**MINI PROJECT**

**On**

**HEALTH LINE**

**SubmittedBy**

**AASHISHPORWAL(160116737084)**

**K ADITYA(160116737085)**

**Under the guidance of**

**Mr.HANUMANTH RAO**

**Asst. Professor,**

**Dept. of IT, CBIT.**



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**CHAITANYA BHARATHI   INSTITUTE OF TECHNOLOGY**

**(Affiliated to Osmania University; Accredited by NBA (AICTE), ISO Certified 9001:2015)**

**Website:** [www.cbit.ac.in](http://www.cbit.ac.in)**,** www.cbitworld.com

**HYDERABAD – 500 075**

**2017-18**

**i**

**CHAITANYA BHARATHI   INSTITUTE OF TECHNOLOGY**

DEPARTMENT OF INFORMATION TECHNOLOGY

**(Affiliated to Osmania University)**

**GANDIPET, HYD111ERABAD – 500 075**



**CERTIFICATE**

This is to certify that the project work titled “**HEALTH LINE**” submitted to **CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY,** in partial fulfilment of the requirements for the award of the completion of 4th semester of B.E. in Information Technology, during the academic year 2017-18, is a record of original work done by**AASHISH PORWAL-160116737084**and**K.ADITYA-160116737085**during the period of study in Dept. of IT, CBIT, HYDERABAD, under our supervision and guidance.

**PROJECT GUIDE       HEAD OF THE DEPARTMENT**

**Mr.Hanumanth Rao        Dr. Suresh Pabboju**

Asst. Professor,       Head of the department

Dept. of IT, CBIT.       Information Technology

       C.B.I.T., HYDERABAD

**ii**

**DECLARATION**

This is to certify that the work reported in the present report titled “**HEALTH LINE**” is a record of work done by us in the Department of Information Technology, Chaitanya Bharathi Institute of Technology, Hyderabad.

No part of the report is copied from books / journals / internet and wherever the portion is taken, the same has been duly referred. The reported results are based on the project work done entirely by us and not copied from any other source.

AASHISHPORWAL(160116737084)

K ADITYA(160116737085)

**ACKNOWLEDGEMENT**

It is our privilege to acknowledge with deep sense of gratitude and devotion for keen personal interest and invaluable guidance rendered by our Project Guide   **Ms. Hanumanth Rao** Assistant Professor, Department of Information, Chaitanya Bharathi Institute of Technology.

We take the opportunity to express our thanks to **Dr.Suresh Pabboju ,** Professor  & Head of IT Department, CBIT for his valuable suggestions and moral support.

We are grateful to our Principal **Dr.P Ravinder Reddy,** Chaitanya Bharati Institute of Technology, for his cooperation and encouragement.

Finally, we also thank all the staff members, faculty of Dept. of IT, CBIT, and our friends, who with their valuable suggestions and support, directly or indirectly helped us in completing this project work.

**iv**

**ABSTRACT**

The main objective of our project is to connect the public with the hospitals and the famous Doctors .From this, it is easy to check the famous hospitals in and around our locality .It is also useful to get the suggestions about the disease which an individual is suffering. From our project we can save the time that have been wasted for searching the correct and famous hospitals for the disease which we are suffering from. It is easy to know the famous and super speciality hospitals in a city or in a state . Our aim is to make out the health services in a smarter way.

**v**

**CONTENTS**

**PageNo.**

**1. Introduction. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .8**

**1.1. Objective . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .8**

**1.2. Motivation. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. . .8**

**1.3. Overview . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .8**

**2. Technologies . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .10**

**2.1. About HTML . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. .. …. . 10**

**2.1.1. Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10**

**2.1.2. HTML Basic tags . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. 11**

**2.1.3. HTML ATTRIBUTES . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . ..11**

**2.1.4. HTML Images . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12**

**2.1.5. HTML Tables . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . ..13**

**2.1.6. HTML Forms . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . … .13**

**2.1.7 php . . . . . . . . . . . .. . . .. . . . . .. . . . . . . . . . . . . . . . . . . . . . . . . .. . . 14**

**2.1.8 MYSQL .. ……. . . . . . . . . . . . . . .. . . . . . . . .. . . . . . . . . . . . . . . . . . . .16**

**3. Software Requirement Specification . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. .20**

**3.1. Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20**

**3.1.1 Purpose of the document . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20**

**3.2. Users and their Characteristics . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .20**

**vi**

**3.3. Software and Hardware Requirements. . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. . .21**

**4. Implementation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .21**

**4.1. Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .10**

**4.2. HTMLphpmysql. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .10**

**5. Testing and Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20**

**5.1. Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22**

**5.2. Testing Objectives . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .22**

**5.3. Output Screens . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .23**

**6. Conclusion and Future Scope . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .25**

**7.Bibliography . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .26**

**vii**

**1. INTRODUCTION**

**1.1Objective**

This system helps decision making and provide 24x7 anytime, anywhere access to information are becoming a vital part of today’s healthcare.It provide details of Hospitals , doctors , contact and suggestion.

**1.2 Motivation**

The main motivation and reason for choosing this project is to make public feel easy to find the hospitals in the city and get the proper medication . The intentions of the system are to reduce over time pay and increases the number of patients accurately and easily.

**1.3 OVERVIEW**

In our project we will provide famous doctor and hospitals around and , inventory. We will provide existing matter and also new entry of hospitals .our project consists of 4 screens after registration and login into account we have to choose the disease name and finally it will display the hospitals list doctor name and contact name.

**1.4 Problem Statement**

It is very important to maintain efficient software to handle information of a Hospital. This application provides away to record this information and to access these in a simple way using this web application one can easily find better hospitals around easily and get proper Medication .By this we can easily contact with the hospitals and get the appointments through phone.

**1.1. ORGANISATION OF REPORT**

* Chapter 1 deals with the Introduction of the project and gives the details about the project in an abstract view.
* Chapter 2 deals with the information about HTML,phpmysql utilization details are discussed in brief.
* Chapter 3 deals with the Software Requirements Specifications which is a specification of the project software and hardware requirements.
* Chapter 4 deals with the Implementation part which includes the tools and softwares that are used.
* Chapter 5 deals with the Testing of the project and screenshots of the project
* Chapter 6 explains the Conclusion and further scope of the project.

2

**2. TECHNOLOGIES**

**2.1 ABOUT HTML**

**2.1.1 INTRODUCTION**

HTML stands for Hyper Text Markup Language, which is the most widely used language on Web to develop web pages.

Hypertext refers to the way in which Web pages (HTML documents) are linked together. Thus the link available on a webpage are called Hypertext.

As its name suggests, HTML is a Markup Language which means you use HTML to simply "mark up" a text document with tags that tell a Web browser how to structure it to display.

Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers.

**2.1.2. HTML Basic Tags**

1.Any document starts with a heading. You can use different sizes for your headings. HTML also has six levels of headings, which use the elements <h1>, <h2>, <h3>, <h4>, <h5>, and <h6>. While displaying any heading, browser adds one line before and one line after that heading.

2.The <p> tag offers a way to structure your text into different paragraphs. Each paragraph of text should go in between an opening <p> and a closing </p>tag .

3.Whenever you use the <br /> element, anything following it starts from the next line. This tag is an example of an empty element, where you do not need opening and closing tags, as there is nothing to go in between them.

4.You can use <center> tag to put any content in the center of the page or any table cell.

**2.1.3 HTML ATTRIBUTES**

An attribute is used to define the characteristics of an HTML element and is placed inside the element's opening tag. All attributes are made up of two parts: a name and a value:

-> The name is the property you want to set. For example, the paragraph <p> element in the example carries an attribute whose name is align, which you can use to indicate the alignment of paragraph on the page.

-> The value is what you want the value of the property to be set and always put within quotations. The below example shows three possible values of align attribute: left, center and right.

Attribute names and attribute values are case-insensitive. However, the World Wide Web Consortium (W3C) recommends lowercase attributes/attribute values in their HTML 4 recommendation.

**CORE ATTRIBUTES**

**1. Id Attribute**

The id attribute of an HTML tag can be used to uniquely identify any element within an HTML page. There are two primary reasons that you might want to use an id attribute on an

element:

-> If an element carries an id attribute as a unique identifier it is possible to identify just that element and its content.

-> If you have two elements of the same name within a Web page (or style sheet), you can use the id attribute to distinguish between elements that have the same name.

**2. Title Attribute**

The title attribute gives a suggested title for the element.Thebehavior of this attribute will depend upon the element that carries it, although it is often displayed as a tooltip when cursor comes over the element or while the element is loading.

**2.1.4 HTML IMAGES**

Images are very important to beautify as well as to depict many complex concepts in simple way on your web page.

**INSERT IMAGES**

You can insert any image in your web page by using <img> tag. Following is the simple syntax to use this tag.

<imgsrc="Image URL" ... attributes-list/>

**SET IMAGE WIDTH /HEIGHT**

You can set image width and height based on your requirement using width and height attributes. You can specify width and height of the image in terms of either pixels or

percentage of its actual size.

**2.1.5 HTML TABLES**

The HTML tables allow web authors to arrange data like text, images, links, other tables, etc. into rows and columns of cells.

The HTML tables are created using the <table> tag in which the <tr> tag is used to create table rows and <td> tag is used to create data cells.

**TABLE HEADING**

Table heading can be defined using <th> tag. This tag will be put to replace <td> tag, which is used to represent actual data cell. Normally you will put your top row as table heading , otherwise you can use <th> element in any row.

**TABLE CAPTION**

The caption tag will serve as a title or explanation for the table and it shows up at the top of the table.

**2.1.6. HTML FORMS**

The <form> Element:The HTML <form> element defines a form that is used to collect user input.

**1.The<input> Element**

The <input> element is the most important form element and can be displayed in several ways, depending on the type attribute.

Type Description

<input type="text"> Defines a one-line text input field

<input type="radio"> Defines a radio button (for selecting one of many choices)

<input type="submit"> Defines a submit button (for submitting the form)

**2.The Action Attribute**

The action attribute defines the action to be performed when the form is submitted.Normally, the form data is sent to a web page on the server when the user clicks on the submit button. The form data is sent to a page on the server called "/action\_page.php". This page contains a server-side script that handles the form data:

<form action="/action\_page.php">

**3.The Method Attribute**

The method attribute specifies the HTTP method (GET or POST) to be used when submitting the form data:

<form action="/action\_page.php" method="POST">

PHP Syntax:

A PHP script is executed on the server, and the plain HTML result is sent back to the browser. A PHP script starts with <?php and ends with ?>.

The default file extension for PHP files is ".php".A PHP file normally contains HTML tags, and some PHP scripting code.

PHP Variables:

In PHP, a variable starts with the $ sign, followed by the name of the variable:

Example :

<?php

$S=”hello world”;

$x=5;

$y=10.5;

?>

After the execution of the statements above, the variable $txt will hold the value Hello world!, the variable $x will hold the value 5, and the variable $y will hold the value 10.5.

PHP ECHO/PRINT:

Echo and print are more or less the same. They are both used to output data to the screen. The differences are small: echo has no return value while print has a return value of 1 so it can be used in expressions. echo can take multiple parameters (although such usage is rare) while print can take one argument.

Echo is marginally faster than print. The echo statement can be used with or without parentheses: echo or echo(). The print statement can be used with or without parentheses: print or print().

PHP Data Types:

Variables can store data of different types, and different data types can do different things.PHP supports the following data types:

• String

• Integer

• Float (floating point numbers - also called double)

• Boolean

• Array

• Object

• NULL

**•** Resource

**MYSQL**

## MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons −

## MySQL is released under an open-source license. So you have nothing to pay to use it.

## MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.

## MySQL uses a standard form of the well-known SQL data language.

## MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.

## MySQL works very quickly and works well even with large data sets.

## MySQL is very friendly to PHP, the most appreciated language for web development.

## MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).

## MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

## **DATA TYPES IN MYSQL:**

MySQL uses many different data types broken into three categories −

* Numeric
* Date and Time
* String Types.

Let us now discuss them in detail.

## Numeric Data Types

MySQL uses all the standard ANSI SQL numeric data types, so if you're coming to MySQL from a different database system, these definitions will look familiar to you. The following list shows the common numeric data types and their descriptions −

* INT − A normal-sized integer that can be signed or unsigned. If signed, the allowable range is from -2147483648 to 2147483647. If unsigned, the allowable range is from 0 to 4294967295. You can specify a width of up to 11 digits.
* TINYINT − A very small integer that can be signed or unsigned. If signed, the allowable range is from -128 to 127. If unsigned, the allowable range is from 0 to 255. You can specify a width of up to 4 digits.
* SMALLINT − A small integer that can be signed or unsigned. If signed, the allowable range is from -32768 to 32767. If unsigned, the allowable range is from 0 to 65535. You can specify a width of up to 5 digits.
* MEDIUMINT − A medium-sized integer that can be signed or unsigned. If signed, the allowable range is from -8388608 to 8388607. If unsigned, the allowable range is from 0 to 16777215. You can specify a width of up to 9 digits.
* BIGINT − A large integer that can be signed or unsigned. If signed, the allowable range is from -9223372036854775808 to 9223372036854775807. If unsigned, the allowable range is from 0 to 18446744073709551615. You can specify a width of up to 20 digits.
* FLOAT(M,D) − A floating-point number that cannot be unsigned. You can define the display length (M) and the number of decimals (D).

* This is not required and will default to 10,2, where 2 is the number of decimals and 10 is the total number of digits (including decimals). Decimal precision can go to 24 places for a FLOAT.
* DOUBLE(M,D) − A double precision floating-point number that cannot be unsigned. You can define the display length (M) and the number of decimals (D). This is not required and will default to 16,4, where 4 is the number of decimals. Decimal precision can go to 53 places for a DOUBLE. REAL is a synonym for DOUBLE.
* DECIMAL(M,D) − An unpacked floating-point number that cannot be unsigned. In the unpacked decimals, each decimal corresponds to one byte. Defining the display length (M) and the number of decimals (D) is required. NUMERIC is a synonym for DECIMAL.

## Date and Time Types

The MySQL date and time datatypes are as follows −

* DATE − A date in YYYY-MM-DD format, between 1000-01-01 and 9999-12-31. For example, December 30th, 1973 would be stored as 1973-12-30.
* DATETIME − A date and time combination in YYYY-MM-DD HH:MM:SS format, between 10
* 00-01-01 00:00:00 and 9999-12-31 23:59:59. For example, 3:30 in the afternoon on December 30th, 1973 would be stored as 1973-12-30 15:30:00.
* TIMESTAMP − A timestamp between midnight, January 1st, 1970 and sometime in 2037. This looks like the previous DATETIME format, only without the hyphens between numbers; 3:30 in the afternoon on December 30th, 1973 would be stored as 19731230153000 ( YYYYMMDDHHMMSS )
* YEAR(M) − Stores a year in a 2-digit or a 4-digit format. If the length is specified as 2 (for example YEAR(2)), YEAR can be

between 1970 to 2069 (70 to 69). If the length is specified as 4, then YEAR can be 1901 to 2155. The default length is 4.

## String Types

Although the numeric and date types are fun, most data you'll store will be in a string format. This list describes the common string datatypes in MySQL.

* CHAR(M) − A fixed-length string between 1 and 255 characters in length (for example CHAR(5)), right-padded with spaces to the specified length when stored. Defining a length is not required, but the default is 1.
* VARCHAR(M) − A variable-length string between 1 and 255 characters in length. For example, VARCHAR(25). You must define a length when creating a VARCHAR field.
* BLOB or TEXT − A field with a maximum length of 65535 characters. BLOBs are "Binary Large Objects" and are used to store large amounts of binary data, such as images or other types of files. Fields defined as TEXT also hold large amounts of data. The difference between the two is that the sorts and comparisons on the stored data are case sensitive on BLOBs and are not case sensitive in TEXT fields. You do not specify a length with BLOB or TEXT.
* TINYBLOB or TINYTEXT − A BLOB or TEXT column with a maximum length of 255 characters. You do not specify a length with TINYBLOB or TINYTEXT
* MEDIUMBLOB or MEDIUMTEXT − A BLOB or TEXT column with a maximum length of 16777215 characters. You do not specify a length with MEDIUMBLOB or MEDIUMTEXT.
* LONGBLOB or LONGTEXT − A BLOB or TEXT column with a maximum length of 4294967295 characters. You do not specify a length with LONGBLOB or LONGTEXT.
* ENUM − An enumeration, which is a fancy term for list. When defining an ENUM, you are creating a list of items from which the value must be selected (or it can be NULL). For example, if you wanted your field to contain "A" or "B" or "C", you would define your ENUM as ENUM ('A', 'B', 'C') and only those values (or NULL) could ever populate that field.

**3. SOFTWARE REQUIREMENT SPECIFICATION**

**3.1 INTRODUCTION**

The requirements specification is a technical specification of requirements for the software products. It is the first step in the requirements analysis process it lists the requirements of a particular software system including functional, performance and security requirements. The requirements also provide usage scenarios from a user, an operational and an administrative perspective. The purpose of software requirements specification is to provide a detailed overview of the software project, its parameters and goals. This describes the project target audience and its user interface, hardware and software requirements. It defines how the client, team and audience see the project and its functionality.

**3.1.1 Purpose of the document**

This software requirement specification describes all the requirements elicited for ìHealth lineî and is intended to be used by the members examining the project and implementing and verifying the application. Unless otherwise noted all requirements are of high priority and are committed.

**3.2. USERS AND THEIR CHARACTERISTICS**

Health line website is useful to users who are looking for proper medication in an interactive way

**3.3 SOFTWARE AND HARDWARE REQUIREMENTS**

Operating System Windows XP or higher version

Programming Languages HTML,CSS, Mysql

Processor Intel(R) Core(TM) i3 CPU M 350 @2.27GHz

RAM 1 GB or more

Disk Space 1GB or more

**4. IMPLEMENTATION**

**4.1 INTRODUCTION**

The success of the software product is determined only when it is successfully implemented according to the requirements. The analysis and the design of the proposed system provide a perfect platform to implement the idea using the specified technology in the desired environment. The implementation of our system is made user friendly.

Any software project is designed in modules and the project is said to be successfully implemented when each of the module is executed individually to obtain the expected result and also, when all the modules are integrated and run together without any errors.

**4.2 HTML**

HTML was used to build the basic structure and display the contents of the WebPages in the website. The basic structure of home page of the website and the articles were built using HTML. They have slight differences in their structures. It includes the CSS and Java Script files required for the website.

**TABLE CREATION IN MYSQL DATABASE:**

To begin with, the table creation command requires the following details −

* Name of the table
* Name of the fields
* Definitions for each field

Syntax

Here is a generic SQL syntax to create a MySQL table −

CREATE TABLE table\_name (column\_name column\_type);

Here, a few items need explanation –

20

* Field Attribute NOT NULL is being used because we do not want this field to be NULL. So, if a user will try to create a record with a NULL value, then MySQL will raise an error.
* Field Attribute AUTO\_INCREMENT tells MySQL to go ahead and add the next available number to the id field.
* Keyword PRIMARY KEY is used to define a column as a primary key. You can use multiple columns separated by a comma to define a primary key.

**5. TESTING AND RESULTS**

**5.1 INTRODUCTION**

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. In fact, testing is the one step in the software engineering process that could be viewed as destructive rather than constructive.

A strategy for software testing integrates software test case design methods into a well-planned series of steps that result in the successful construction of software. Testing is the set of activities that can be planned in advance and conducted systematically. The underlying motivation of program testing is to affirm software quality with methods that can economically and effectively apply to both strategic to both large and small-scale systems.

**5.2 TESTING OBJECTIVES**

The main objective of performance testing is designed to test whether the websiteís display is as expected and whether the website is functioning properly or not.

As the test results are gathered and evaluated they begin to give a qualitative indication of the reliability of the website. If proper output is not obtained, the overall quality of the Website is questioned. If, on the other hand, all the results which are not successful, are encountered, and are easily modifiable, then the following conclusion can be made: The tests are inadequate as the requirements mentioned are not compatible. The testing includes:

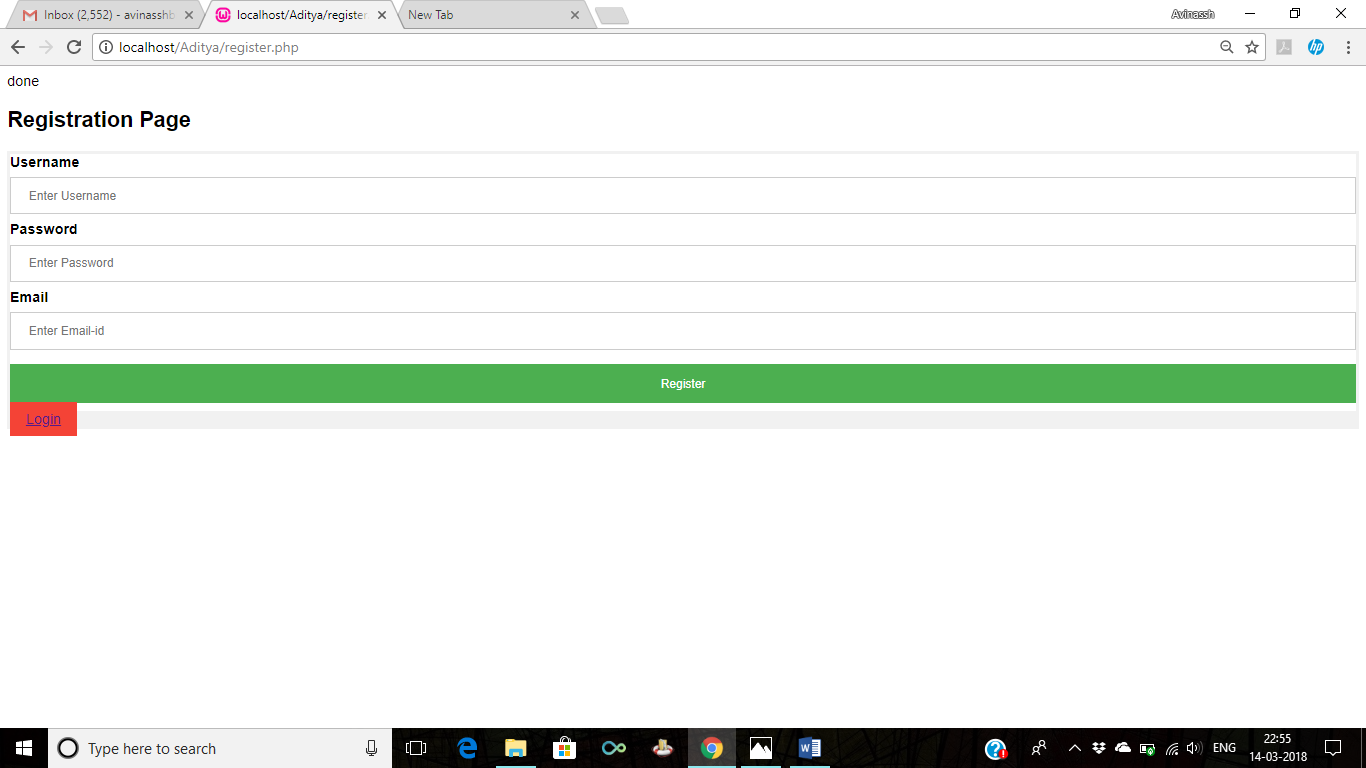
-> Checking whether the information is displayed or not.

-> Checking whether all the links between each webpage in the website works or is misdirected.

-> Verifying if all the pictures are displayed and none of the files are corrupted.

11

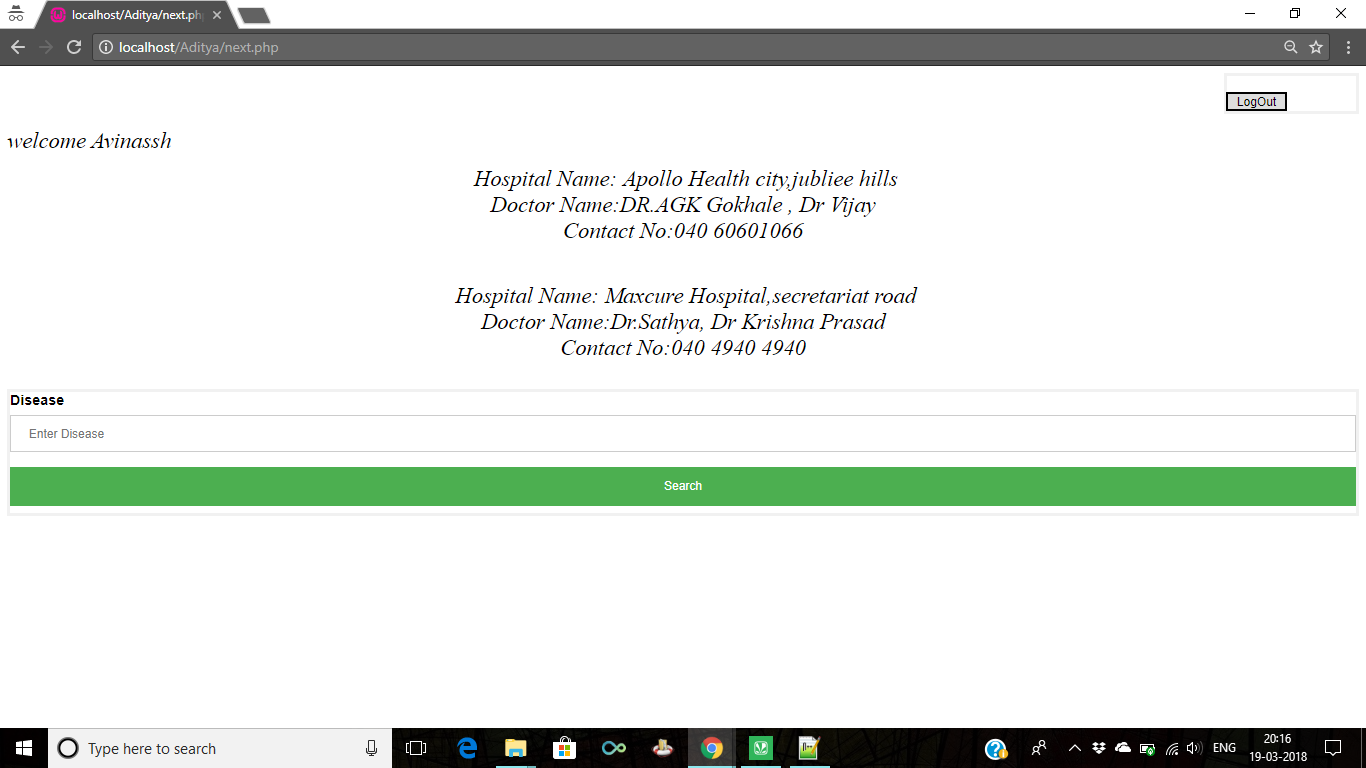
**5.3. OUTPUT SCREENS:**



**Fig 5.1.HOME Page**

The above figure shows the home page of the website containing login and registration and the overview of the page.

12



**Fig 5.2.**

Display of the hospitals . Admin page to add the updated content

13

**6. CONCLUSION AND FUTURE SCOPE**

Our website “**Health line**” is designed in such a way that future modifications can be done easily. The following conclusion can be deduced from the development of our project.

-> Dynamic updation of the hospitals from other servers

-> Developing it has app and connecting it to the travelling services to go to the hospitals directly

-> Location of the hospitals review of the hospitals and the rating of the hospitals can be included in our project.

Using our website the users can easily find the Hospitals.This is a place for public using this we can find famous hospitals for the specified disease which we are suffering from.

**7. BIBLIOGRAPHY**

**Websites:**

[1] https://www.w3schools.com/html/default.asp

[2] https://www.w3schools.com/mysql/default.asp

[3] https://www.w3schools.com/php/default.asp

15