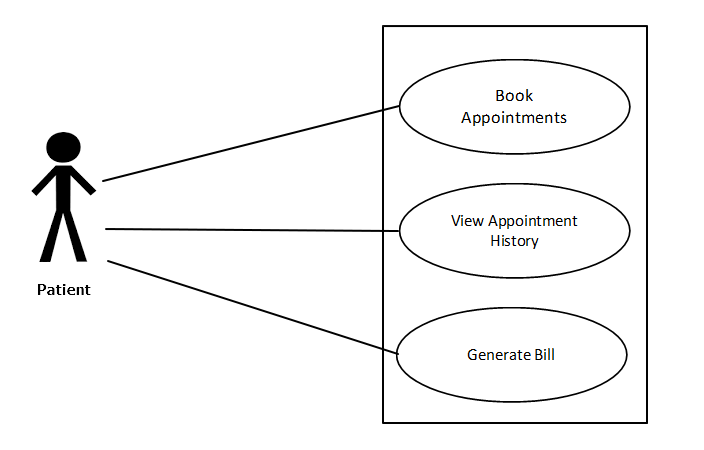
Initial step-by-step description:

For this use case to be initiated, the admin must be on the website of the hospital management system.

1. The admin selects the option they want to perform action on.
2. The website returns the corresponding page.
3. The admin fills in the entries.



1. The admin clicks submit.
2. The system adds the data to the Database.
3. The system returns the Admin to the Admin Home Page.
   * 1. d. Use Case: Patient selects the action



##### Fig. 2.4 Patient Selects the action

Brief Description:

The patient chooses any action they want to perform.

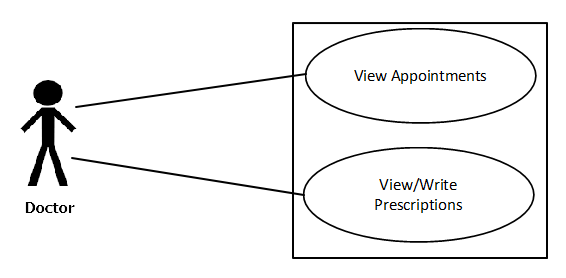
Initial step-by-step description:

For this use case to be initiated, the patient must be logged in.

* + - 1. The patient selects the action they want to perform.



* + - 1. The website returns the corresponding page.
      2. Then the patient can make the changes required to be made.
      3. The system adds the data to the Database.
      4. The system returns the patient to the patient welcome Page.
    1. Use Case: Doctor selects the action



##### Fig. 2.4 Doctor Selects the action

Brief Description:

The doctor chooses any action they want to perform.

Initial step-by-step description:

For this use case to be initiated, the doctor must be logged in.

* + - 1. The doctor selects the action they want to perform.
      2. The website returns the corresponding page.
      3. The system adds the data to the Database.
      4. The system returns the Admin to the Admin Home Page.



##### Requirement specifications

* + 1. ***Functional Requirements***

##### Table 2.2 Access Home Page

| **Use Case Name:** | Access Home Page |
| --- | --- |
| **Priority** | Essential |
| **Trigger** | Menu selection |
| **Precondition** | User is on the home page. |
| **Basic Path** | 1. The user connects to the hospital management system using a web browser. 2. The user selects the home link on the website home page. 3. The system passes the user to the   website Home Page. |
| **Alternate Path** | N/A |
| **Postcondition** | The User is on the Home Page |
| **Other** |  |

**Table 2.3. Patient Login or Signup**

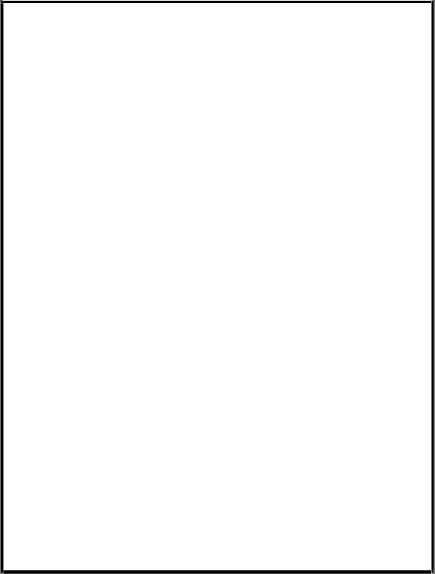
| **Use Case Name:** | Patient Login or Signup |
| --- | --- |
| **Priority** | Essential |
| **Trigger** | Selects |
| **Precondition** | The User is on the Home Page |

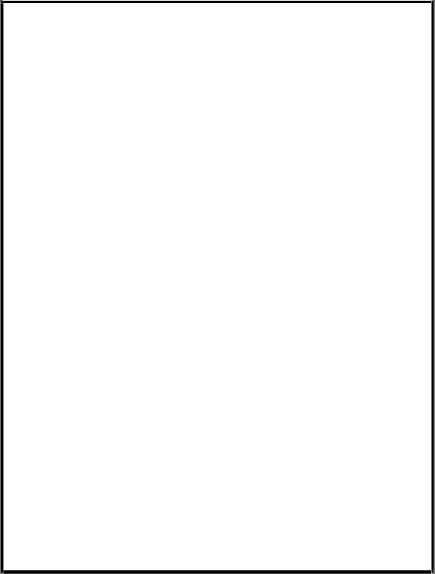


| **Basic Path** | 1. The Patient selects the “Login” option. 2. If the user is already registered, he successfully gets logged in. 3. Else user has to sign up. 4. The hospital management system returns to registration page. 5. The User fills in the form. 6. The User clicks register. 7. The System retains information in the database. 8. The System returns the user to the   website Home Page. |
| --- | --- |
| **Postcondition** | The user is on the home page. |
| **Other** |  |

##### Table 2.4 Patient performs an action

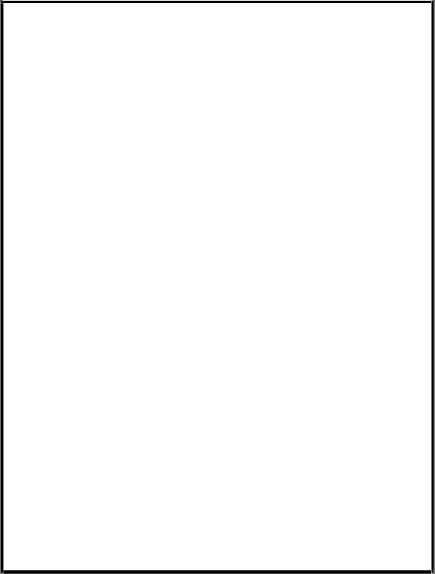
| **Use Case Name:** | Patient selects what action they want to perform. |
| --- | --- |
| **Priority** | Essential |
| **Trigger** | Menu selection |
| **Precondition** | The Patient must be logged in for the used case. |
| **Basic Path** | 1. Patient selects what action they want to perform. 2. The website returns the corresponding page. 3. The website returns the corresponding page. 4. Then the patient can make the changes required to be made. 5. The system adds the data to the Database. 6. The system returns the Patient to the patient welcome Page. |
| **Alternate Path** | N/A |
| **Postcondition** | A record is created or updated in the related Table of the Database. |
| **Exception Path** | 1. If the connection is terminated before the form is submitted, the fields are cleared and the website is returned to the wait state. 2. If the connection is terminated after the form is submitted, but before the Admin   is returned to the Admin Home Page, the |
|  | record is created in the Table of the  Database. |
| **Other** |  |





**Table 2.5 The doctor performs an action**

| **Use Case Name:** | The doctor chooses any action they want to perform |
| --- | --- |
| **Priority** | Essential |
| **Trigger** | Menu selection |
| **Precondition** | The doctor must be logged in and on the hospital management system page. |
| **Basic Path** | * + - 1. The doctor selects the action they want to perform.       2. The website returns the corresponding page.       3. The system adds the data to the Database.       4. The system returns the doctor to the doctor welcome Page. |
| **Alternate Path** |  |
| **Postcondition** | The student will get the result and it is stored in the database. |
| **Exception Path** | 1. If the connection is terminated before the form is submitted, the fields are cleared and the website is returned to the wait  state. |
| **Other** |  |



##### Hardware Specification

##### Client Side:

* Google 101.0/ Fire Fox 91.0/ internet explorer 11
* Processor: Intel(R) Core(TM) i3.
* RAM : 4 GB
* Hard Disk : 80GB

##### Server Side:

* Processor: Intel(R) Core(TM) i3
* RAM: 4 GB
* Disk space: 4GB

##### Software Specification Client Side:

* Bootstrap Framework
* Web Browser
* Google 101.0/ Fire Fox 91.0/ internet explorer 11

##### Web Server:

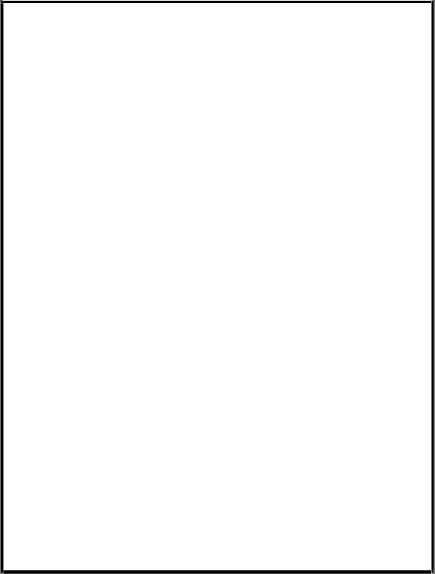
* Bootstrao Framework
* Windows 10

##### Data Base Server:

* MySQL Server

##### Hardware and Software Requirements in detail Hardware Requirements:

* Processor: Intel(R) Core(TM) i3.
* Google 101.0/ Fire Fox 91.0/ internet explorer 11
* RAM : 512 MB
* Hard Disk : 4 GB
* Disk space : 4GB



##### Software Requirements:

* Microsoft Visual Studio 2010
  + Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It can be used to develop console and graphical user interfaceapplications along with Windows Forms applications, web sites, web applications, and web services in both native code together with managed code for all platforms supported by Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework and Microsoft Silverlight.
  + .NET Framework 3.5
  + Visual C#
* Web Browser Internet Explorer 6.0 and above
* Windows 10



# CHAPTER – III

***SYSTEM ANALYSIS AND DESIGN***



### System Analysis and Design

##### Current System

Hospitals currently use a manual system for the management and maintenance of critical information. The current system requires numerous paper forms, with data stores spread throughout 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.

##### Proposed System

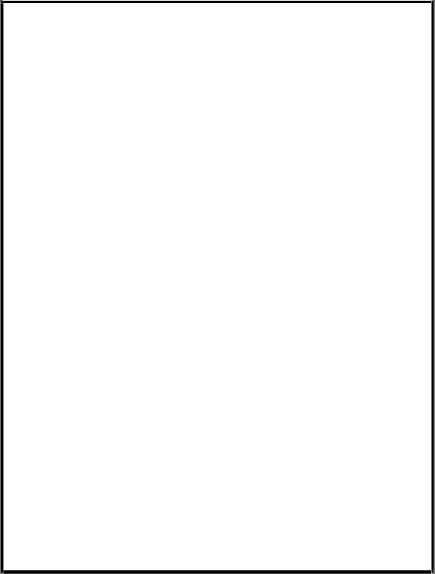
The Hospital Management System is designed for any hospital to replace their existing manual paper-based system. The new system is to control the information of patients. Room availability, staff and operating room schedules and patient invoices. These services are to be provided in an efficient, cost-effective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

##### FEASIBILITY STUDY

The feasibility of the project is analyzed 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 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.



##### SOFTWARE SPECIFICATION

**HTML:**

**HTML or Hypertext Markup Language** is the standard [markup language](http://en.wikipedia.org/wiki/Markup_language) used to create [web pages](http://en.wikipedia.org/wiki/Web_page).

HTML is written in the form of [HTML elements](http://en.wikipedia.org/wiki/HTML_element) consisting of tags enclosed in [angle brackets](http://en.wikipedia.org/wiki/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](http://en.wikipedia.org/wiki/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](http://en.wikipedia.org/wiki/Semantic) along with cues for presentation, making it a [markup language](http://en.wikipedia.org/wiki/Markup_language) rather than a [programming language](http://en.wikipedia.org/wiki/Programming_language).

HTML elements form the building blocks of all [websites](http://en.wikipedia.org/wiki/Website). HTML allows [images and objects](http://en.wikipedia.org/wiki/Img_(HTML_element)) to be embedded and can be used to create [interactive forms](http://en.wikipedia.org/wiki/Fieldset). It provides a means to create [structured documents](http://en.wikipedia.org/wiki/Structured_document) by denoting structural [semantics](http://en.wikipedia.org/wiki/Semantic) for text such as headings, paragraphs, lists, [links](http://en.wikipedia.org/wiki/Hyperlink), quotes and other items. It can embed [scripts](http://en.wikipedia.org/wiki/Scripting_language) written in languages such as [JavaScript](http://en.wikipedia.org/wiki/JavaScript) which affect the behavior of HTML web pages.

**CASCADING STYLE SHEETS (CSS):**

It is a [style sheet language](http://en.wikipedia.org/wiki/Style_sheet_language) used for describing the [look and formatting](http://en.wikipedia.org/wiki/Presentation_semantics) of a document written in a [markup language](http://en.wikipedia.org/wiki/Markup_language). While most often used to style [web pages](http://en.wikipedia.org/wiki/Web_page) and [interfaces](http://en.wikipedia.org/wiki/Interface_(computing)) written in [HTML](http://en.wikipedia.org/wiki/HTML) and [XHTML](http://en.wikipedia.org/wiki/XHTML), the language can be applied to any kind of [XML](http://en.wikipedia.org/wiki/XML) document, including [plain XML](http://en.wikipedia.org/wiki/Plain_Old_XML), [SVG](http://en.wikipedia.org/wiki/Scalable_Vector_Graphics) and [XUL](http://en.wikipedia.org/wiki/XUL). CSS is a cornerstone specification of [the web](http://en.wikipedia.org/wiki/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](http://en.wikipedia.org/wiki/Page_layout), [colors](http://en.wikipedia.org/wiki/Color), and [fonts](http://en.wikipedia.org/wiki/Typeface).[[1]](http://en.wikipedia.org/wiki/Cascading_Style_Sheets#cite_note-1) This separation can improve content [accessibility](http://en.wikipedia.org/wiki/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](http://en.wikipedia.org/wiki/Screen_reader)) and on [Braille-based](http://en.wikipedia.org/wiki/Braille_display), 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.

**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:

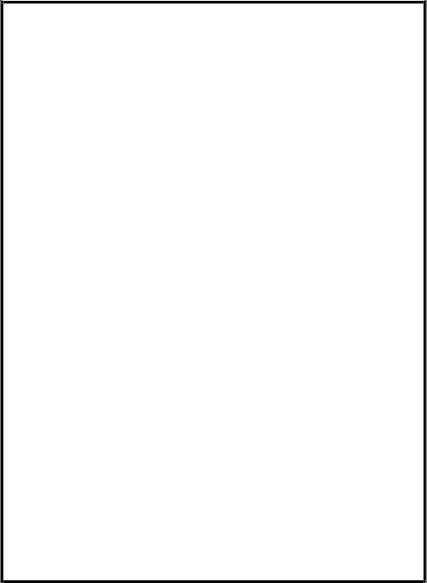
Written in C and C++.

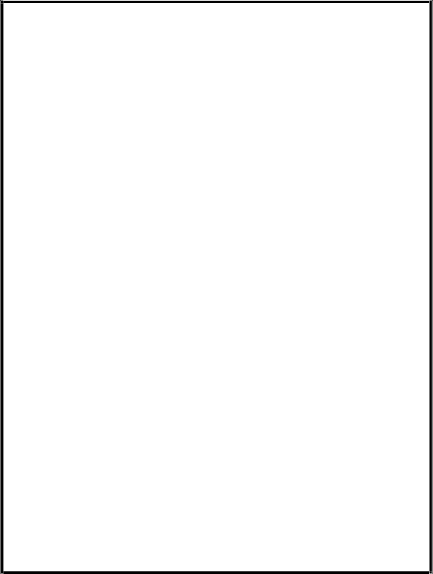
Tested with a broad range of different compilers.

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](https://dev.mysql.com/doc/refman/5.0/en/char.html), [VARCHAR](https://dev.mysql.com/doc/refman/5.0/en/char.html), [BLOB](https://dev.mysql.com/doc/refman/5.0/en/blob.html), or [TEXT](https://dev.mysql.com/doc/refman/5.0/en/blob.html) 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](https://dev.mysql.com/doc/refman/5.0/en/server-options.html#option_mysqld_enable-named-pipe) option. In MySQL 4.1 and higher, Windows servers also support shared-memory connections if started with the [--shared-memory](https://dev.mysql.com/doc/refman/5.0/en/server-options.html#option_mysqld_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:**

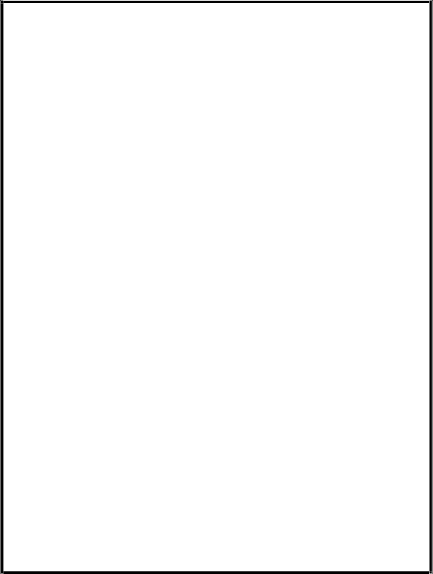
MySQL includes several client and utility programs. These include both command-line programs such as [**mysqldump**](https://dev.mysql.com/doc/refman/5.0/en/mysqldump.html) and [**mysqladmin**](https://dev.mysql.com/doc/refman/5.0/en/mysqladmin.html), and graphical programs such as [MySQL Workbench](http://dev.mysql.com/doc/refman/5.1/en/workbench.html).

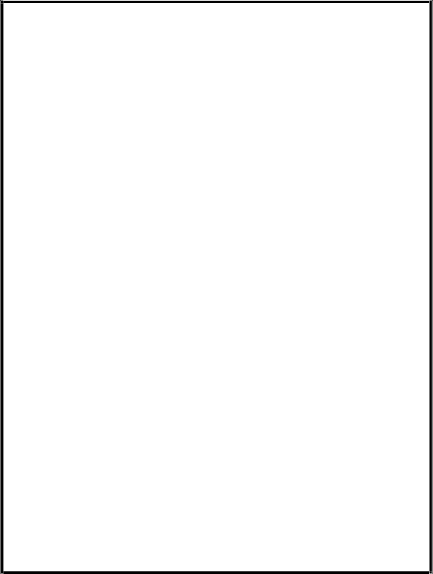
MySQL Server has built-in support for SQL statements to check, optimize, and repair tables. These statements are available from the command line through the [**mysqlcheck**](https://dev.mysql.com/doc/refman/5.0/en/mysqlcheck.html) client. MySQL also includes [**myisamchk**](https://dev.mysql.com/doc/refman/5.0/en/myisamchk.html), 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



**JAVASCRIPT:**

JavaScript is the scripting language of the Web. All modern HTML pages are using JavaScript. A scripting language is a lightweight programming language.JavaScript code can be inserted into any HTML page, and it can be executed by all types of web browsers. JavaScript is easy to learn.

**WHY TO USE JAVASCRIPT**:

JavaScript is one of the 3 languages all web developers  must learn:

* HTML to define the content of web pages
* CSS to specify the layout of web pages
* JavaScript to specify the behavior of web pages

**Example**

x = document.getElementById("demo");  //Find the HTML element with id="demo"  
x.innerHTML = "Hello JavaScript";     //Change the content of the HTML element

**document.getElementById()** is one of the most commonly used HTML DOM methods.

**OTHER USES OF JAVASCRIPT**:

* Delete HTML elements
* Create new HTML elements
* Copy HTML elements
* In HTML, JavaScript is a sequence of statements that can be executed by the web browser.

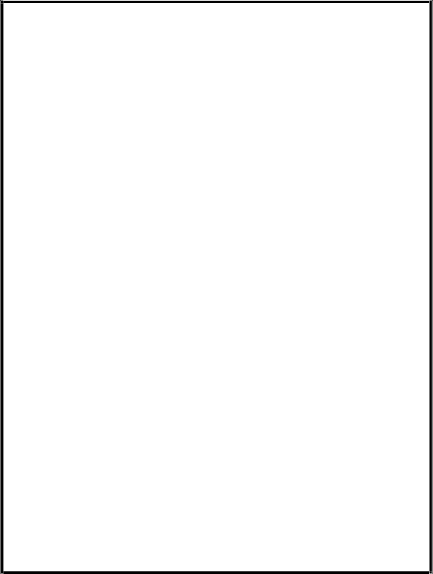
**JAVASCRIPT STATEMENTS**:

* JavaScript statements are "commands" to the browser.
* The purpose of the statements is to tell the browser what to do.
* This JavaScript statement tells the browser to write "Hello Dolly" inside an HTML element with id="demo":

Semicolon;

* Semicolon separates JavaScript statements.
* Normally you add a semicolon at the end of each executable statement.
* Using semicolons also makes it possible to write many statements on one line.

**JAVASCRIPT CODE:**

* JavaScript code (or just JavaScript) is a sequence of JavaScript statements.
* Each statement is executed by the browser in the sequence they are written.
* This example will manipulate two HTML elements:
* Example
* document.getElementById("demo").innerHTML="Hello Dolly";  
  document.getElementById("myDIV").innerHTML="How are you?";

## JAVASCRIPT PROPERTIES:

* Properties are the values associated with a JavaScript object.
* A JavaScript object is a collection of unordered properties.
* Properties can usually be changed, added, and deleted, but some are read only.

**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?**

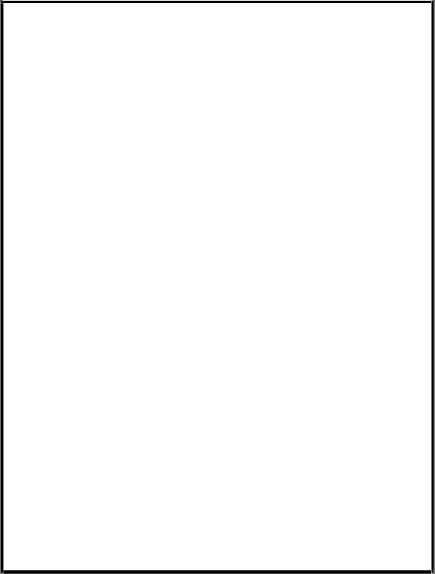
* PHP files can contain text, HTML, CSS, JavaScript, and PHP code
* PHP code are executed on the server, and the result is returned to the browser as plain HTML
* PHP files have extension ".php"

**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](http://www.php.net/)



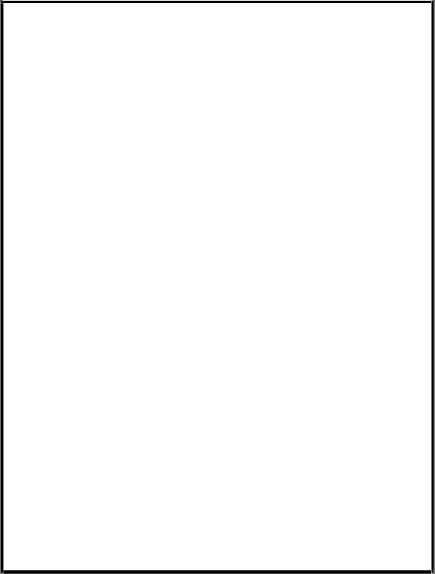


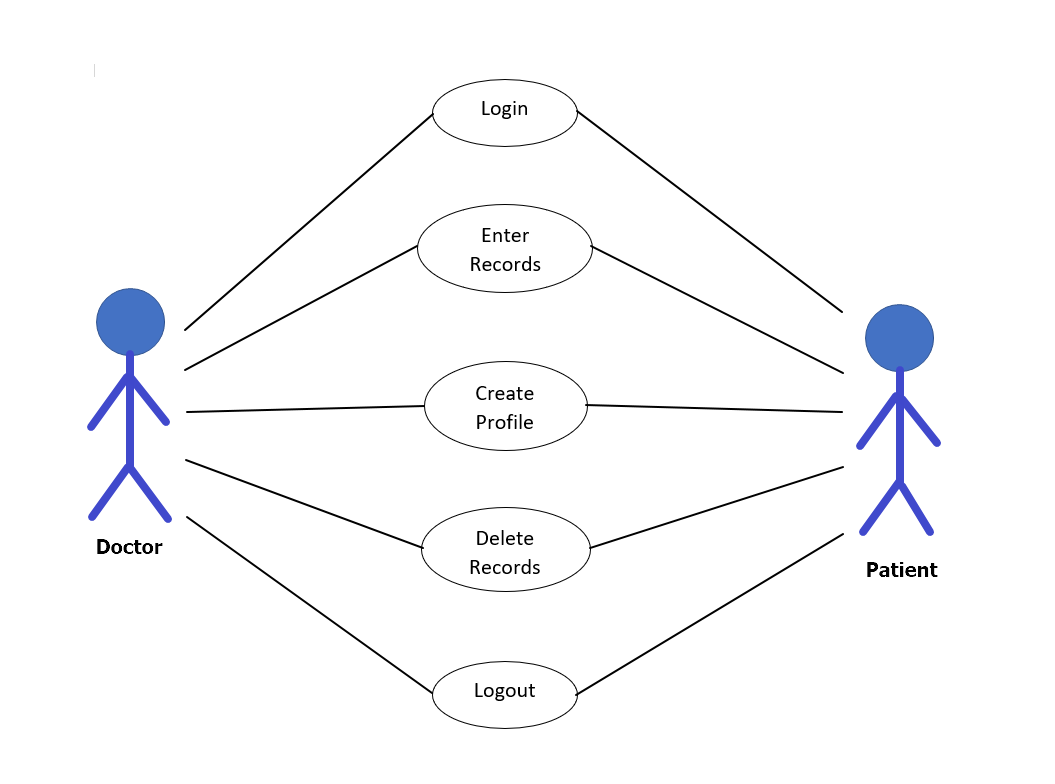
* 1. **Use Case Diagram**

**3.5.1. Level -0**

##### 

**Fig. 3.12. Level-0 Use case diagram**

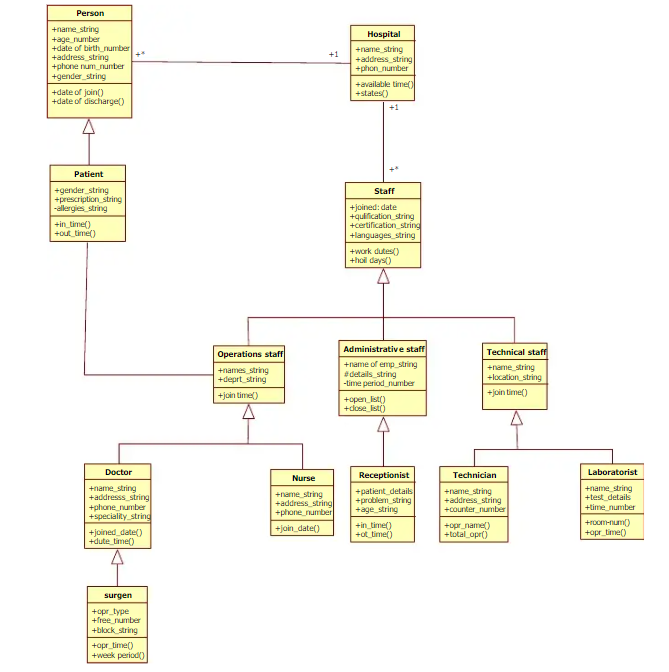
**3.5.2. Level-1**

****

**Fig. 3.12. Level-1 Use case diagram**



##### 3.5.2. Class Diagram

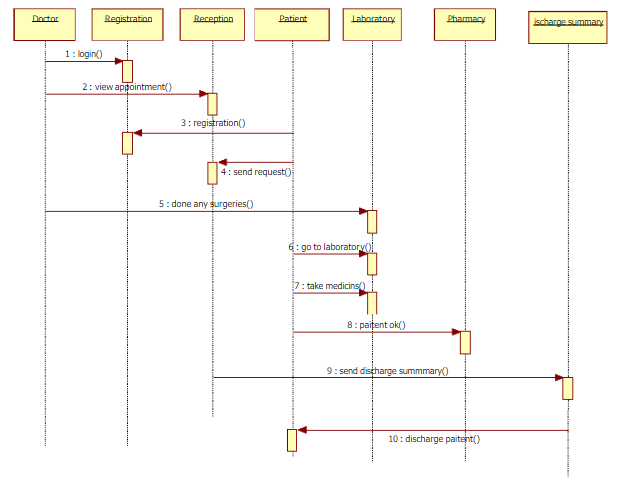


**Fig. 3.14 Class Diagram**





* + 1. **Sequence Diagrams**



##### 



# CHAPTER – IV DATA DICTIONARY



### Data Dictionary

A data dictionary is a catalog-a-repository of the elements in a system. As the name suggests, their elements center on data and the way they are structured to meet user requirements and organization needs. In a data dictionary you will find a list of all the elements composing the data flowing through a system. The major elements are data flows, data stores and processes. The data dictionary stores details and descriptions of these elements.

If analysis want to know characters are in a data item by what other names it is referenced in the system, or where it is referenced in the system, or where it is issued in the system, they should be able to find the answers in issued in the system, they should be able to find the answer in properly developed data dictionary.

The Dictionary contains two types of description for the data following through the system.

##### Data Elements

The most fundamental data is the elements. They are building blocks for all other data in the system. Data elements are also alternatively known as fields, data item or elementary item.

##### Data Structure

A data structure is a set if items that are related to one another and described a components in the system.

### Table Details

##### Table 4.1. Admin Login

| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| --- | --- | --- | --- | --- |
| Username | Unique username  of the user |  | 50 | varchar |
| Password | User password |  | 50 | Varchar |



##### Table 4.5. Patient Login

| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| --- | --- | --- | --- | --- |
| Email-id | Unique Email-id  of the user | Not null | 50 | Varchar |
| Password | User Password |  | 50 | Varchar |

##### Table 4.6. Patient Login

| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| --- | --- | --- | --- | --- |
| Username | Unique Username  of the user | Not null | 50 | Varchar |
| Password | User Password |  | 50 | Varchar |



**Table 4.6. Member Details**

| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| --- | --- | --- | --- | --- |
| First Name | First Name of the user |  | 50 | Varchar |
| Last Name | Last Name of the user |  | 50 | Varchar |
| Phone | Phone number of the user |  | 10 | int |
| Email-id | Unique Email-id of the user |  | 50 | Varchar |
| Password | Unique  Username |  | 50 | Varchar |



# CHAPTER – V SCREEN SHOTS

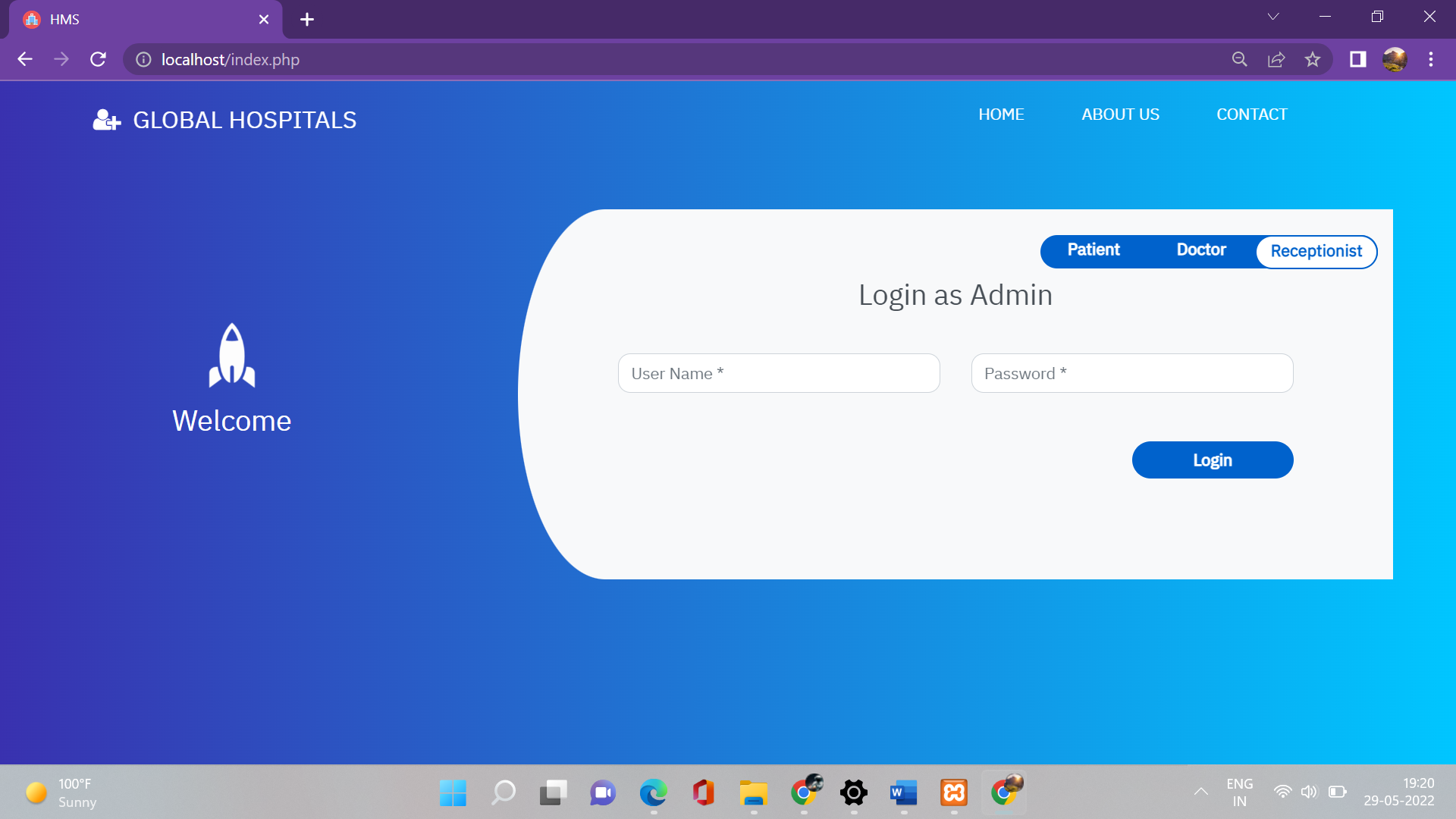


### Screen Shots

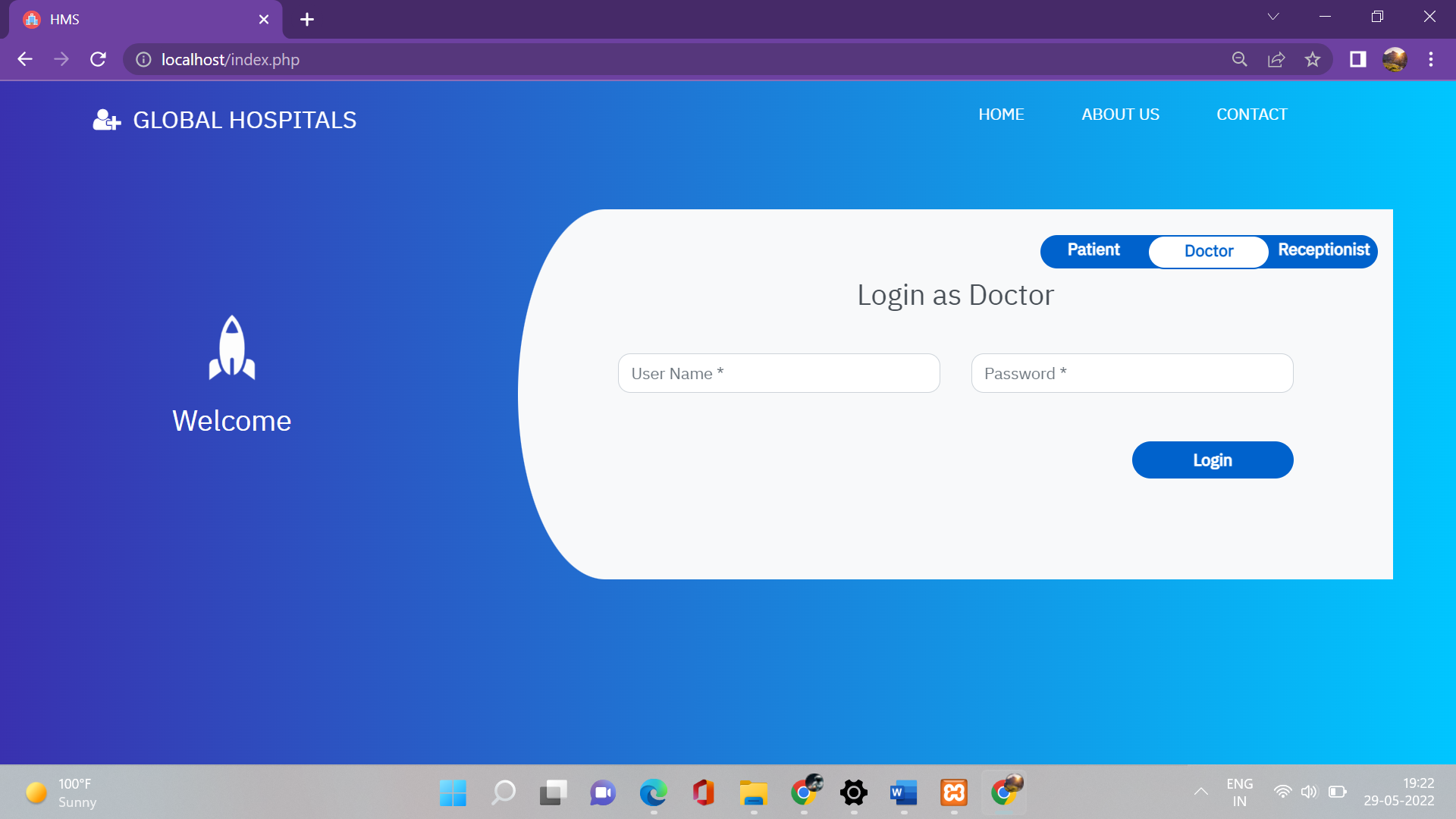
### Patient Login

### image

* + 1. **Administration Login Page**

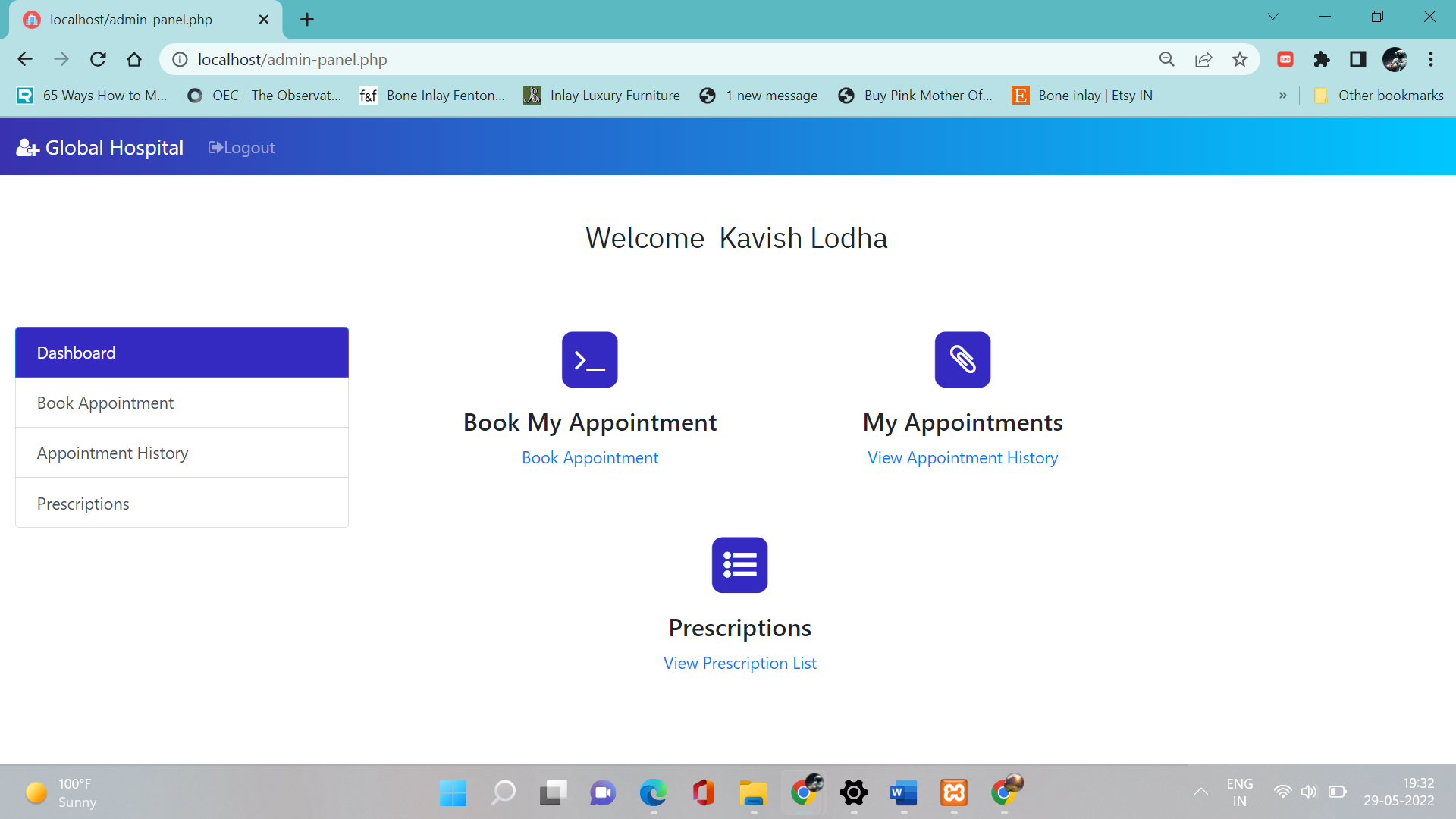


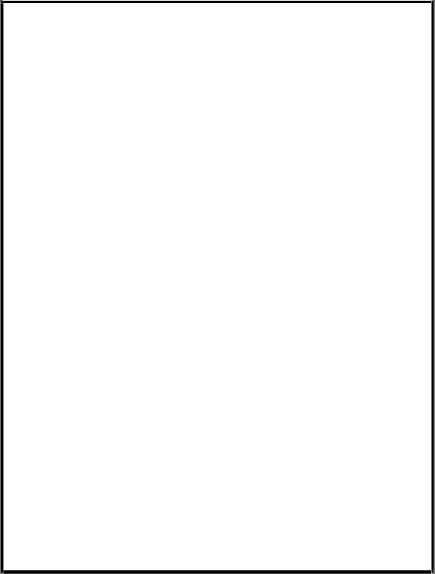
### Doctor Login





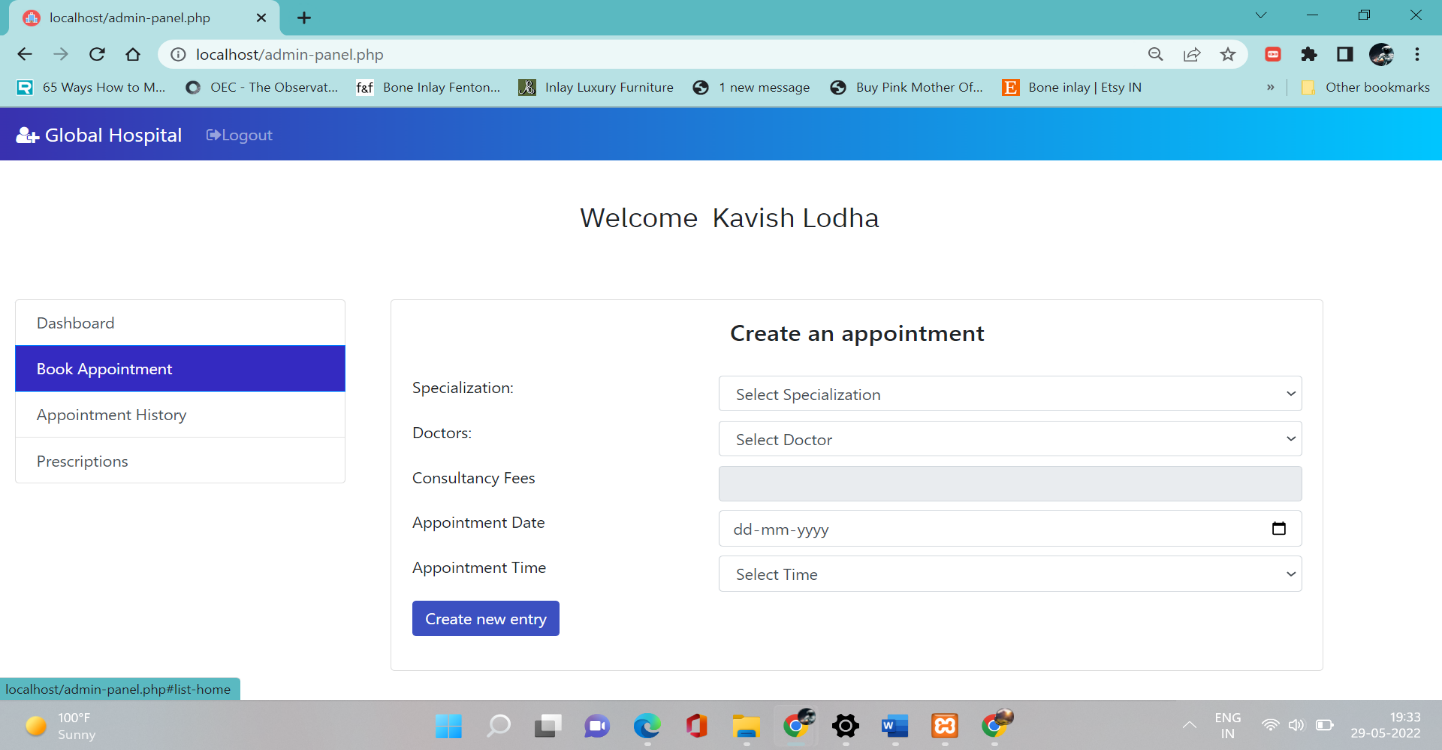
* + 1. **Home Page**



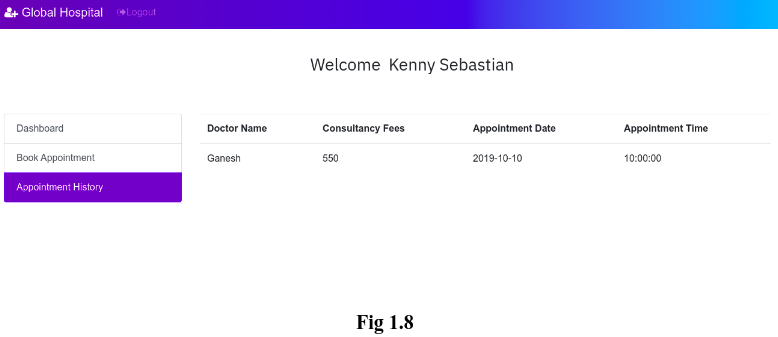


### Patient Module

* + - 1. **Book Appointment**



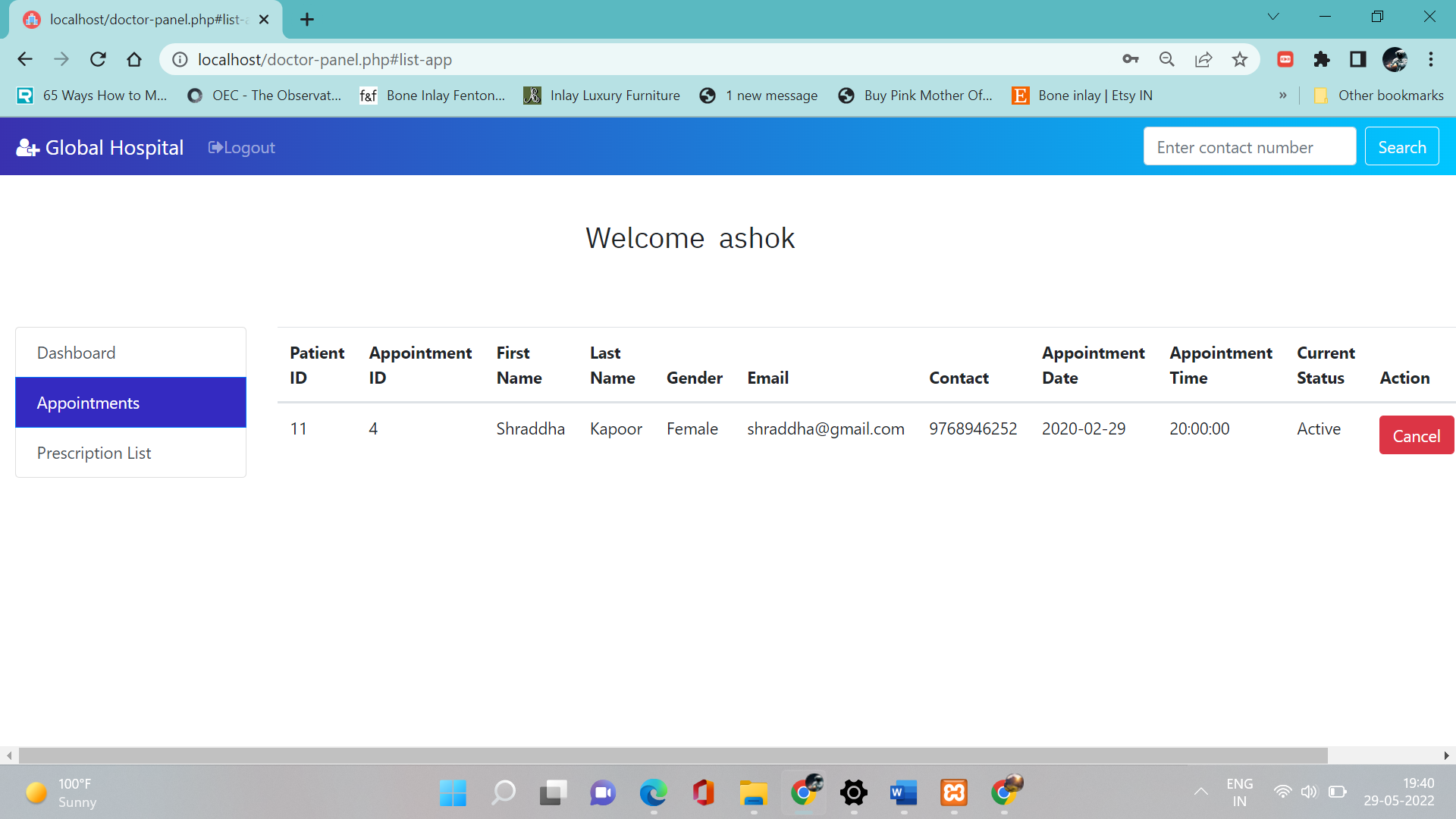
* + - 1. **View Appointment**





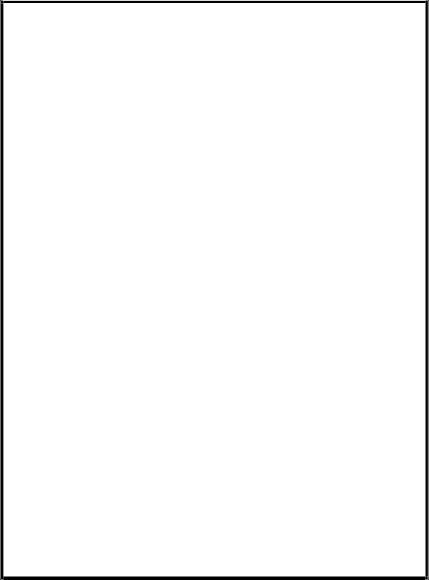
* + 1. **Doctor Module**

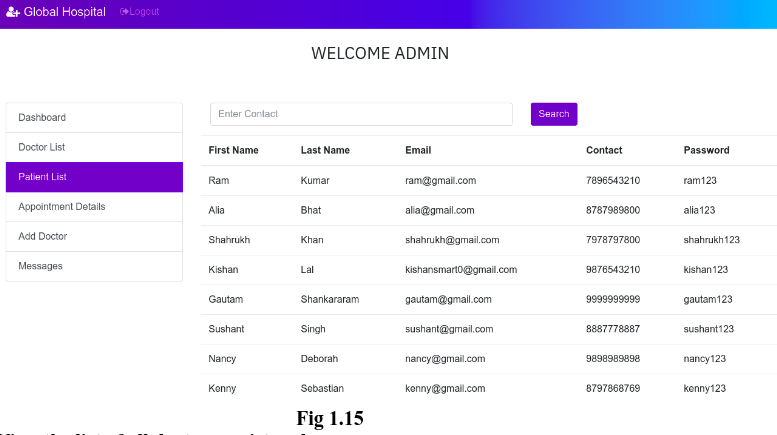
1. **View Appointment**



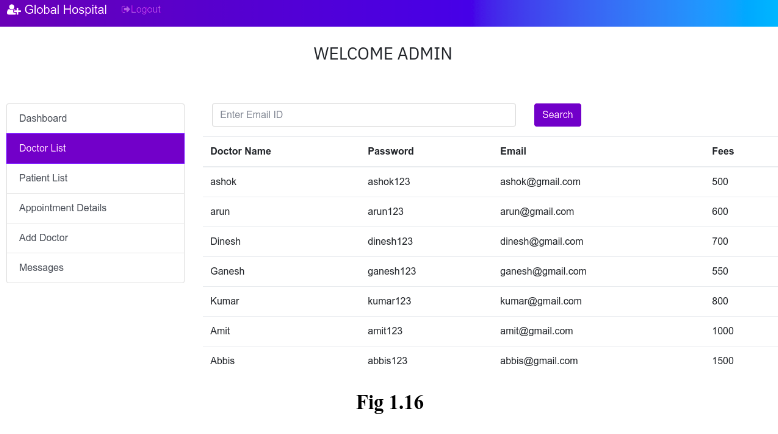
### View Prescriptions

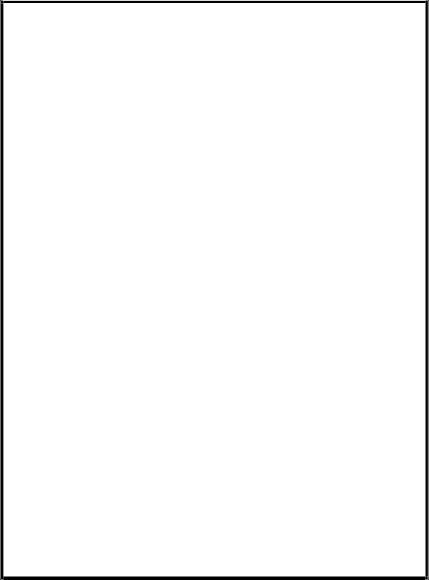


* + 1. **Admin Module**
       1. **View lists of Patients**

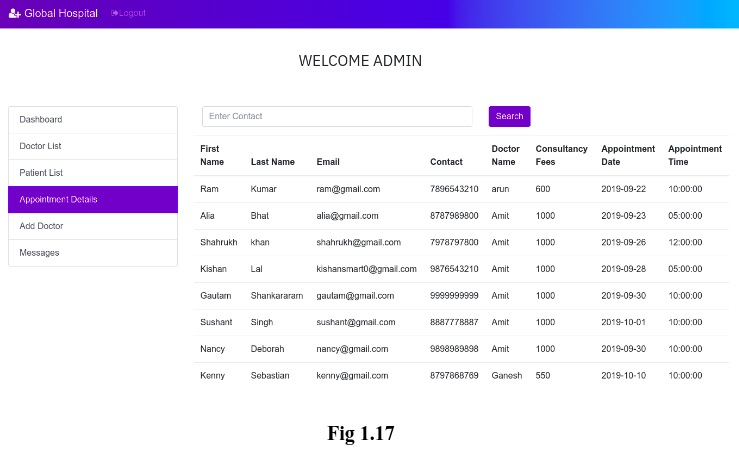


* + - 1. **View all the doctors registered**

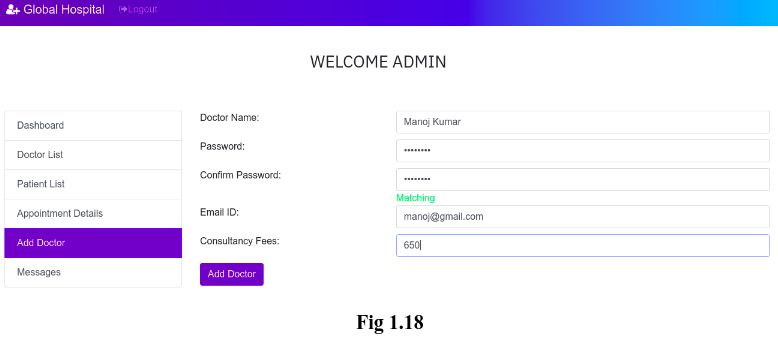


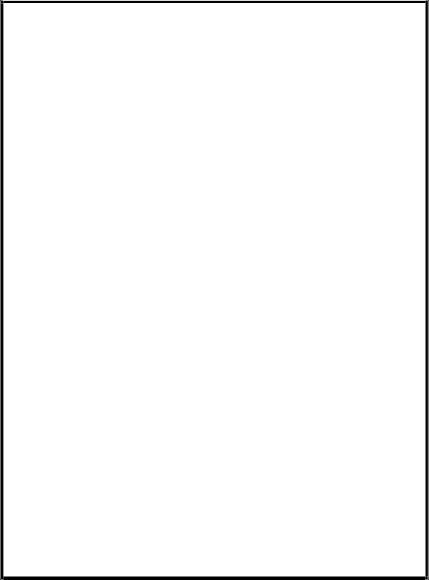


* + - 1. **View the Appointments**

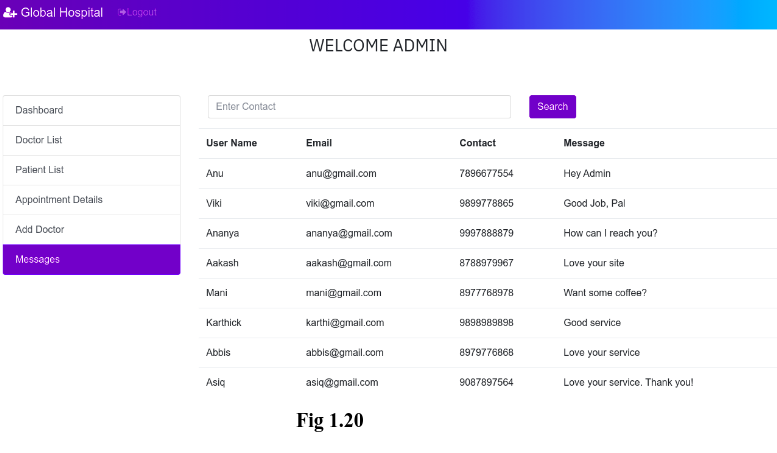


* + - 1. **Adding Doctors**

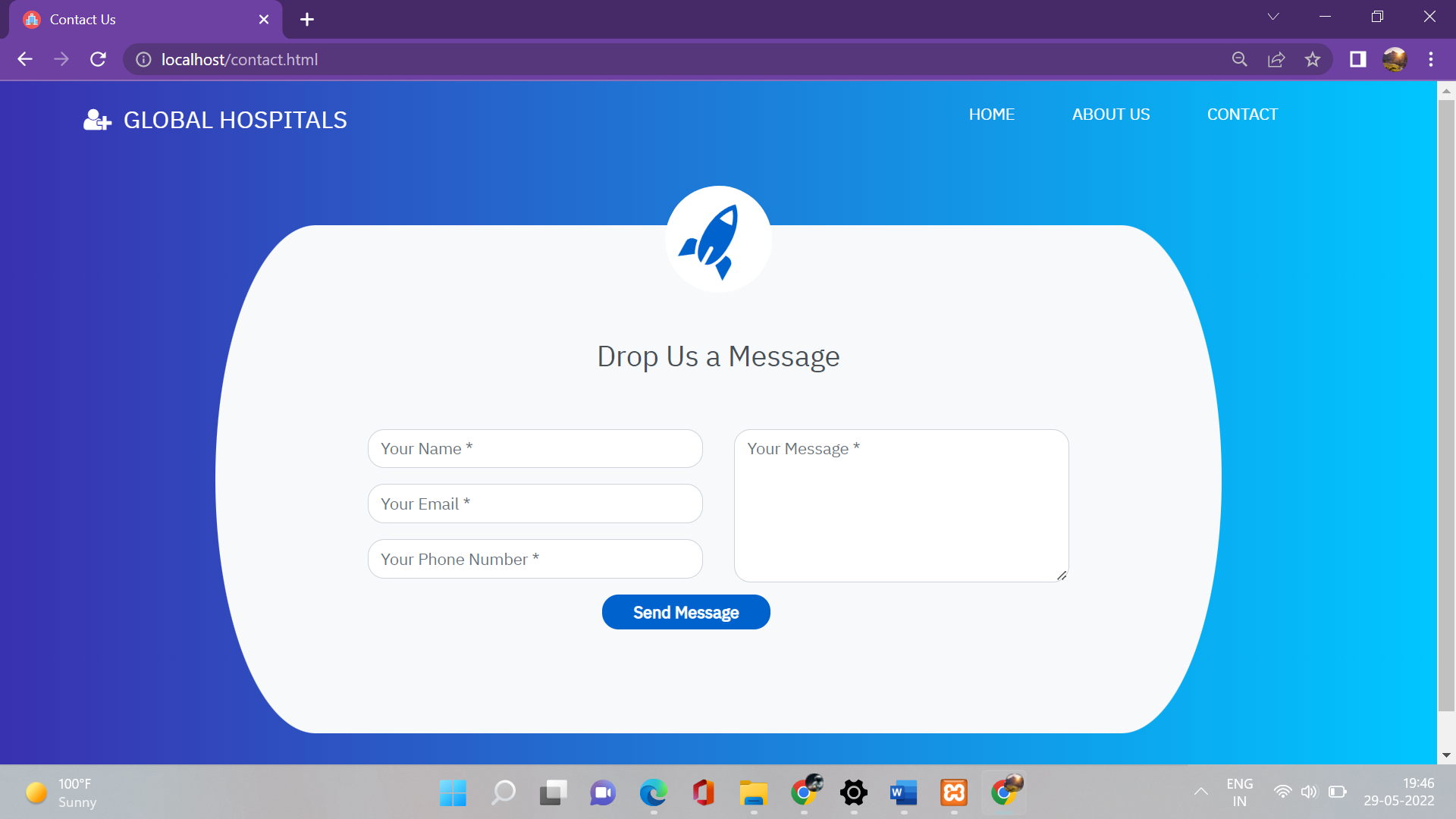




* + - 1. **View user’s queries/feedback**



* + 1. **Contact Us**







# CHAPTER – VI TESTING



### Testing

##### Testing Methodology

Companies rely on software more than ever to provide and manage information with strategic and operational importance and to provide key decision support. Rising customer expectations for fault-free, requirements-exact software have increased awareness of the importance of software testing as a critical activity.

We begin the testing process by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The process very

fies that the application meets the requirements specified in the system requirements document and is bug free. At the end of each testing day, we prepare a summary of completed and failed tests. Applications are not allowed to launch until all identified problems are fixed. A report is prepared at the end of testing to show exactly what was tested and to list the final outcomes.

Our software testing methodology is applied in three distinct phases: unit testing, system testing, and acceptance were testing.

**Unit Testing**: The programmers conduct unit testing during the development phase. Programmers can test their specific functionality individually or with other units. However, unit testing is designed to test small pieces of functionality rather than the system as a whole. This allows the programmers to conduct the first round of testing to eliminate bugs before they reach the testing staff. In unit testing the analyst tests the programs making up a system.

For this reason, unit testing is sometimes called program testing. Unit testing gives stress on the modules independently of one another, to find errors. This helps the tester in detecting errors in coding and logic that are contained within that module alone. The errors resulting from the interaction between modules are initially avoided.

For example, a hotel information system consists of modules to handle reservations; guest checking and checkout; restaurant, room service and miscellaneous charges; convention activities; and accounts receivable billing. For each, it provides the ability to enter, modify or retrieve data and respond to different types of inquiries or print reports. The test cases needed for unit testing should exercise each condition and option.



Unit testing can be performed from the bottom up, starting with smallest and lowest-level modules and proceeding one at a time. For each module in bottom-up testing a short program is used to execute the module and provides the needed data, so that the module is asked to perform the way it will when embedded within the larger system.

**System Testing*:*** The objective of system testing is to ensure that all individual programs are working as expected, that the programs link together to meet the requirements specified and to ensure that the computer system and the associated clerical and other procedures work together.

The initial phase of system testing is the responsibility of the analyst who determines what conditions are to be tested, generates test data, produced a schedule of expected results, runs the tests and compares the computer produced results with the expected results with the expected results.

The analyst may also be involved in procedures testing. When the analyst is satisfied that the system is working properly, he hands it over to the users for testing. The importance of system testing by the user must be stressed. Ultimately it is the user must verify the system and give the go-ahead.

During testing, the system is used experimentally to ensure that the software does not fail, i.e., that it will run according to its specifications and in the way users expect it to. Special test data is input for processing (test plan) and the results are examined to locate unexpected results.

A limited number of users may also be allowed to use the system so analysts can see whether they try to use it in unexpected ways. It is preferably to find these surprises before the organization implements the system and depends on it. In many organizations, testing is performed by persons other than those who write the original programs. Using persons who do not know how certain parts were designed or programmed ensures more complete and unbiased testing and more reliable software.

The system is tested as a complete, integrated system. System testing first occurs in the development environment but eventually is conducted in the production environment. Functionality and performance testing are designed to catch bugs in the system, unexpected results, or other ways in which the system does not meet the stated requirements.

The testers create detailed scenarios to test the strength and limits of the system, trying to break it if possible. Editorial reviews not only correct typographical and grammatical errors, but also



improve the system’s overall usability by ensuring that on-screen language is clear and helpful to users. Accessibility reviews ensure that the system is accessible to users with disabilities.

System testing consists of the following five steps:

1. Program testing
2. String testing
3. System testing
4. System documentation
5. User acceptance testing

##### Program Testing

A program represents the logical elements of a system. For a program to run satisfactorily, it must compile and test data correctly and tie in properly with other programs. It is the responsibility of a programmer to have an error free program. At

The time of testing the system, there exists two types of errors that should be checked. These errors are syntax and logic.

A syntax error is a program statement that violates one or more rules of the language in which it is written. An improperly defined field dimension or omitted key words are common syntax errors. These errors are shown through error messages generated by the computer. A logic error, on the other hand, deals with incorrect data fields out of range items, and invalid combinations.

Since the logical errors are not detected by compiler, the programmer must examine the output carefully to detect them. When a program is tested, the actual output is compared with the expected output. When there is a discrepancy, the sequence of the instructions, must be traced to determine the problem. The process is facilitated by breaking the program down into self- contained portions, each of which can be checked at certain key points.

##### String Testing

Programs are invariably related to one another and interact in a total system. Each program is tested to see whether it conforms to related programs in the system. Each part of the system is tested against the entire module with both test and live data before the whole system is ready to be tested.



##### System Testing

System testing is designed to uncover weaknesses that were not found in earlier tests. This includes forced system failure and validation of total system as it will be implemented by its user in the operational environment. Under this testing, generally we

Take low volumes of transactions based on live data. This volume is increased until the maximum level for each transaction type is reached.

The total system is also tested for recovery and fallback after various major failures to ensure that no data are lost during the emergency.

All this is done with the old system still in operation. When we see that the proposed system is successful in the test, the old system is discontinued.

##### System Documentation

All design and test documentation should be well prepared and kept in the library for future reference. The library is the central location for maintenance of the new system.

##### User Acceptance Testing

An acceptance test has the objective of selling the user on the validity and reliability of the system. It verifies that the system's procedures operate to system specifications and that the integrity of important data is maintained. Performance of an acceptance test is actually the user's show. User motivation is very important for the successful performance of the system. After that a comprehensive test report is prepared. This report shows the system's tolerance, performance range, error rate and accuracy.



**Table 6.1 Test Report with test data**

| **TEST REPORT WITH TEST DATA**  (To be filled by System Analyst/Programmer) | | |
| --- | --- | --- |
| **Project Name : Online Examination System** | | |
| **S No.** | **Testing Parameter** | **Observations** |
| A. | INTERFACE TESTING   1. User-friendliness 2. Consistent menus | OK NA |
| B. | CONTROL FLOW TESTING   1. IF-THEN-ELSE 2. DO WHILE 3. CASE-SWITCH | OK  OK OK |
| C. | VALIDATION TESTING   1. Check for improper or inconsistent typing 2. Check for erroneous initialization or default values 3. Check for incorrect variable names 4. Check for inconsistent Data Types 5. Check for relational/arithmetic operators | OK OK OK OK  OK |
| D. | DATA INTEGRITY/SECURITY TESTING   1. Data Insertion/ Deletion/ Updating 2. Boundary condition (Underflow, Overflow Exception) 3. Check for unauthorized access of data 4. Check for data availability | OK OK OK OK |



| E. | EFFICIENCY TESTING   1. Throughput of the system 2. Response time of the system 3. Online disk storage required by the system 4. Primary memory required by the system | OK OK OK  OK |
| --- | --- | --- |
| F. | ERROR HANDLING ROUTINES   1. Error description are intelligent/ understandable 2. Error recovery is smooth 3. All error handling routines are tested and executed at least once | OK OK OK |



# CHAPTER – VII

***CONCLUSION AND FUTURE ENHANCEMENTS***



### Limitations

The new system has been designed to meet almost all of the user requirements but it too has certain limitations some of which can be enhanced in the future enhancements or updates

* **Achieve good quality ratings.**
* **Better revenue management.**
* **Avoid errors and track every single detail.**
* **Improved clinical decision-making.**
* **Improve data security.**
* **Establish your hospital as technically**



### Future Enhancements

Enhancements are the perquisite for development of a system. Every existing system has proposed enhancements which make it better and easier to use and more secure. The enhancements that have been proposed for this system are listed here.

* Ability to accept the appointment by the doctor to acknowledge the patient that their appointment has been approved.
* User should not be allowed to register if he/she tries to provide the already registered email ID.
* The password should be encrypted and the password field shouldn't be displayed in the admin panel.
* Implementation of pagination for all the list view across the application.
* Bug fix - Bill payment receipt contains multiple record if the patient has associated with the same doctor multiple times.
* Addition of more fields in the prescription statement to make it more specific one.
* Addition of more details on payment - such as date of the payment made, amount paid, etc.
* Implementation of export button in admin module to export all details to an excel sheet.

### Conclusion

Since we are entering details of the patients electronically in the” Hospital Management System”, data will be secured. Using this application we can retrieve patient’s history with a single click. Thus processing information will be faster. It guarantees accurate maintenance of Patient details. It easily reduces the book keeping task and thus reduces the human effort and increases accuracy speed.





# CHAPTER – VIII BIBLIOGRAPHY



### Bibliography

* + - * List of useful Websites
* <https://getbootstrap.com/>
* <https://www.mysql.com/>
* <https://www.php.net/>
* <https://www.apachefriends.org/index.html>
* <https://www.academia.edu/>