**CHAPTER 1**

**INTRODUCTION**

Today many clinics need software, which enable them to manage all aspects of the clinics and health-cares centers on the offline application as well as web application. Clinic Management System is a web based application to handle the aspects and services of clinics like manage records of patients, staffs, Prescription printing service, Review-date, expense report, Patient-report and Reminder for review case.

Clinic Management System is a admin based software which developed under the platform of PHP (Codeigniter) and architecture is based on MVC i.e; Model View Controller. It is only for the business enhancement and provide easiness while fulfill the required needs for records maintenance.

**1.1 Background of the problem**

The main objective of this project is to automate process of clinic and health-cares which covers all the Maintenance processes like:

1. Customer Registration
2. Print Prescription with Unique Id.
3. Patient Review (If need).
4. Reminder for the Review date of Patient.

**1.2 Problem Statement**

Clinic management system is a web application for admin that allow the admin of this software to maintain their regular aspects of the clinics like new patient visit to clinic they have to register first, Old patient need of review their case so they only got updated using the already assigned unique id.

To obtain the details from the software admin have to first generate the id of customer after that admin will able to update ,search or modify their new or existing data.

To use the further specialty of the software i.e. prescription printing with the patient interrelated details like id, name, date and reports admin have to first register the new patient..Details of staff are also available, admin can first have to register the staff with their required details after admin can access to their data whenever he/she need,

To access CMS, the admin would go to login page, and enter the username and password. Some financial institutions have set up additional security steps for access, but there is no consistency to the approach adopted.

**1.3 Purpose of the project**

The main purpose of this project is to handle the records and manage the records without the use of any pen paper. All the records are manage, update as well as register with the help of this software. It reduces the overhead of admin to maintain the record without having any manual action.

**1.4 Scope of the project**

Scope of this project is very wide. It aims for providing services to admin of the software to enhance their business by the use of digital system rather than the use of maintaining records manually.

**CHAPTER 2**

**REQUIREMENT ANALYSIS**

**2.1 Feasibility Study**

Feasibility study is the determination whether or not a project is worth doing. The process followed in making this is called feasibility study. When the developer is building any system, he/she should make sure that the system he/she is going to build is feasible. Since the feasibility study may lead to commitment of large resources, it becomes necessary that it should be conducted competently and that no fundamental errors of judgment are made.

Three important test of feasibility are described below:

When complex problems and opportunities are to be defined, it is generally desirable to conduct a preliminary investigation called feasibility study. Feasibility study involves investigating the information needs of the prospective end users and the objectives, constraints, basic resource requirements, costs, benefits and feasibility of a proposed project, by intent, the feasibility study is very rough analysis of the viability of the project. It is, however, a highly desirable checkpoint that should be completed before committing more resources.

The outcome of the feasibility study is very clear. So depending upon the aspect on which feasibility is being done is categorized into following classes:-

* Technical feasibility
* Economical feasibility
* Operational feasibility
* Time feasibility

**2.1.1 Technical Feasibility Study**

This involves questions such as whether the technology needed for the system exists, how difficult it will be to build, and whether the firm has enough experience using that technology. The assessment is based on an outline design of system requirements in terms of Input, Output, Fields, Programs, and Procedures. This can be qualified in terms of volumes of data, trends, frequency of updating, etc.

Technical Feasibility is concerned with the availability and capability of hardware, software and people. In Technical Feasibility Study we studied on those facts that are covering the whole technological facts such as:

* Hardware requirement specification
* Software requirement specification
* Technology used
* Whether the required technology is available or nor?
* Whether the required resources are available-

**2.1.2 Economical Feasibility Study**

It is the most frequently used method for evaluating the effectiveness of the system. The procedure is to determine the benefits and savings that are expected from the system in comparison with the cost involved. If the benefits outweigh the cost, then system is said to be economically feasible. Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system.

**2.1.3 Operational Feasibility Study**

It is mainly related to human organizational and political aspects. For operational feasibility study we appointed a small group of people, which included the office employees. This group worked with the new system and found that the system was very user friendly, and very efficient. It was found that no special training was required to operate the system. The burden of the project manager who wants to give the issue to other employees and other staff members was reduced. It takes about 5 minutes to enter the issue details as compared to the 30-40 minute of the manual system. And the employees found that they can prepare the statistical reports with more speed, accuracy and in very less time.

**2.1.4 Time Feasibility Study**

#### Time feasibility is a determination of whether a proposed project can be implemented fully within a stipulated time frame. If a project takes too much time it is likely to be rejected.

**2.2 Software Specifications**

* G U I : HTML, BOOTSTRAP.
* Language : Advance PHP (Codeigniter)
* D B M S : MYSQL
* Operating System : Microsoft Windows 7
* Application : CLINIC/HEALTHCARE
* Documentation : Microsoft Word 2007
* Image Processor : Microsoft Paint
* Software : Adobe DREAMWEAVER CC

* 1. **Hardware Specifications**
* Processor : CORE I3
* Memory : 4GB
* Hard disk : 160 GB
* Keyboard : Multimedia(104 keys)
* Mouse : 3 Button scroll
* Monitor : 17”Color Monitor
* Screen Resolution : 1240 X 860 Pixels

**CHAPTER 3**

**REQUIREMENT PROCESS**

The goal of requirement process is to create and maintain a system requirement document. The overall process includes the high level requirement engineering sub process. They are some sub process which is given below:

**3.1** **Requirement Specification**

* Interview
* Studying the existing system
* Brain storming

**3.2 Requirement Analysis**

* Anomaly
* Inconsistency
* Incompleteness

**3.3 Requirement Elicitation**

Requirement specification is a final work product produce by software engineer.

The SRS document should clear the following aspect of system.

* Functional requirement
* Non Functional requirement

**3.4 Requirement Validation**

* Omission
* Incorrect fact
* Ambiguity

**CHAPTER 4**

**SYSTEM DESIGN**

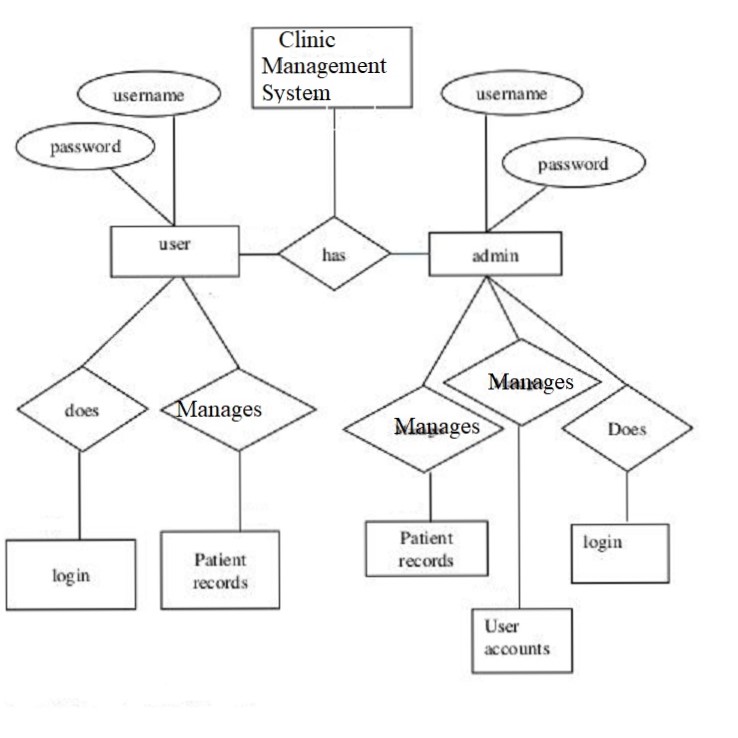
System design take place when specification phase is completed and approved by customer .

**4.1 ER Diagram**

ER diagram stands for entity-relationship diagram,It is a graphical representation of entities and their relationships to each other.

There are three basic elements in ER models:

1. Entities are the "things" about which we seek information.
2. Attributes are the data we collect about the entities.
3. Relationships provide the structure needed to draw information from multiple entities.



**4.2 DFD**

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an [information system](http://en.wikipedia.org/wiki/Information_system). A DFD shows what kinds of data will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel (which is shown on a [flowchart](http://en.wikipedia.org/wiki/Flowchart)). A Data Flow Diagram (DFD) is also known as a Process Model. Process Modeling is an analysis technique used to capture the flow of inputs through a system (or group of processes) to their resulting output. On the context diagram (also known as the 'Level 0 DFD') the system's interactions with the outside world are modeled purely in terms of data flows across the system boundary.

This context-level DFD is next "exploded", to produce a Level 1 DFD that shows some of the detail of the system being modeled. The Level 1 DFD shows how the system is divided into sub-systems (processes), each of which deals with one or more of the data flows to or from an external agent, and which together provide all of the functionality of the system as a whole.

**4.2.1 0-Level DFD**

Records

Patient/Staff

**4.2.2 1-Level DFD**

New Staff Registration

New Patient

Registration

Patient Report

Staff Report

Patient Management

Expiry Alert

Staff Management

Print Prifcription

**CHAPTER 5**

**DATABASE DESIGN**

**5.1 Table name: login**

TABLE NAME: users

TABLE DESCRIPTION: It keeps the customers id and password that helps the bank to authenticate the customers.

|  |  |  |
| --- | --- | --- |
| **users table** | **Type** | **Constraint** |
| Id | Int | Auto generated |
| username | Varchar(30) | Primary key |
| Password | Varchar(30) | Not null |
| Active | Int(11) | Not null |

**5.2 Table name: accounts**

TABLE NAME: New

TABLE DESCRIPTION: It keeps the information of all the registered patients.

|  |  |  |
| --- | --- | --- |
| **New Table** | **Type** | **Constraint** |
| Id | Integer | Auto generated |
| First | Varchar(30) | Not null |
| Last | Varchar(30) | Not null |
| Date | Varchar(40) | Not null |
| Contact | Varchar(10) | Not null |
| Dob | Varchar(10) | Not null |
| Age | Varchar(20) | Not null |
| Address | Varchar(100) | Not null |
| City | Varchar(20) | Not null |
| State | Varchar(20) | Not null |
| Aadhar | Varchar(30) | Not null |
| Email | Varchar(30) | Not null |
| Purpose | Varchar(20) | Not null |

**5.3 Table name:**

TABLE NAME: review

TABLE DESCRIPTION: It keeps the information of all the Patient who have to re-visit clinic for review their case.

|  |  |  |
| --- | --- | --- |
| **Review Table** | **Type** | **Constraint** |
| Id | int(11) | Auto generated |
| user\_id | Varchar(30) | NOT NULL |
| p\_id | Varchar(30) | NOT NULL |
| Last | Varchar(20) | NOT NULL |
| Date | Varchar(20) | NOT NULL |
| Contact | varchar(10) | NOT NULL |
| Email | varchar(30) | NOT NULL |
| Purpose | varchar(20) | NOT NULL |
| Address | varchar(200) | NOT NULL |
| rewdate | varchar(10) | NOT NULL |
| status | int(1) | NOT NULL |

**5.4 Table name:**

TABLE NAME: Staff

TABLE DESCRIPTION: It keeps the information of all the Working Staff of the clinic.

|  |  |  |
| --- | --- | --- |
| **Staff Table** | **Type** | **Constraint** |
| Id | int(11) | Auto generated |
| user\_id | int(11) | NOT NULL |
| Name | Varchar(20) | NOT NULL |
| Date | Varchar(10) | NOT NULL |
| Address | Varchar(30) | NOT NULL |
| Contact | Varchar(10) | NOT NULL |
| Aadhar | Varchar(15) | NOT NULL |
| Age | Varchar(2) | NOT NULL |
| Salary | Varchar(10) | NOT NULL |
| Profile | Varchar(10) | NOT NULL |

**CHAPTER 6**

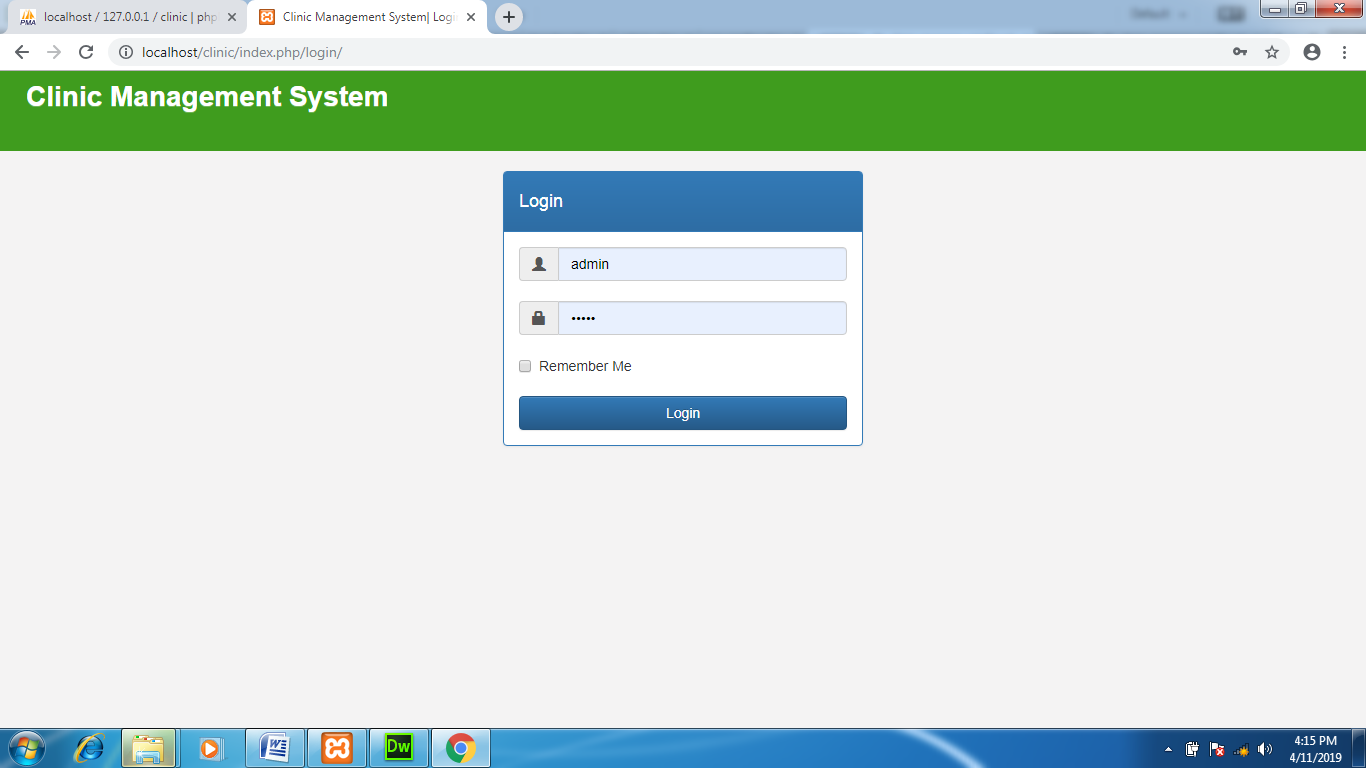
**CODING AND WEB PAGES**

**6.1 Login page: index.html**

It is the main starting page of this project. Note that Clinic Management System is a admin based software so, only admin have the right to access the software.

In this page there is two fields:

* Username(Admin)
* Password(Admin)

****

**CODING:**

**6.1.1 View of login page: login.php**

<div class="container">

<div class="row">

<div class="col-md-4">

</div>

<div class="col-md-4" id="login">

<div class="panel panel-primary">

<div class="panel-heading"><h4> Login</h4> </div>

<div class="panel-body">

<form method="post" class="bs-example bs-example-form" action="<?php echo base\_url(); ?>index.php/login/validate\_login/">

<div class="input-group">

<span class="input-group-addon"><i class="glyphicon glyphicon-user"></i></span>

<input type="text" name="username" placeholder="username" class="form-control">

</div><!-- input group closed-->

<br>

<div class="input-group">

<span class="input-group-addon"><i class="glyphicon glyphicon-lock"></i></span>

<input type="password" name="password" placeholder="password" class="form-control">

</div> <br>

<div class="input-group">

<label class="checkbox-inline"><input type="checkbox" name="remember" value="1">Remember Me</label>

</div> <br>

<div class="input-group center-block">

<input type="submit" name="login" class="btn btn-primary center-block" value="Login" style="width:100%">

</div>

<div id="logstatus" style="visibility:none" class="text-danger">

</div>

</form>

</div>

</div>

</div>

<div class="col md-4"></div>

</div><!-- row closed -->

</div><!-- container closed -->

**6.1.2 Controller of Login Page:Login.php**

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class Login extends CI\_Controller {

public function index()

{

if($this->session->userdata('user\_id')!==NULL){

redirect("/home/");

}

$data=array();

$data['title']="Login";

$this->load->view('header',$data);

$this->load->view('nav');

$this->load->view('login');

$this->load->view('footer');

}

public function validate\_login(){

$autoload['model'] = array('Login\_model'=>'login');

if($this->input->post('login')!==NULL){

$username=$this->input->post('username');

$password=$this->input->post('password');

$this->load->model('Login\_model');

$data['username']=$username;

$data['password']=$password;

$array=$this->Login\_model->login($data);

if(is\_array($array)){

$user\_id=$array['id'];

//$\_SESSION['user\_id']=$user\_id;

$this->session->set\_userdata("user\_id",$user\_id);

redirect("/home/");

}

else{

echo $array;

}

}

else{

redirect("/login/");

}

}

public function logout(){

$this->session->unset\_userdata("user\_id");

redirect("/login/");

}

}

**6.1.3 Model of Login Page: Login\_model.php**

**CODING**

<?php

class Login\_model extends CI\_Model{

public function login($data){

$run=$this->db->get\_where("users",$data);

if($run->num\_rows()==1){

$result=$run->row\_array();

return $result;

}

else{

return "Username or Password is wrong!";

}

}

}

?>

<?php

class Login\_model extends CI\_Model{

public function login($data){

$run=$this->db->get\_where("users",$data);

if($run->num\_rows()==1){

$result=$run->row\_array();

return $result;

}

else{

return "Username or Password is wrong!";

}

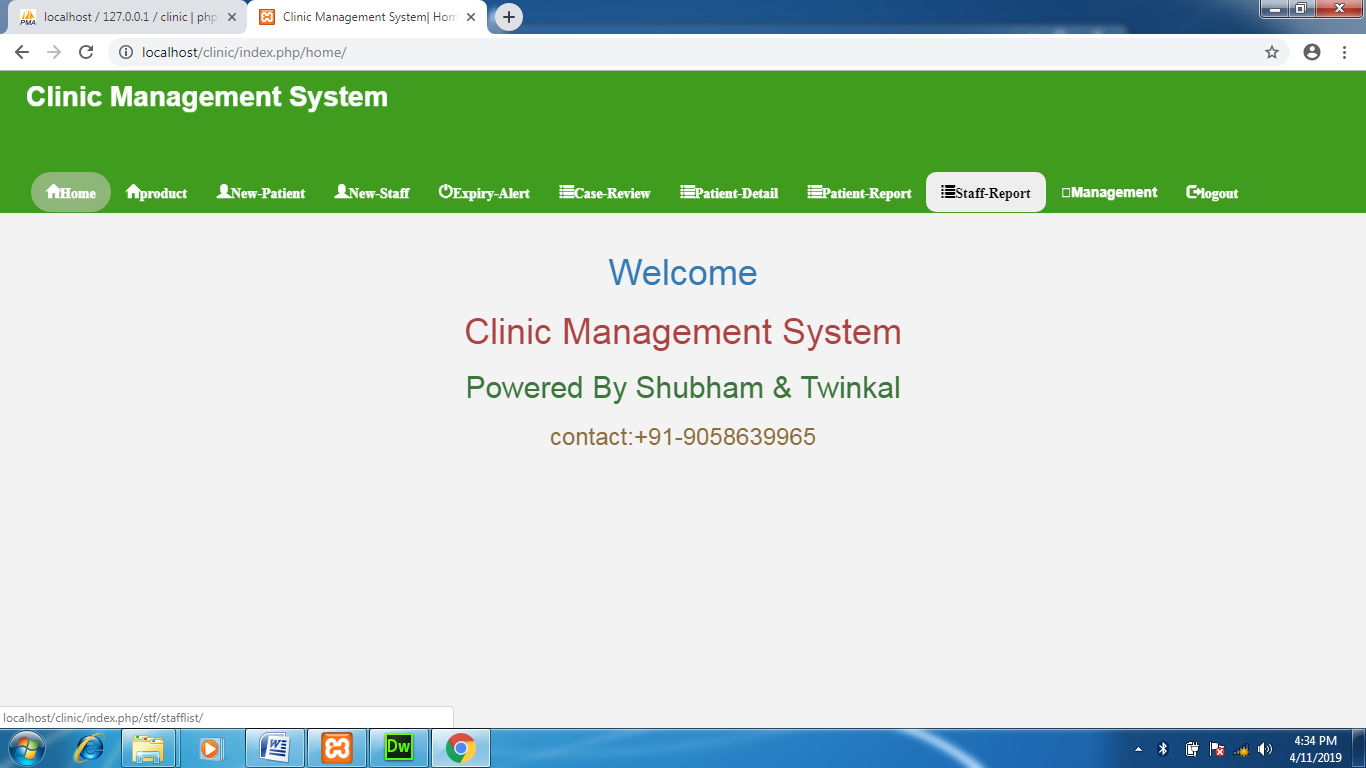
}

}

?>

**6.2. Frontend Of Clinic Management System**

When you Succesfully logged in login button will bring you to index.php.



**CODING**

**6.2.1: View Of index.php:**

<section>

<div class="container">

<div class="row">

<div class="col-md-12 ">

<h1 align="center" class="text-primary">Welcome </h1>

<h1 align="center" class="text-danger"> Clinic Management System</h1>

<h2 align="center" class=" text-success">Powered By Shubham & Twinkal</h2>

<h3 align="center" class="text-warning">contact:+91-9058639965</h3>

</div>

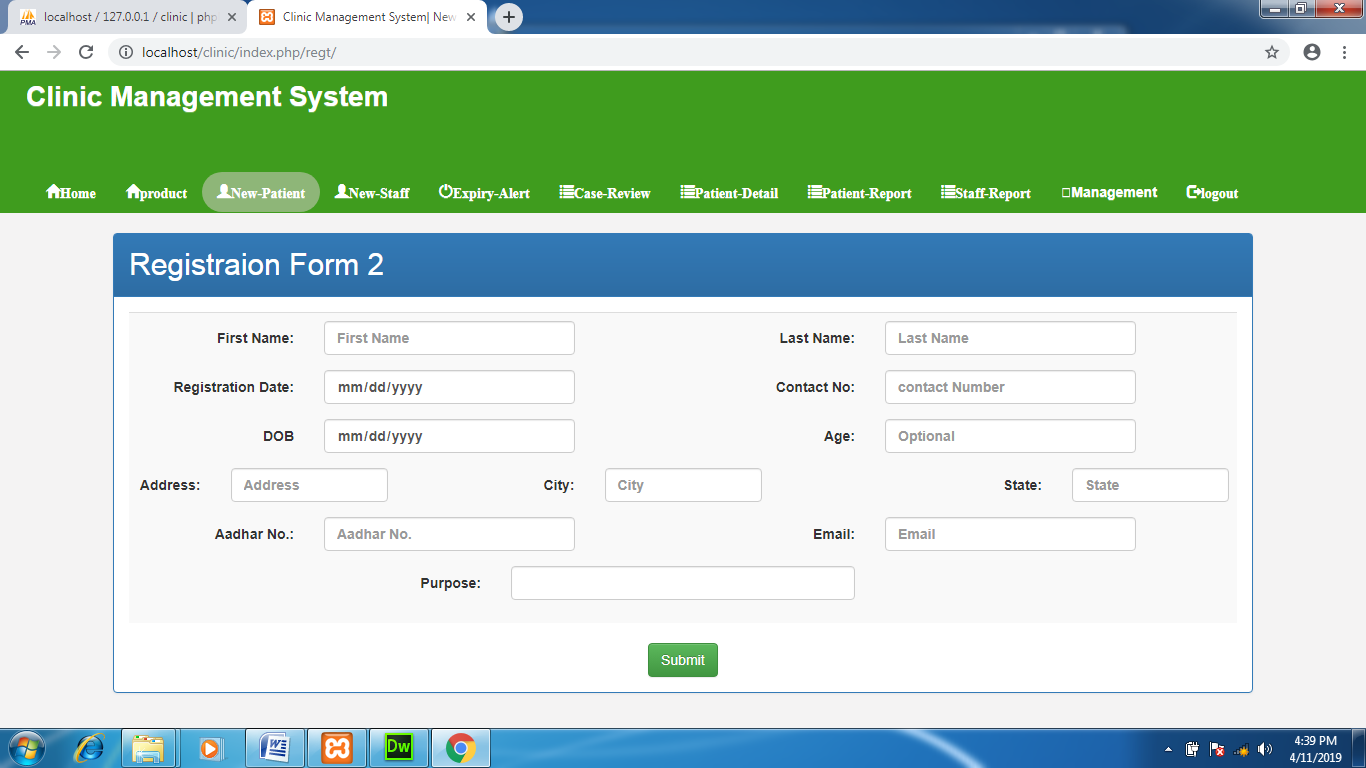
</div>

</div>

</section>

**6.3: New Patient Registration :New.php**

When you choose ‘New’ tab in the navigation menu, this button will bring you to this page regis.php .

****

**CODING**

**6.3.1: View of regis.php:**

<div class="text-center text-success"><?php echo $this->session->flashdata('msg'); ?>

</div>

<div class="container">

<div class="panel panel-primary">

<div class="panel-heading">

<font size="+2">Registraion Form <?php echo$this->session->userdata('user\_id')?></font>

</div>

<div class="panel-body">

<form class="form-horizontal" method="post" action="<?php echo base\_url(); ?>index.php/regt/new\_register/">

<table class="table table-striped">

<tbody><th>

<input type="text" class="hidden" name="user\_id" placeholder="First Name" class="form-control" required="true" value="<?php echo$this->session->userdata('user\_id')?>">

<div class="form-group">

<label class="control-label col-md-2">First Name: </label>

<div class="col-md-3">

<input type="text" name="first" placeholder="First Name" class="form-control" required="true">

</div>

<div class="col-md-1"></div>

<label class="control-label col-md-2">Last Name: </label>

<div class="col-md-3">

<input type="text" name="last" placeholder="Last Name" class="form-control" required="true">

</div>

<div class="col-md-1"></div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Registration Date: </label>

<div class="col-md-3">

<input type="date" name="date" class="form-control" placeholder="mm/dd/yyyy" required="true">

</div>

<div class="col-md-1"></div>

<label class="control-label col-md-2">Contact No: </label>

<div class="col-md-3">

<input type="text" name="contact" placeholder="contact Number" class="form-control" required="true">

</div>

<div class="col-md-1"></div>

</div>

<div class="form-group">

<label class="control-label col-md-2">DOB </label>

<div class="col-md-3">

<input type="date" id="dob" name="dob" class="form-control" placeholder="mm/dd/yyyy" required="true" onkeypress="return isNumberKey(event);">

</div>

<div class="col-md-1"></div>

<label class="control-label col-md-2">Age:</label>

<div class="col-md-3">

<input type="text" name="age" id="age" onkeypress="return isNumberKey(event);" placeholder="Optional" class="form-control" onFocus="getage()">

</div>

<div class="col-md-1"></div>

</div>

<div class="form-group">

<label class="control-label col-md-1">Address: </label>

<div class="col-md-2">

<input type="text" name="address" placeholder="Address" class="form-control" required="true">

</div>

<label class="control-label col-md-2">City:</label>

<div class="col-md-2">

<input type="text" name="city" placeholder="City" class="form-control" required="true">

</div>

<div class="col-md-1"></div>

<label class="control-label col-md-2">State:</label>

<div class="col-md-2">

<input type="text" name="state" placeholder="State" class="form-control" required="true">

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Aadhar No.: </label>

<div class="col-md-3">

<input type="text" name="aadhar" placeholder="Aadhar No." class="form-control" required="true">

</div>

<div class="col-md-1"></div>

<label class="control-label col-md-2">Email: </label>

<div class="col-md-3">

<input type="email" name="email" placeholder="Email" class="form-control" required="true">

</div>

<div class="col-md-1"></div>

</div>

<div class="form-group">

<div class="col-md-2"></div>

<label class="control-label col-md-2">Purpose: </label>

<div class="col-md-4">

<input type="text" name="purpose" class="form-control" required="true">

</div>

<div class="col-md-1"></div>

</div>

</th>

</tbody>

</table>

<table align="center">

<td>

<button type="submit" class="btn btn-success" name="register">Submit</button></td>

</table>

</div>

</div>

</form>

</div><!--end of container-->

<script>

$('#dob,#age').on('keyup',getage);

function getage(){

var today = new Date();

var n = today.getFullYear();

var dob = new Date($('#dob').val());

var d=dob.getFullYear();

var age=n-d;

//var m = today.getMonth() - birthDate.getMonth();

//var restdue=fin\_amount-insamount;

//$('#restdue').val(restdue);

//var dob = '19800810';

$('#age').val(age);

}

</script>

**6.3.2:Controller of new :Regt.php**

class Regt extends CI\_controller{

public function \_\_construct(){

parent::\_\_construct();

$this->load->model('Database\_model');

if($this->session->userdata('user\_id')===NULL){

redirect("/login/");

}

}

public function index(){

$data=array();

$data['title']="New Registration";

$this->load->view('header',$data);

$this->load->view('nav');

$this->load->view('regis');

$this->load->view('footer');

}

public function new\_register(){

if($this->input->post('register')!==NULL){

$data['user\_id']=$this->input->post('user\_id');

$data['first']=$this->input->post('first');

$data['last']=$this->input->post('last');

$data['date']=$this->input->post('date');

$data['contact']=$this->input->post('contact');

$data['dob']=$this->input->post('dob');

$data['age']=$this->input->post('age');

$data['address']=$this->input->post('address');

$data['city']=$this->input->post('city');

$data['state']=$this->input->post('state');

$data['aadhar']=$this->input->post('aadhar');

$data['email']=$this->input->post('email');

$data['purpose']=$this->input->post('purpose');

$run=$this->Database\_model->insertbyarray("new",$data);

if($run===true){

//$user\_id=$this->input->('id');

//$\_SESSION["userId"]=$id;

$dob=$this->input->post('dob');

$where="`dob`='$dob'";

$getpid=$this->Database\_model->getlastdata("new",$where);

echo $pid=$getpid['id'];

$msg="<div class='alert alert-success'><strong>Record Added Sucessfully Your ID IS'$pid' </strong></div>";

$arr=array("msg"=>$msg);

$this->session->set\_flashdata($arr);

redirect("/regt/print\_pref/?id=$pid");

}

}

else{

redirect("/regt/");

}

}

**6.3.3:Model of New :Database\_Model.php**

public function insertbyquery($table,$columns,$values){

$query="INSERT INTO $table ($columns) VALUES ($values)";

//echo $query;

$run=$this->db->query($query);

if($run){

return true;

}

}

public function insertbyarray($table,$data){

$run=$this->db->insert($table,$data);

if($run){

return true;

}

}

**6.4:New Staff Registration**

**CODING**

When you chose ‘new-staff option in the nav menu login itwill bring you to this page staff.php.

**6.4.1:View Of New-staff: staff.php**

<div class=”text-center text-success”><?php echo $this->session->flashdata(‘msg’); ?></div>

<div class=”container”>

<div class=”panel panel-primary”>

<div class=”panel-heading”>

<font size=”+2”>Registraion Form</font>

</div>

<div class=”panel-body”>

<form class=”form-horizontal” method=”post” action=”<?php echo base\_url(); ?>index.php/stf/staff\_register/”>

<table class=”table table-striped”>

<tbody><th>

<input type=”text” name=”user\_id” class=”hidden” value=”<?php echo $this->session->userdata(‘user\_id’);?>” required=”true”>

<div class=”form-group”>

<label class=”control-label col-md-2”>Name: </label>

<div class=”col-md-3”>

<input type=”text” name=”name” placeholder=”Enter Name” class=”form-control” required=”true”>

</div>

<div class=”col-md-1”></div>

<label class=”control-label col-md-2”>Date Of Joining: </label>

<div class=”col-md-3”>

<input type=”date” name=”date” placeholder=”Date Of Joining” class=”form-control” required=”true”>

</div>

<div class=”col-md-1”></div>

</div>

<div class=”form-group”>

<label class=”control-label col-md-2”>Address: </label>

<div class=”col-md-3”>

<input type=”text” name=”address” class=”form-control” placeholder=”Address” required=”true”>

</div>

<div class=”col-md-1”></div>

<label class=”control-label col-md-2”>Contact No: </label>

<div class=”col-md-3”>

<input type=”text” name=”contact” placeholder=”contact Number” class=”form-control” required=”true”>

</div>

<div class=”col-md-1”></div>

</div>

<div class=”form-group”>

<label class=”control-label col-md-2”>Aadhar: </label>

<div class=”col-md-3”>

<input type=”text” name=”aadhar” class=”form-control” placeholder=”Aadhar” required=”true”>

</div>

<div class=”col-md-1”></div>

<label class=”control-label col-md-2”>Age:</label>

<div class=”col-md-3”>

<input type=”text” name=”age” placeholder=”Optional” class=”form-control”>

</div>

<div class=”col-md-1”></div>

</div>

<div class=”form-group”>

<label class=”control-label col-md-2”>Salary: </label>

<div class=”col-md-3”>

<input type=”text” name=”salary” placeholder=”Salary” class=”form-control” required=”true”>

</div>

<div class=”col-md-1”></div>

<label class=”control-label col-md-2”>Profile: </label>

<div class=”col-md-3”>

<select class=”form-control” name=”profile”>

<option>--select--</option>

<option>Volunteers</option>

<option>Clinical Asistants</option>

<option>Porters</option>

<option>Patient services ssistants</option>

</select>

</div>

<div class=”col-md-1”></div>

</div>

</th>

</tbody>

</table>

<table align=”center”>

<td><button type=”submit” class=”btn btn-success” name=”register”>Submit</button></td>

</table>

</div> </div>

</form>

</div><!—end of container🡪

**6.4.2:Controller of New Staff: stf/staff\_register.php**

public function staff\_register(){

if($this->input->post(‘register’)!==NULL){

//$user\_id=$\_SESSION[‘user\_id’];

$data[‘user\_id’]=$this->input->post(‘user\_id’);

$data[‘name’]=$this->input->post(‘name’);

$data[‘date’]=$this->input->post(‘date’);

$data[‘address’]=$this->input->post(‘address’);

$data[‘contact’]=$this->input->post(‘contact’);

$data[‘aadhar’]=$this->input->post(‘aadhar’);

$data[‘age’]=$this->input->post(‘age’);

$data[‘salary’]=$this->input->post(‘salary’);

$data[‘profile’]=$this->input->post(‘profile’);

$run=$this->Database\_model->insertbyarray(“staff”,$data);

if($run===true){

//$user\_id=$this->input->(‘id’);

//$\_SESSION[“userId”]=$id;

$contact=$this->input->post(‘contact’);

$where=”`contact`=$contact”;

$getptid=$this->Database\_model->getlastdata(“staff”,$where);

$pid=$getptid[‘id’];

$msg=”<div class=’alert alert-success’><strong>Record Added Sucessfully Your ID IS’$pid’ </strong></div>”;

$arr=array(“msg”=>$msg);

$this->session->set\_flashdata($arr);

redirect(“/stf/”);

}

}

else{

redirect(“/stf/”);

}

}

**6.4.3:Model of New-Staff: Database\_Model**

public function insertbyquery($table,$columns,$values){

$query="INSERT INTO $table ($columns) VALUES ($values)";

//echo $query;

$run=$this->db->query($query);

if($run){

return true;

}

}

public function getlastdata($table,$where){

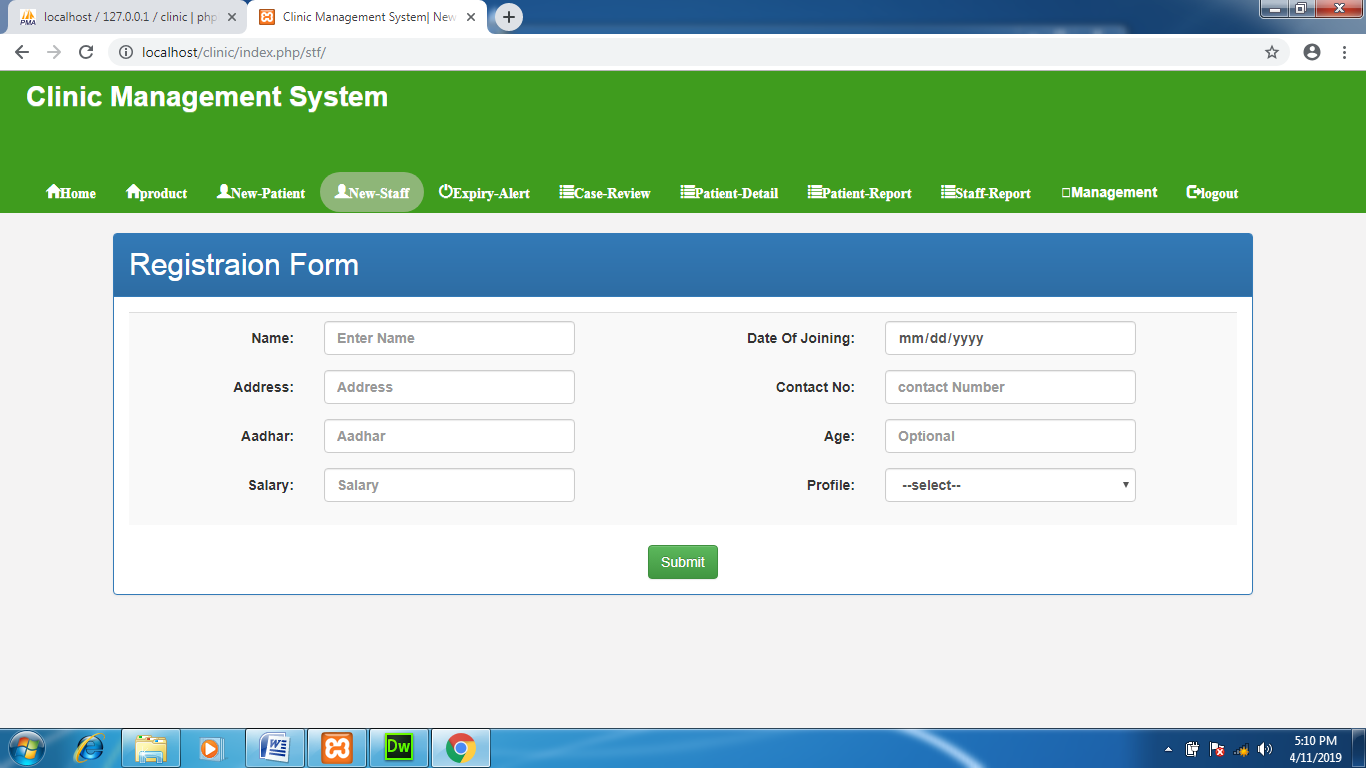
$query="SELECT \* from $table where $where order by `id` desc limit 1";

$run=$this->db->query($query);

$result=$run->row\_array();

return $result;

}

****

**6.5: Expiry Alert:**

**CODING**

**6.5.1:View of Expiry-Alert: expiry.php**

<section class="container">

<div class="panel panel-primary">

<div class="panel-heading">

<font size="+2">Reminder</font>

</div>

<div class="panel-body">

<div class="row">

<div col-md-12>

<?php

foreach($array as $regt)

{

?>

<table class="table table-bordered">

<tbody align="center" class="navbg">

<tr> <td>Patient Id:</td>

<td>Patient First Name:</td>

<td>Patient Last Name:</td>

<td>Patient Address:</td>

<td>Patient Phone:</td>

<td>purpose:</td>

<td>Review-Date:</td>

</tr>

<tr>

<td>

<?php

//echo $dept->dname;

echo $regt['p\_id'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['first'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['last'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['address'];

?>

</td>

<td>

<?php echo $regt['contact'];?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['purpose'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['rewdate'];

?>

</td>

</tr>

<?php

}

//print\_r($pagefilters);

// $pagefilters=http\_build\_query($pagefilters);

// if($pagefilters!=""){

// $pagefilters="?".$pagefilters;

//}

if($pages>1)

{

?>

<tr>

<td colspan="8">

<?php

echo $paging;

?><?php /\*?><ul class="pagination">

<?php

for($i=1;$i<=$pages;$i++){

?>

<li class="page-item">

<a class="page-link"

href="<?php echo base\_url(); ?>index.php/regt/customerlist/page/<?php echo $i; ?>/<?php echo $pagefilters; ?>"><?php \*/?>

<?php // echo $i; ?>

</a>

<!-- </li>

</ul>-->

</td>

</tr>

</tbody>

</table>

<?php

}

**?>**

</div><!--end pof panel body-->

</div>

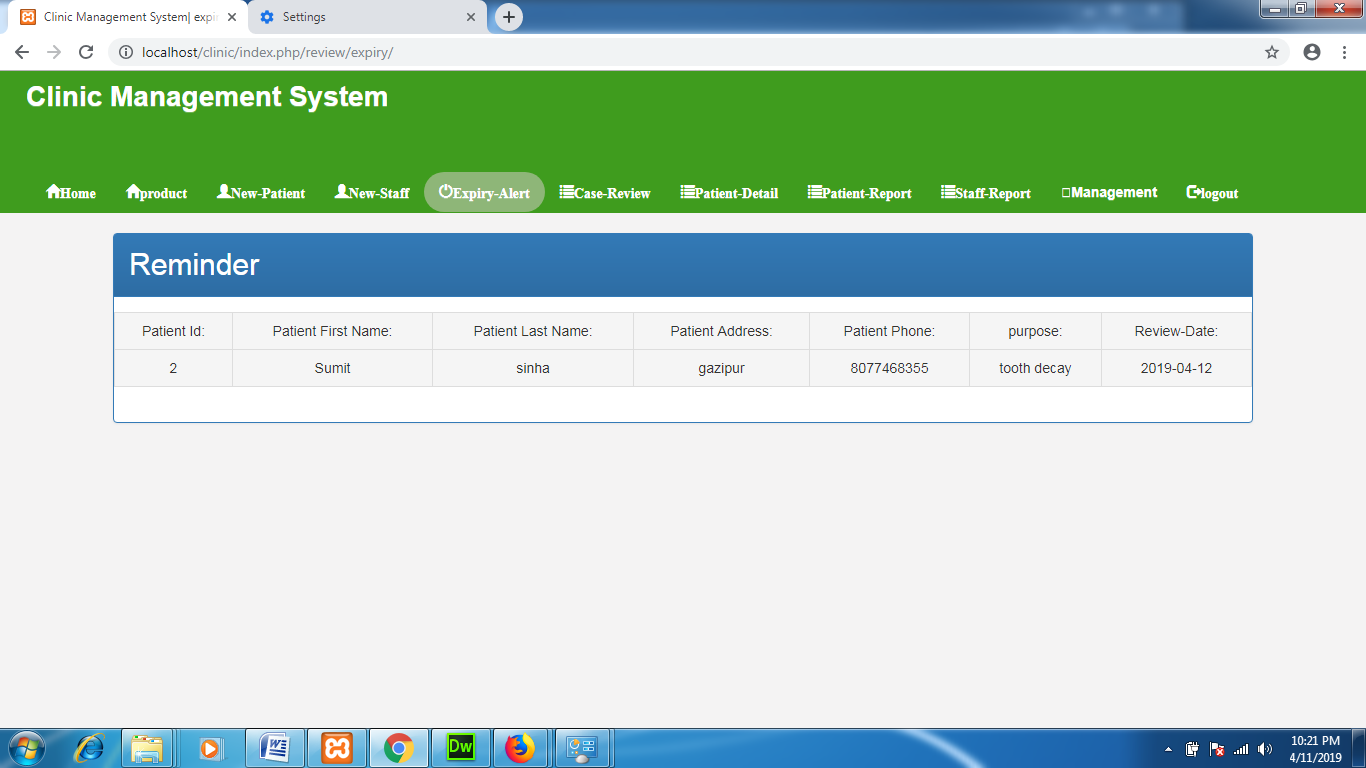
</div>

</div><!-- end of container-->

</div><!-- end of panel body-->

</div> <!--- end of main panel-->

</section><!--end of section-->

****

**6.6: Case Review:**

When you choose ‘Case Review’ option in the nav menu it will bring you to this page getid.php.

**CODING**

**6.6.1: View of Case Review:Getid.php**

<div class="text-center text-success"><?php echo $this->session->flashdata('msg'); ?></div>

<div class="container">

<div class="tab-pane">

<div class="panel panel-primary">

<div class="panel-heading">

<font size="+2">Case Review</font>

</div>

<div class="panel-body">

<div class="row"> <div class="col-md-2"> </div>

<div class="col-md-8">

<form method="post" action="<?php echo base\_url(); ?>index.php/review/getcid/">

<table class="table table-bordered">

<div class="form-group">

<label class="col-md-4 control-label">Patient-ID/Phone</label>

<div class="col-md-4 inputGroupContainer">

<div class="input-group">

<span class="input-group-addon"><i class="glyphicon glyphicon-user"></i></span>

<input name="id" placeholder="Patient-Id/phone" class="form-control" value="<?php //echo $payment['id']; ?>" type="number"></div>

</div>

<!-- <tbody><tr>

<th width="14%"><span class="glyphicon glyphicon-user"></span> Enter Customer ID</th>

<td width="20%"><input type="text" name="id" required class="form-control">

</td>

</tr>-->

<table align="center">

<tr><td>

<button type="submit" class="btn btn-success" name="review">Submit</button></td>

</tr>

</table>

<!-- </tbody>-->

</table>

</form>

</div>

</div>

</div>

</div>

</div>

</div>

</div><!-- container -->

**CODING:**

**6.6.2:Controller of Case-Review:**

<?php

class Review extends CI\_controller{

public function \_\_construct(){

parent::\_\_construct();

$this->load->model('Database\_model');

if($this->session->userdata('user\_id')===NULL){

redirect("/login/");

}

}

public function index(){

$data=array();

$data['title']="Review";

$this->load->view('header',$data);

$this->load->view('nav');

//$this->load->view('review');

$this->load->view('footer');

}

public function getid(){

$data=array();

$data['title']="update info";

$this->load->view('header',$data);

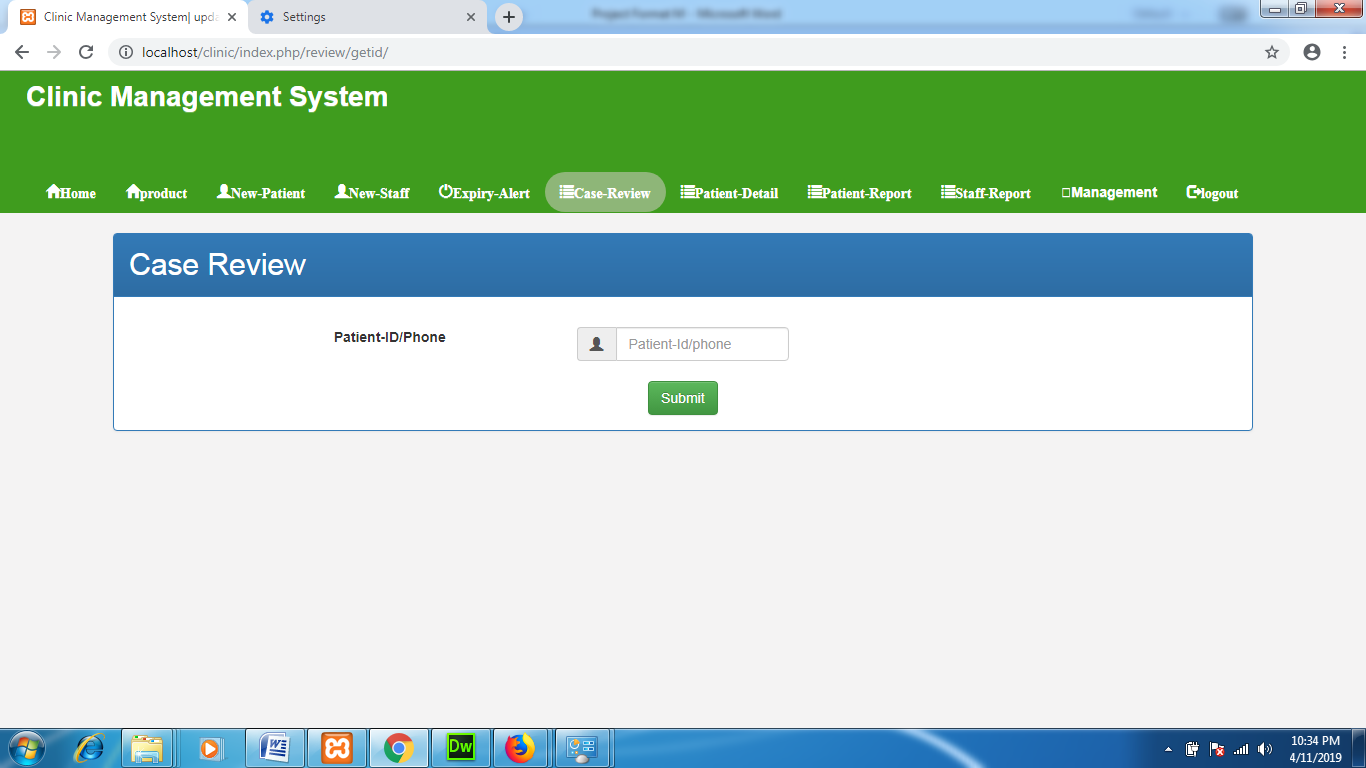
$this->load->view('nav');

$this->load->view('getid',$data);

$this->load->view('footer');

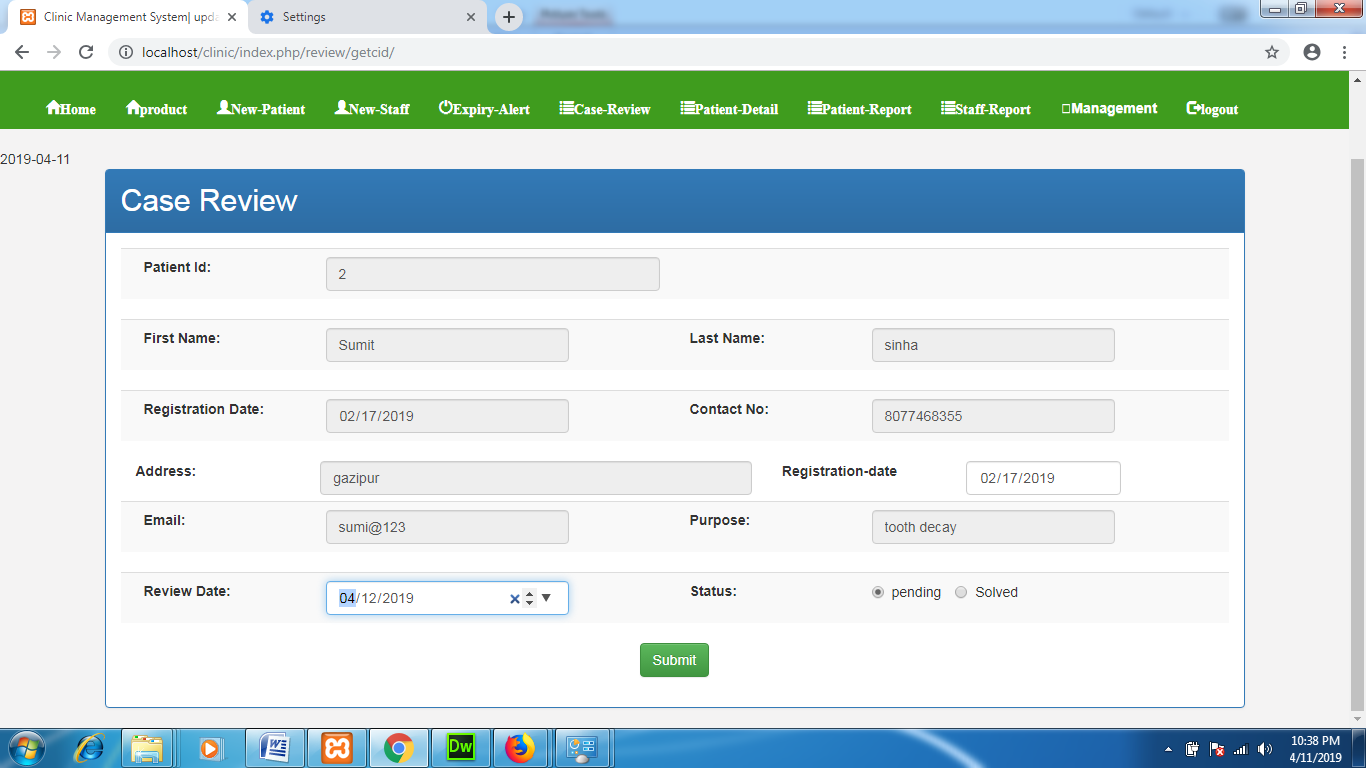
}

?>



**6.7 : GETCID**

On Submitting The patient id Previous record of the Case of the patient will be fetched and status will be update with next date for review.

****

After submitting the Review form with Review date admin as well as Patient will get notification for the next date to visit the clinic.

**CODING**

**6.7.1Controller Of Review Date:Getcid.php**

public function getcid(){

if($this->input->post('review')!==NULL){

$table="new";

$user\_id=$\_SESSION['user\_id'];

$id=$this->input->post('id');

$where="id=$id and user\_id=$user\_id";;

$review=$this->Database\_model->getsingledata($table,$where);

if($review!==NULL){

$this->load->model('Database\_model');

$depts=$this->Database\_model->getrows($table,$where="");

$data=array();

$data['title']="update info";

$data['review']=$review;

$this->load->view('header',$data);

$this->load->view('nav');

$this->load->view('getcid',$data);

$this->load->view('footer');

}

else{

$msg= "<div class='alert alert-info'><strong>No id found</strong></div>";

$arr=array("msg"=>$msg);

$this->session->set\_flashdata($arr);

redirect('/review/getid/');

}

}

else{

redirect('/review/getid/');

}

}

public function get\_review(){

if($this->input->post('rew')!==NULL){

$rev\_date=date('Y-m-d');

$data['user\_id']=$this->input->post('user\_id');

$data['rev\_date']=$rev\_date;

$data['p\_id']=$this->input->post('id');

$data['first']=$this->input->post('first');

$data['last']=$this->input->post('last');

$data['rdate']=$this->input->post('rdate');

$data['contact']=$this->input->post('contact');

$data['email']=$this->input->post('email');

$data['purpose']=$this->input->post('purpose');

$data['rewdate']=$this->input->post('rewdate');

$data['address']=$this->input->post('address');

$data['status']=$this->input->post('status');

$run=$this->Database\_model->insertbyarray($table="review",$data);

$data=array();

//$data['title']="get\_review";

//$data['review']=$review;

$this->load->view('header',$data);

$this->load->view('nav');

$this->load->view('getcid',$data);

$this->load->view('footer');

if($run===true){

//$user\_id=$this->input->post('id');

//$where="`id`='$user\_id'";

//$reg=$this->Database\_model->update("new",$data1,$where);

//$getpid=$this->Database\_model->getlastdata("review","`user\_id`='$user\_id'");

//$pid=$getpid['id'];

$msg= "<div class='alert alert-info'><strong>Record Updated Sucessfully</strong></div>";

$arr=array("msg"=>$msg);

$this->session->set\_flashdata($arr);

redirect("review/getid/");

}

}

else{

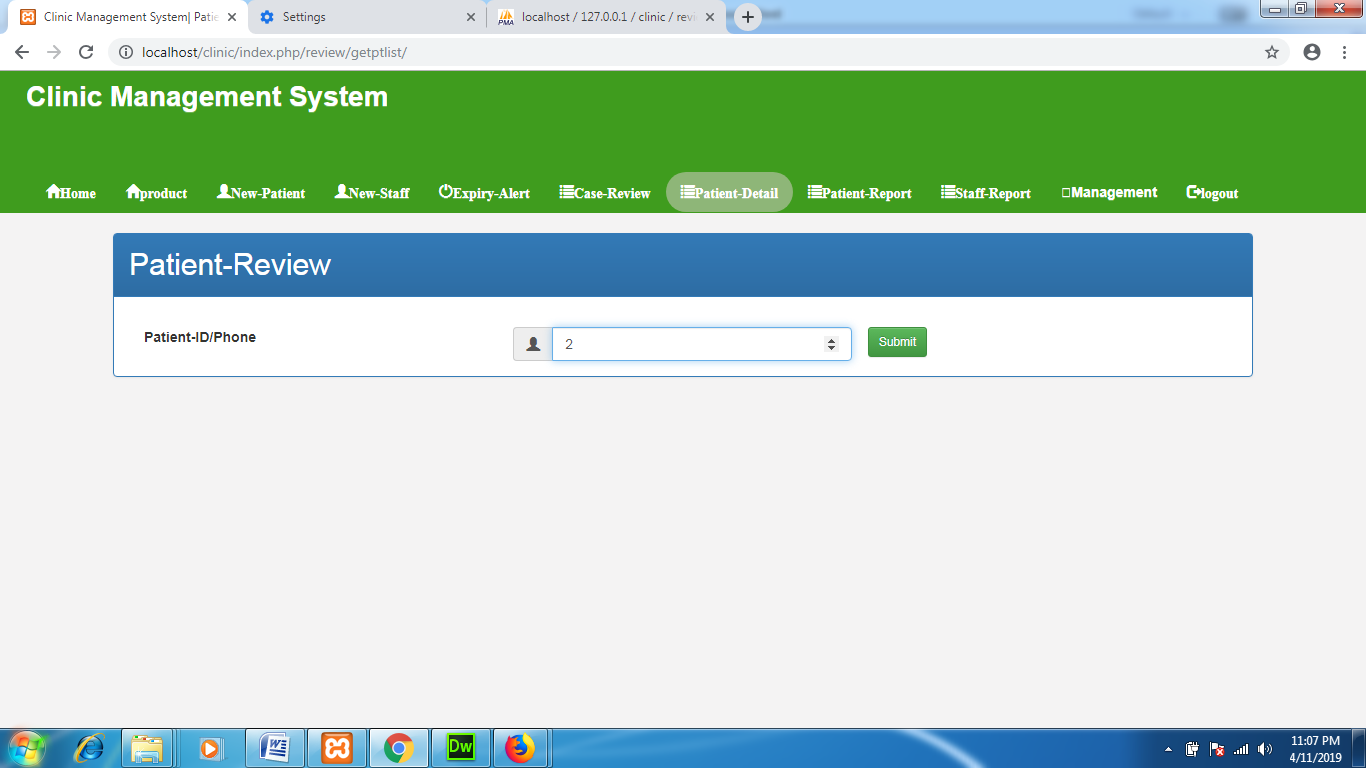
redirect("/getcid/");

}

}

**6.8 : Patient\_Detail:Getptlist.php**

When you choose ‘Patient-Detail’option in the nav menu it will bring you to this page getptlist.php.



**CODING**

**6.8.1:View Of Patient\_Detail:Getptlist.php.**

<div class="text-center"><?php echo $this->session->flashdata('msg'); ?></div>

<div class="container">

<div class=”panel panel-primary”>

<div class=”panel-heading”>

<font size=”+2”>Patient-Review</font>

</div>

<div class=”panel-body”>

<form action=”<?php echo base\_url(); ?>index.php/review/getptdisplay/”>

<div class=”form-group”>

<label class=”col-md-4 control-label”>Patient-ID/Phone</label>

<div class=”col-md-4 inputGroupContainer”>

<div class=”input-group”>

<span class=”input-group-addon”><I class=”glyphicon glyphicon-user”></i></span>

<input name=”query” placeholder=”Patiend-ID/phone” class=”form-control” value=”” type=”number”></div>

</div>

</div>

<input type=”submit” name=”search” class=”btn btn-success btn-sm”>

</form>

</div>

</div>

</div>

</div>

</div>

</form>

</div>

</div><!—end of container🡪

**6.8.2:Cotroller of Patient-Detail:**

public function getptlist(){

$data=array();

$data['title']="Patient-list";

$this->load->view('header',$data);

$this->load->view('nav');

$this->load->view('getptlist',$data);

$this->load->view('footer');

}

public function getptdisplay(){

$data=array();

$data['title']="Patient-display";

$count=5;

if($this->uri->segment('4')){

$page=$this->uri->segment('4');

}

else{

$page=1;

}

$offset=($page-1)\*$count;

$limit="$offset,$count";

$pagefilters=array();

$table="review";

$where=1;

$query="";

if($this->input->get('query')!=NULL)

{

$query=trim($this->input->get('query'));

$where.=" and (p\_id = '$query')";

$pagefilters['query']=$query;

}

else

{

$msg= "<div class='alert alert-info'><strong>No id found</strong></div>";

$arr=array("msg"=>$msg);

$this->session->set\_flashdata($arr);

redirect('/review/getptlist/');

}

$array=$this->Database\_model->getrows($table,$where,$limit);

$rowcount=$this->Database\_model->getcount($table,$where);

$pages=ceil($rowcount/$count);

$data['array']=$array;

$data['query']=$query;

$patient=$this->Database\_model->getsingledata("new","contact = '$query' or id = '$query'");

$data['rowcount']=$patient['id']-$offset;

$data['pages']=$pages;

$data['pagefilters']=$pagefilters;

$this->load->library('paging');

$url=base\_url()."index.php/review/getptdisplay/";

$paging=$this->paging->createpaging($url,$pages,$pagefilters);

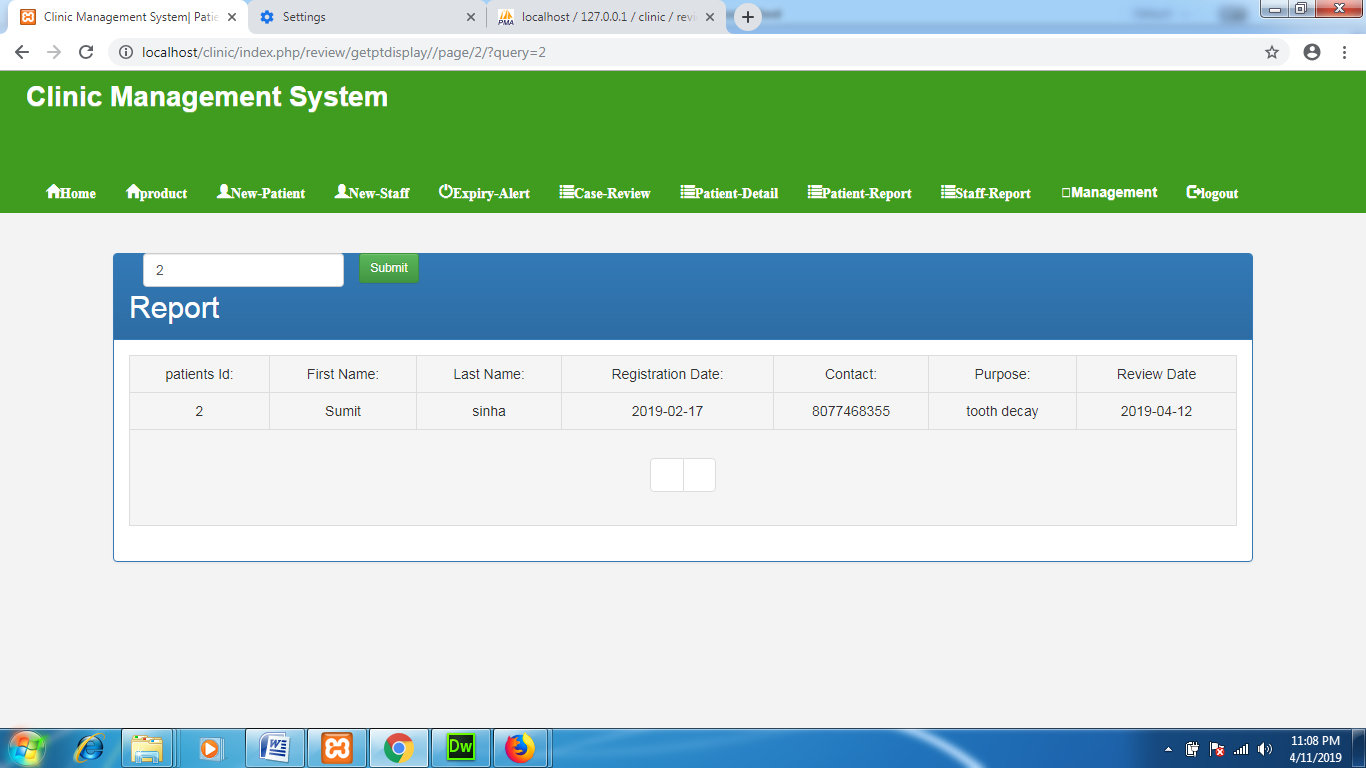
$data['paging']=$paging;

$this->load->view('header',$data);

$this->load->view('nav');

$this->load->view('casereview',$data);

$this->load->view('footer');



When patient id is submitted it will fetch the records of the individual patient i.e how many times he/she revisited the clinic.

**6.9: Patient-Report: Patientlist.php**

**CODING**

**6.9.1View OF Patient-Report:Patientlist.php**

<div class="container">

<div class="text-center"><?php echo $this->session->flashdata('msg'); ?></div>

<div class="collapse details" id="details"></div>

<div class="col-md-12"><br>

<form action="<?php echo base\_url(); ?>index.php/regt/patientlist/" method="get">

<div class="col-md-4">

<div class="row">

<div class="col-md-10">

<input type="number" name="query" class="form-control" placeholder="patient-phone/id" value="<?php echo $query; ?>">

</div>

<input type="submit" name="search" class="btn btn-success btn-sm">

</div>

</div>

</form>

</div>

<br><div class="row">

<div col-md-12>

<div class="container-fluid">

<div class="panel panel-primary">

<div class="panel-heading">

<font size="+2">Report</font>

</div>

<div class="panel-body">

<?php

//print\_r($array);

if(empty($array)){

echo "<div class='alert alert-danger glyphicon glyphicon-ban-circle'><strong>No Record Found Or Input Mismatch</strong></div>";

}

else{

foreach($array as $regt){

?>

<table class="table table-bordered">

<tbody align="center" class="navbg">

<tr><td>Patient-Id:</td>

<td>Patient First Name:</td>

<td>Patient Last Name:</td>

<td>Patient Address:</td>

<td>Patient Contact:</td>

<td>Patient Age:</td>

<td>Patient Aadhar:</td>

<td>Patient Email:</td>

<td>Purpose:</td>

<td>Modify:</td>

</tr>

<tr>

<td>

<?php

//echo $dept->dname;

echo $regt['id'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['first'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['last'];

?>

</td>

<td>

<?php echo $regt['address'];?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['contact'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['age'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['aadhar'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['email'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['purpose'];

?>

</td>

<td><a href="<?php echo base\_url(); ?>index.php/regt/editpatient/<?php echo $regt['id']; ?>">Modify</a></td>

</tr>

<?php

}

//print\_r($pagefilters);

$pagefilters=http\_build\_query($pagefilters);

if($pagefilters!=""){

$pagefilters="?".$pagefilters;

}

if($pages>1){

?>

<tr>

<td colspan="8">

<?php

echo $paging;

?><?php /\*?><ul class="pagination">

<?php

for($i=1;$i<=$pages;$i++){

?>

<li class="page-item">

<a class="page-link"

href="<?php echo base\_url(); ?>index.php/regt/customerlist/page/<?php echo $i; ?>/<?php echo $pagefilters; ?>"><?php \*/?>

<?php // echo $i; ?>

</a>

<!-- </li>

</ul>-->

</td>

</tr>

</tbody>

</table>

<?php

}

}

?>

</div><!--end pof panel body-->

</div></div>

</div>

</div>

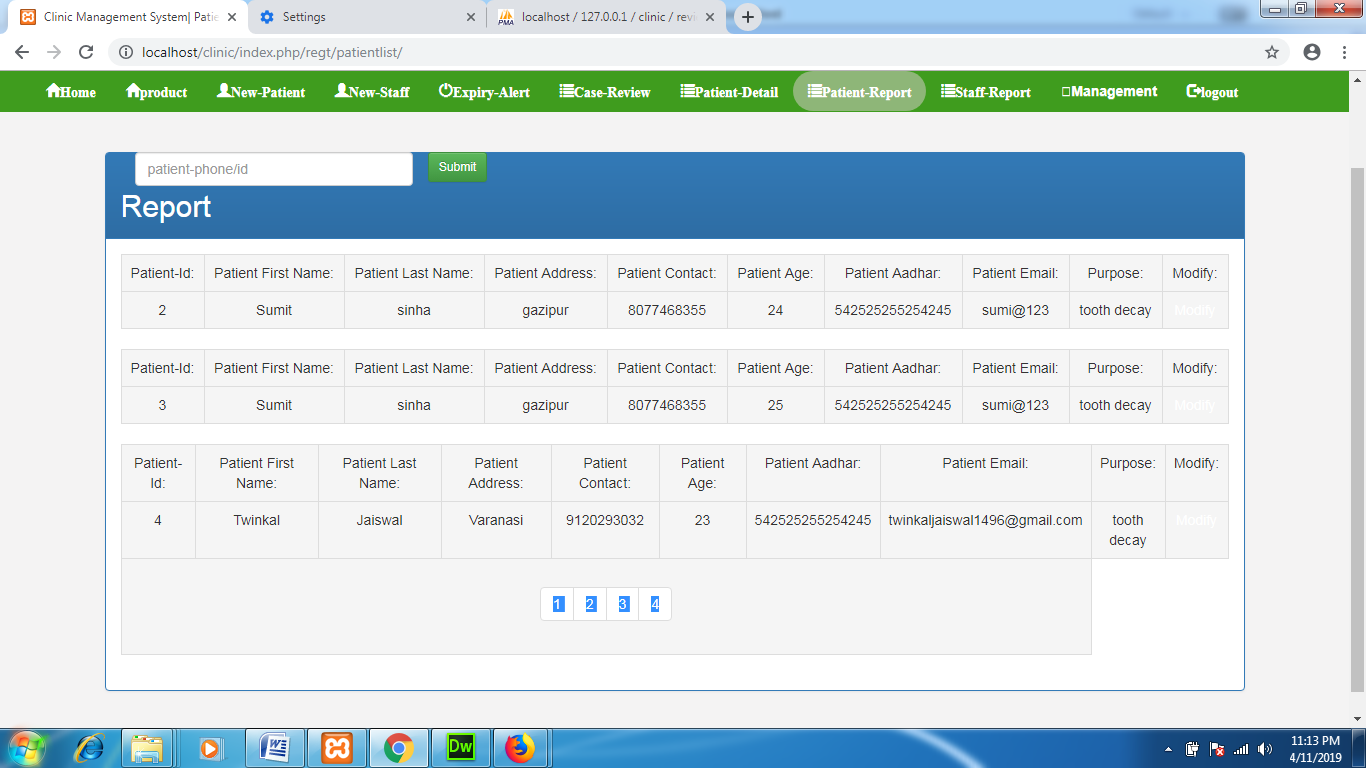
</div>

</div>

</div><!--end of row-->

</div><!--- end of second container--> </div>

<!--end of main container--><br><br>

****

This page shows the Total number of Patient Available in the record

**6.9.2Controller Of Patient\_Report: Regt.php**

public function patientlist(){

$data=array();

$data['title']="Patient-List";

$count=3;

if($this->uri->segment('4')){ $page=$this->uri->segment('4'); }

else{

$page=1;

}

$offset=($page-1)\*$count;

$limit="$offset,$count";

$pagefilters=array();

$table="new";

$user\_id=$\_SESSION['user\_id'];

$where="user\_id=$user\_id";

$query="";

$pagefilters['query']=$query;

if($this->input->get('query')!=NULL){

$query=trim($this->input->get('query'));

$where.=" and (contact ='$query' or id ='$query')";

}

$array=$this->Database\_model->getrows($table,$where,$limit);

$rowcount=$this->Database\_model->getcount($table,$where);

$pages=ceil($rowcount/$count);

$data['array']=$array;

$data['query']=$query;

$data['pages']=$pages;

$data['pagefilters']=$pagefilters;

$this->load->library('paging');

$url=base\_url()."index.php/regt/patientlist/";

$paging=$this->paging->createpaging($url,$pages,$pagefilters);

$data['paging']=$paging;

$this->load->view('header',$data);

$this->load->view('nav');

$this->load->view('patientlist',$data);

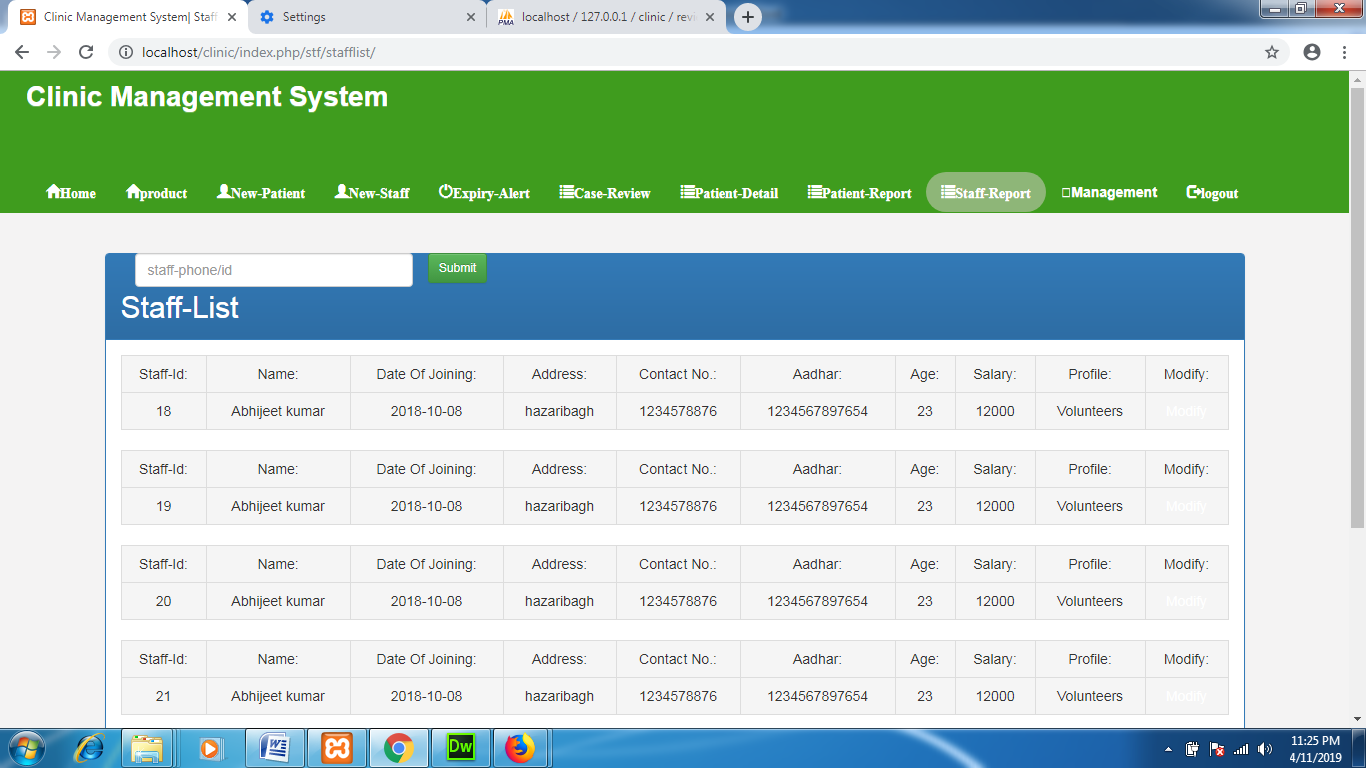
$this->load->view('footer');

}

}

**6.10 : Sttaff Report**

When we choose ‘Staff-Report’ Option in the nav menu,it will bring to the stafflist.php.



**CODING**

**6.10.1View Of Staff-Report:Stafflist.php.\**

<div class="container">

<div class="text-center"><?php echo $this->session->flashdata('msg'); ?></div>

<div class="collapse details" id="details"></div>

<div class="col-md-12"><br>

<form action="<?php echo base\_url(); ?>index.php/stf/stafflist/" method="get">

<div class="col-md-4">

<div class="row">

<div class="col-md-10">

<input type="number" name="query" class="form-control" placeholder="staff-phone/id" value="<?php echo $query; ?>">

</div>

<input type="submit" name="search" class="btn btn-success btn-sm">

</div>

</div>

</form>

</div>

<br><div class="row">

<div col-md-12>

<div class="container-fluid">

<div class="panel panel-primary">

<div class="panel-heading">

<font size="+2">Staff-List</font>

</div>

<div class="panel-body">

<?php

//print\_r($array);

if(empty($array)){

echo "<div class='alert alert-danger glyphicon glyphicon-ban-circle'><strong>No Record Found Or Input Mismatch</strong></div>";

}

else{

foreach($array as $regt){

?>

<table class="table table-bordered">

<tbody align="center" class="navbg">

<tr><td>Staff-Id:</td>

<td>Name:</td>

<td>Date Of Joining:</td>

<td>Address:</td>

<td>Contact No.:</td>

<td>Aadhar:</td>

<td>Age:</td>

<td>Salary:</td>

<td>Profile:</td>

<td>Modify:</td>

</tr>

<tr>

<td>

<?php

//echo $dept->dname;

echo $regt['id'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['name'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['date'];

?>

</td>

<td>

<?php echo $regt['address'];?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['contact'];

?>

</td>

<td> <?php

//echo $dept->dname;

echo $regt['aadhar'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['age'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['salary'];

?>

</td>

<td>

<?php

//echo $dept->dname;

echo $regt['profile'];

?>

</td>

<td><a href="<?php echo base\_url(); ?>index.php/stf/editstaff/<?php echo $regt['id']; ?>">Modify</a></td>

</tr>

<?php

}

//print\_r($pagefilters);

$pagefilters=http\_build\_query($pagefilters);

if($pagefilters!=""){

$pagefilters="?".$pagefilters;

}

if($pages>1){

?>

<tr>

<td colspan="8">

<?php

echo $paging;

?><?php /\*?><ul class="pagination">

<?php

for($i=1;$i<=$pages;$i++){

?>

<li class="page-item">

<a class="page-link"

href="<?php echo base\_url(); ?>index.php/regt/customerlist/page/<?php echo $i; ?>/<?php echo $pagefilters; ?>"><?php \*/?>

<?php // echo $i; ?>

</a>

<!-- </li>

</ul>-->

</td>

</tr>

</tbody>

</table>

<?php

}

}

?>

</div><!--end pof panel body-->

</div></div>

</div>

</div>

</div>

</div>

</div><!--end of row-->

</div><!--- end of second container-->

</div>

<!--end of main container--><br><br>

**6.10.2:Controller of Patient-List:Stafflist.php**

public function stafflist(){

$data=array();

$data['title']="Staff-List";

$count=8;

if($this->uri->segment('4')){ $page=$this->uri->segment('4'); }

else{

$page=1;

}

$offset=($page-1)\*$count;

$limit="$offset,$count";

$pagefilters=array();

$table="staff";

$where=1;

$query="";

$pagefilters['query']=$query;

if($this->input->get('query')!=NULL){

$query=trim($this->input->get('query'));

$where.=" and (contact ='$query' or id ='$query')";

}

$array=$this->Database\_model->getrows($table,$where,$limit);

$rowcount=$this->Database\_model->getcount($table,$where);

$pages=ceil($rowcount/$count);

$data['array']=$array;

$data['query']=$query;

$data['pages']=$pages;

$data['pagefilters']=$pagefilters;

$this->load->library('paging');

$url=base\_url()."index.php/stf/stafflist/";

$paging=$this->paging->createpaging($url,$pages,$pagefilters);

$data['paging']=$paging;

$this->load->view('header',$data);

$this->load->view('nav');

$this->load->view('stafflist',$data);

$this->load->view('footer');

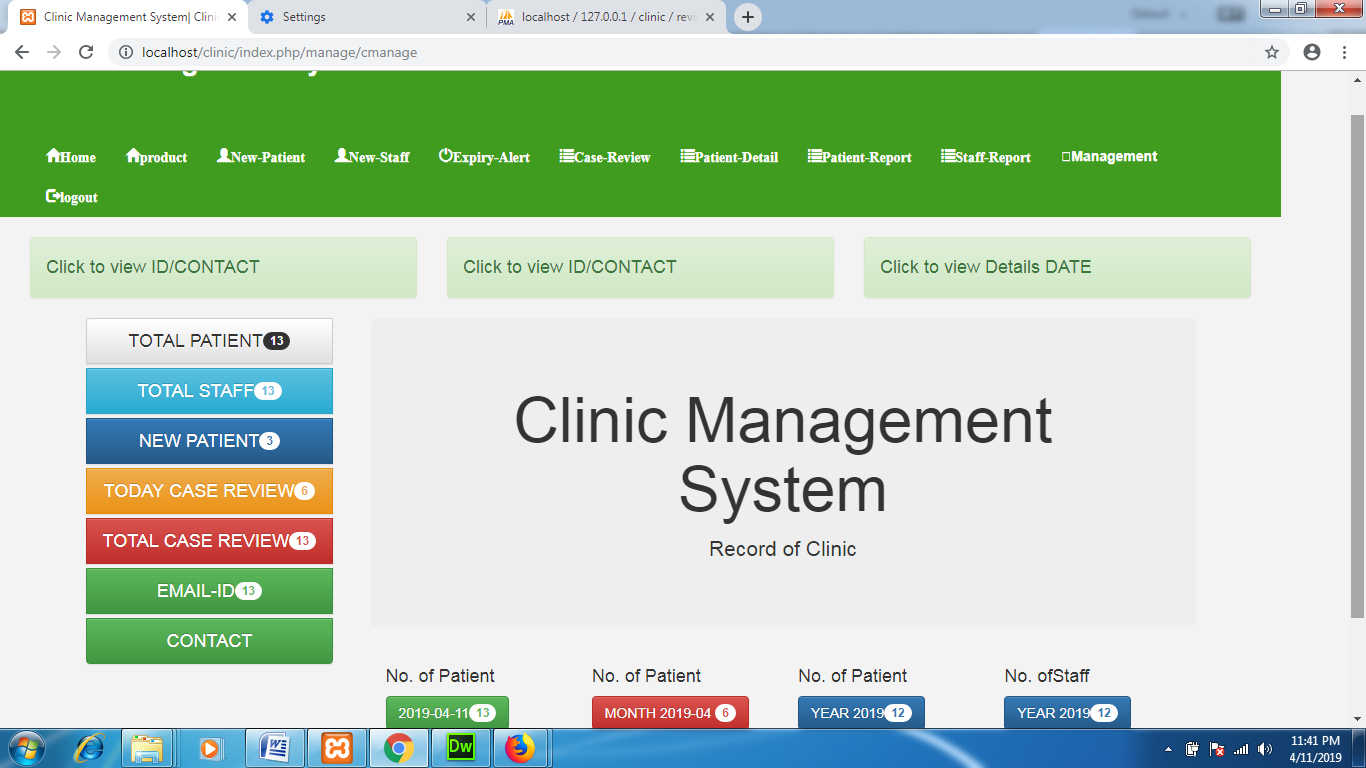
}

}

**6.11 : Management**

This page shows the Counting of overall records based on the different attributes like new patient, review patient, total no. of patient, today case handled etc.

.

****

**CODING**

**6.11.1: View Of Management:Cmange.php**

<?php

//print\_r($array1)

//foreach($array5 as $mail)

//echo $mail['email'];

//count($mail);

?>

<!doctype html>

<html>

<head>

<meta charset="utf-8">

<title>Untitled Document</title>

</head>

<body>

<div class="row">

<div class="container-fluid">

<div class="col-md-12">

<div class="col-md-4">

<div class="panel-group">

<div class="panel panel-success">

<div class="panel-heading"><h4 data-toggle="collapse" data-target="#panelcollapse">Click to view ID/CONTACT</h4></div>

<div class="panel-body collapse" id="panelcollapse">

<form action="<?php echo base\_url(); ?>index.php/manage/patientlist/" method="get">

<div class="row">

<div class="col-md-6">

<input type="text" name="query" class="form-control" placeholder="patient-phone/id" value="<?php //echo $query; ?>">

</div>

<input type="submit" name="search" class="btn btn-success btn-sm">

</div>

</div></form>

</div>

</div>

</div>

<div class="col-md-4">

<div class="panel-group">

<div class="panel panel-success">

<div class="panel-heading"><h4 data-toggle="collapse" data-target="#panelcollapse">Click to view ID/CONTACT</h4></div>

<div class="panel-body collapse" id="panelcollapse">

<form action="<?php echo base\_url(); ?>index.php/manage/patientlist/" method="get">

<div class="row">

<div class="col-md-6">

<input type="text" name="query" class="form-control" placeholder="patient-phone/id" value="<?php //echo $query; ?>">

</div>

<input type="submit" name="search" class="btn btn-success btn-sm">

</div>

</div></form>

</div>

</div>

</div>

<div class="col-md-4">

<div class="panel-group">

<div class="panel panel-success">

<div class="panel-heading"><h4 data-toggle="collapse" data-target="#panelcollapse1">Click to view Details DATE</h4></div>

<div class="panel-body collapse" id="panelcollapse1">

<form action="<?php echo base\_url(); ?>index.php/manage/patientlist/" method="get">

<div class="row">

<div class="col-md-6">

<input type="date" name="query" class="form-control" placeholder="patient-phone/id" value="<?php //echo $query; ?>">

</div>

<input type="submit" name="search" class="btn btn-success btn-sm">

</div>

</div>

</form>

</div>

</div>

</div>

</div><!--end of search-->

<!--<div class="col-md-2" style="float:left">

<div class="btn-group btn-group-vertical my-edit">

<button type="button" class="btn btn-default btn-sm ">Default Button</button>

<button type="button" class="btn btn-info btn-sm ">Info Button</button>

<button type="button" class="btn btn-primary btn-sm ">Primary Button</button>

<button type="button" class="btn btn-warning btn-sm">Warning Button</button>

<button type="button" class="btn btn-danger btn-sm">Danger Button</button>

<button type="button" class="btn btn-success btn-sm">Success Button</button>

<button type="button" class="btn btn-link btn-sm">Link Button</button>

</div><!-- //btn-group--></div>

<div class="row">

<div class="container">

<div class="col-sm-3">

<div class="btn-group btn-group-vertical my-edit " style="float:left; ">

<button type="button" class="btn btn-default btn-lg ">TOTAL PATIENT<span class="badge"><?php echo($rowcount);?></span></button>

<button type="button" class="btn btn-info btn-lg ">TOTAL STAFF<span class="badge"><?php echo($rowcount);?></span></button>

<button type="button" class="btn btn-primary btn-lg ">NEW PATIENT<span class="badge"><?php echo($rowcount1);?></span></button>

<button type="button" class="btn btn-warning btn-lg">TODAY CASE REVIEW<span class="badge"><?php echo($rowcount5);?></span></button>

<button type="button" class="btn btn-danger btn-lg">TOTAL CASE REVIEW<span class="badge"><?php echo($rowcount);?></span></button>

<button type="button" class="btn btn-success btn-lg" >EMAIL-ID<span class="badge"><?php echo($rowcount);?></span></button>

<button type="button" class="btn btn-success btn-lg">CONTACT</button>

</div><!-- //btn-group-->

</div>

<div class="col-sm-9" style="float:left;">

<div class="jumbotron text-center">

<h1>Clinic Management System</h1>

<p>Record of Clinic</p>

</div>

<div class="col-sm-3">

<h4>No. of Patient </h4>

<button type="button" class="btn btn-success"> <?php

echo date('Y-m-d');

?><span class="badge"><?php echo($rowcount);?></span>

</button>

</div>

<div class="col-sm-3">

<h4>No. of Patient </h4>

<button type="button" class="btn btn-danger">MONTH <?php

echo date('Y-m');

?> <span class="badge"><?php echo($rowcount2);?></span>

</button>

</div>

<div class="col-sm-3">

<h4>No. of Patient </h4>

<button type="button" class="btn btn-primary" style="size:+2;">YEAR <?php

echo date('Y');

?><span class="badge"><?php echo($rowcount3);?></span>

</button>

</div>

<div class="col-sm-3">

<h4>No. ofStaff </h4>

<button type="button" class="btn btn-primary" style="size:+2;">YEAR <?php

echo date('Y');

?><span class="badge"><?php echo($rowcount3);?></span>

</button>

</div>

<br>

<br>

<br>

<br>

<br>

</div><!--end of row-->

</div><!--end of container-->

</div>

</div><!--end of row-->

</div>

</div><!-- end of container-->

<div class="container">

<section class="my-modal-box">

<div class="container">

<div class="row">

<div class="col-md-12">

<button type="button" class="btn btn-success btn-lg" data-toggle="modal" data-target="#pops-modal">Click me Mail\_ids <?php echo $rowcount?></button>

<div id="pops-modal" class="modal fade" role="dialog">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header">

<button class="close" data-dismiss="modal">&times;</button>

<h3 class="modal-title"> My First pops Layout</h3>

</div><!-- //modal-header-->

<div class="modal-body">

<section class="row">

<div class="container">

<div class="col-md-6">

<?php foreach($array as $mail) { ?>

<table border="2" class=" table table-responsive table-hover">

<tr>

<th>ID</th>

<th>Name</th>

<th>Email</th>

</tr>

<tr>

<td><?php echo $mail['id'] ?></td>

<td><?php echo $mail['first']?></td>

<td><?php echo $mail['email'] ?></td>

</tr>

</table><?php }?>

</div>

</div>

</section>

</div><!-- //modal-body-->

<div class="modal-footer">

<button class="btn btn-default btn-lg" data-dismiss="modal">Close Me</button>

</div><!-- //modal-footer-->

</div><!-- //modal-content-->

</div><!-- //modal-dialog-->

</div><!-- //pops-modal-->

</div>

</div>

</div>

</section>

</div><!--modal1-->

<div class="container">

<section class="my-modal-box">

<div class="container">

<div class="row">

<div class="col-md-12">

<button type="button" class="btn btn-success btn-lg" data-toggle="modal" data-target="#pops-modal">Click me Mail\_ids <?php echo $rowcount?></button>

<div id="pops-modal" class="modal fade" role="dialog">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header">

<button class="close" data-dismiss="modal">&times;</button>

<h3 class="modal-title">New Patient Details</h3>

</div><!-- //modal-header-->

<div class="modal-body">

<section class="row">

<div class="container">

<div class="col-md-6">

<?php foreach($array1 as $new) { ?>

<table border="2" class=" table table-responsive table-hover">

<tr>

<th>ID</th>

<th>Name</th>

<th>Email</th>

</tr>

<tr>

<td><?php echo $new['id'] ?></td>

<td><?php echo $new['first']?></td>

<td><?php echo $new['email'] ?></td>

</tr>

</table><?php }?>

</div>

</div>

</section>

</div><!-- //modal-body-->

<div class="modal-footer">

<button class="btn btn-default btn-lg" data-dismiss="modal">Close Me</button>

</div><!-- //modal-footer-->

</div><!-- //modal-content-->

</div><!-- //modal-dialog-->

</div><!-- //pops-modal-->

</div>

</div>

</div>

</section>

</div><!--modal2-->

</body>

</html>

**6.11.2: Controller Of Management:Manage.php**

**<?php**

class Manage extends CI\_controller{

public function \_\_construct(){

parent::\_\_construct();

$this->load->model('Database\_model');

$this->load->library('javascript');

if($this->session->userdata('user\_id')===NULL){

redirect("/login/");

}

}

public function cmanage(){

$data=array();

$data['title']="Clinic Management";

$table="new";

$table2="review";

$where4="rev\_date=CURDATE( )";

$where=1;

$where1="DATE =CURDATE( )" ;

$where2="DATE>=(CURDATE()-INTERVAL 1 MONTH)";

$where3="DATE =YEAR(CURDATE( ))";

//$=$currentdate;

//$limit="$offset,$count";

$array=$this->Database\_model->getrows\_d("new",$where);

$array1=$this->Database\_model->getrows\_d("new",$where1);

$array2=$this->Database\_model->getrows\_d("new",$where2);

$array3=$this->Database\_model->getrows\_d("new",$where3);

$array4=$this->Database\_model->getrows\_d("review",$where4);

$array5=$this->Database\_model->getrows\_d("review",$where);

$array6=$this->Database\_model->getrows\_d("new",$where);

$rowcount1=$this->Database\_model->getcount("new",$where1);

$rowcount=$this->Database\_model->getcount("new",$where);

$rowcount2=$this->Database\_model->getcount("new",$where2);

$rowcount3=$this->Database\_model->getcount("new",$where3);

$rowcount4=$this->Database\_model->getcount("review",$where4);

$rowcount5=$this->Database\_model->getcount("review",$where);

$rowcount6=$this->Database\_model->getcountmail("new",$where);

//$where="id=$id";

//$pages=ceil($rowcount/$count);

//$data['pages']=$pages;

//$this->load->library('paging');

$query="";

//$pagefilters['query']=$query;

if($this->input->get('query')!=NULL){

$query=trim($this->input->get('query'));

$where.=" and (contact ='$query' or id ='$query')";

}

//$data['rowcount']=$rowcount;

$data['array']=$array;

$data['array1']=$array1;

$data['array2']=$array2;

$data['array3']=$array3;

$data['array4']=$array4;

$data['array5']=$array5;

$data['array6']=$array6;

$data['rowcount']=$rowcount;

$data['rowcount1']=$rowcount1;

$data['rowcount2']=$rowcount2;

$data['rowcount3']=$rowcount3;

$data['rowcount4']=$rowcount4;

$data['rowcount5']=$rowcount5;

$data['rowcount6']=$rowcount6;

//$data['pagefilters']=$pagefilters;

$url=base\_url()."index.php/manage/cmanage/";

//$paging=$this->paging->createpaging($url,$page,$pagefilters);

//$data['paging']=$paging;

$this->load->view('header',$data);

$this->load->view('nav');

$this->load->view('cmanage',$data);

//$this->load->view('mail\_list',$data);

$this->load->view('footer');

}

public function patientlist(){

$data=array();

$data['title']="Patient-List";

$count=8;

if($this->uri->segment('4')){ $page=$this->uri->segment('4'); }

else{

$page=1;

}

$offset=($page-1)\*$count;

$limit="$offset,$count";

$pagefilters=array();

$table="new";

$user\_id=$\_SESSION['user\_id'];

$where="user\_id=$user\_id";

$query="";

$pagefilters['query']=$query;

if($this->input->get('query')!=NULL){

$query=trim($this->input->get('query'));

$where.=" and (contact ='$query' or id ='$query' or first ='$query' or date ='$query')";

}

$array=$this->Database\_model->getrows($table,$where,$limit);

$rowcount=$this->Database\_model->getcount($table,$where);

$pages=ceil($rowcount/$count);

$data['array']=$array;

$data['query']=$query;

$data['pages']=$pages;

$data['pagefilters']=$pagefilters;

$this->load->library('paging');

$url=base\_url()."index.php/manage/patientlist/";

$paging=$this->paging->createpaging($url,$pages,$pagefilters);

$data['paging']=$paging;

$this->load->view('header',$data);

$this->load->view('nav');

$this->load->view('patientlist',$data);

$this->load->view('footer');

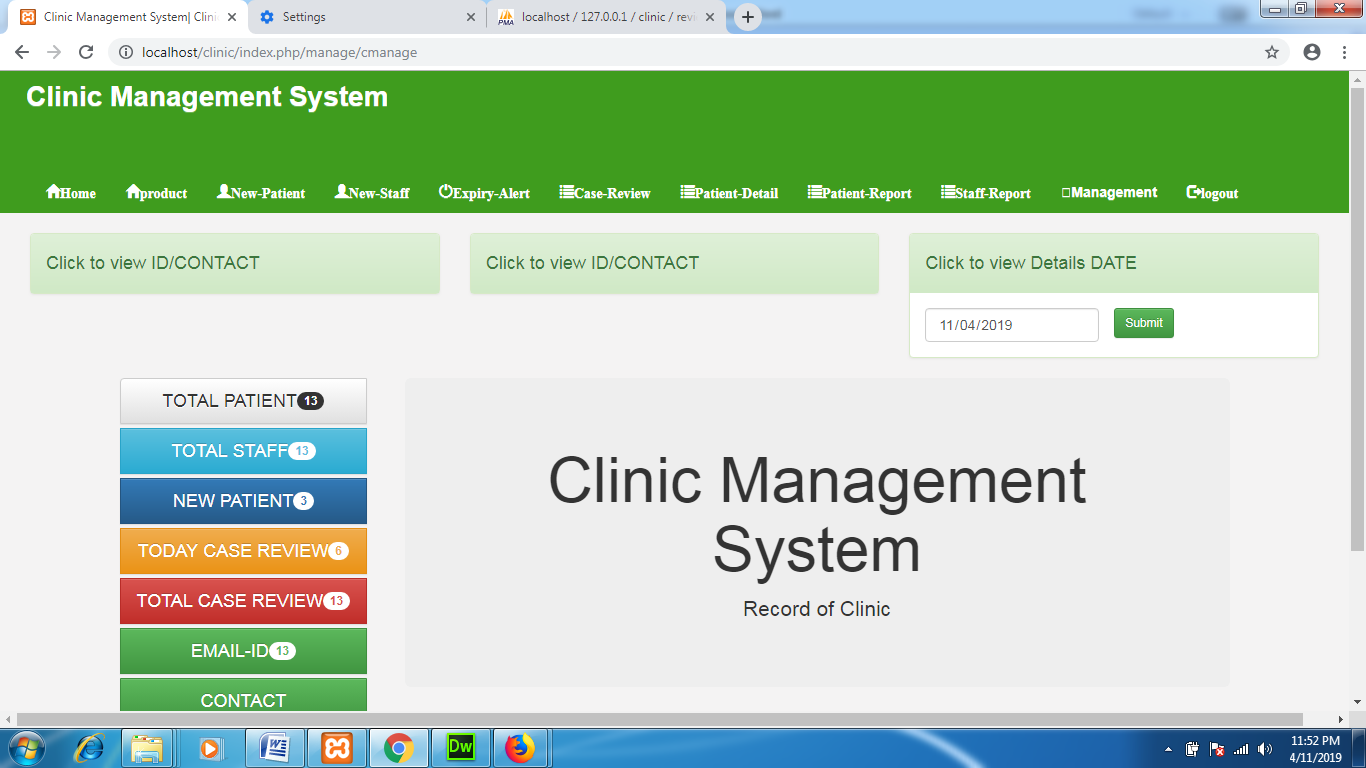
}

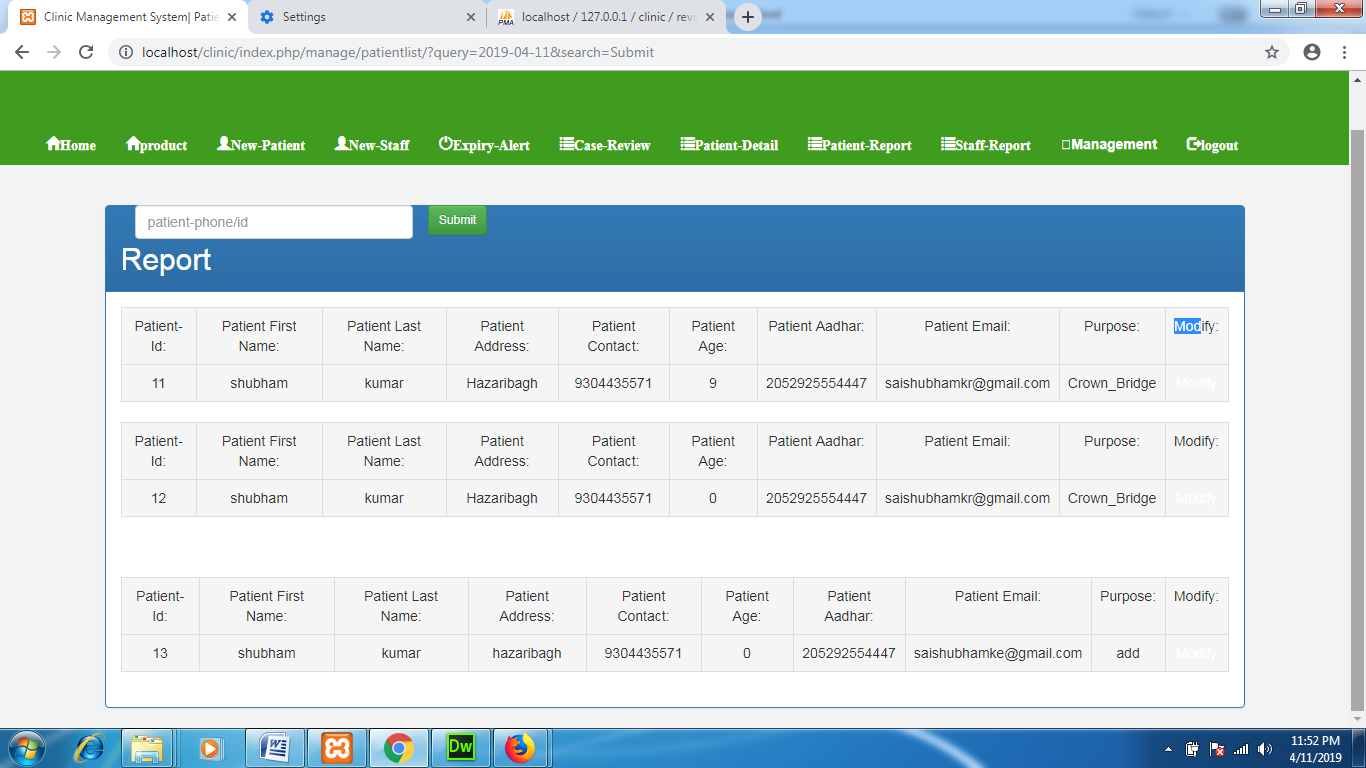
}

?>

#### 6.11.3. View Details DATE

It will Fetch the records of the Patient according to the submitted date.



Searched Report based on date

**CHAPTER 7**

**TEST PLAN**

Software testing is an activity performs to uncover the error and it is a critical element of software quality assurance and represents the ultimate review of software specification, design and coding. Hence the importance of software testing and its implications with respect to software quality can’t be overemphasized. It is used to detect errors. Testing is a dynamic method for verification and validation, where the system to be tested is executed and the behavior of the system is observed

**7.1 Objectives of Testing**

* Minimum cost in the maintenance because error may be reduces in testing.
* Minimum chance of failure.
* Testing is a process of executing a program with the intent of finding an error.
* A good test case is one that has a high probability of finding an as-yet-

undiscovered error.

**7.2 Types of Testing**

**7.2.1 Unit Testing**

Unit testing focuses verification effort on the smallest unit of software design module. Using the detailed description as a guide, important control paths are tested to uncover errors within the boundary of the module. The various modules that have undergone types of input data were given to ensure the consistency of the module and to ensure that information properly flows into and out of the program unit under test. The database integrity is validated during all steps in the execution.

**7.2.2 Integration Testing**

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit-tested modules and build a program structure that has been detected by design. There are two types of integration testing namely, **Top-Down Integration** and **Bottom-Up Integration**. For the system a combination of both the types known as **Sandwich Testing** has been used. The Top-Down Integration was used during the integration of the various Masters and the Bottom-up Integration method was used during the integration of the various modules of the process generation and the report generation.

**7.2.3 Validation Testing**

Once the software was completely assembled as a package; interface errors uncovered and corrected, final series of software tests for validity of the software was carried out. Module interfaces (Graphical User Interfaces developed in a Visual Environment) were also tested for proper working of the windows, pull-down menus, mouse operations, data entry using these interfaces, and the report formats. At the end of this testing phase, it was ensured that the function or performance characteristics of the software conform to the specifications provided by the management and hence are accepted.

A series of these acceptance tests of the system were carried out at the place where this software system was developed (Alpha Testing) in a natural setting and in a controlled environment. The client was called over for testing the software and “by looking over the shoulder” of the client/user of the system, all the errors and usage problems were recorded and corrected. Later this software application was put on live testing (Beta Testing) for the client in its original environment. (i.e., at the office of the company).During this phase, it was ensured that all elements of the software configuration have been properly developed, are catalogued, and have the necessary detail to support the maintenance phase of the software life cycle (Configuration Review).

**7.2.4 Black Box Testing**

Black box testing focuses on the functional requirements of the software. That is, Black Box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program. Black Box Testing is not an alternative to white-box techniques. Rather, it is a complementary approach that is likely to uncover a different class of errors than white-box methods Black-Box Testing attempts to find errors in the following categories:

* Incorrect or missing functions.
* Interface errors.
* Errors in data structures or external data base access.
* Performance errors.
* Initialization and termination errors.

Unlike White Box Testing, which is performed early in the testing process, Black Box Testing tends to be applied during later stages of testing. Because Black Box Testing purposely disregards control structure, attention is focused on the information domain

**7.2.5 White Box Testing**

White Box Testing knowing the internal workings of a product tests can be conducted to ensure that internal operations are performed according to specifications and all internal components have been adequately exercised.Using white box testing methods the test cases that can derived are:

* All independent paths with in a module have been exercised at least once.
* Exercise all logical decisions on their true and false sides.
* Execute all loops at their boundaries and within their operational bounds.
* Exercise internal data structures to ensure their validity.

**CHAPTER 8**

**MAINTAINANCE**

**7.1 Need of Maintenance**

For correct fact

* Improve the quality of project.
* User satisfaction.
* To add new feature.

**7.2 Types of Maintenance**

There are four categories of maintenance: corrective, adaptive, perfective & preventive.

**7.2.1 Corrective Maintainance**

Reactive modification of a software product performed after delivery to correct discovered problems.

**7.2.2 Adaptive Maintainance**

Modification of a software product performed after delivery to keep a software product usable in a changed or changing environment.

**7.2.3 Perfective Maintainance**

Modification of a software product after delivery to improve performance or maintainability.

**7.2.4 Preventive Maintainance**

Modification of a software product after delivery to detect and correct latent faults in the software product before them become effective faults.

**CHAPTER 9**

**CONCLUSION**

In final project the scope of the project expanded as much as needed. In addition to providing all necessary features of a browser we will also try to add some new features to this so that we can make it more preferable to others. In spite of having a short period of time and a challenging environment to cope with, we are concentrating on building a efficient and preferable proxy server application, that can suitably meet its requirements, what it meant for. Now a day’s all Clinics and Health care centers are moving towards online software management solution.

**BIBLIOGRAPHY**

* Executive editors: Alain Abran, James W. Moore; editors Pierre Bourque, Robert Dupuis, ed (March 2005). ["Chapter 2: Software Requirements"](http://www.computer.org/portal/web/swebok/html/ch2). [*Guide to the software engineering body of knowledge*](http://www.swebok.org) (2004 ed.). Los Alamitos, CA: IEEE Computer Society Press. [ISBN](http://en.wikipedia.org/wiki/International_Standard_Book_Number) [0-7695-2330-7](http://en.wikipedia.org/wiki/Special:BookSources/0-7695-2330-7). Retrieved 2007-02-08. "It is widely acknowledged within the software industry that software engineering projects are critically vulnerable when these activities are performed poorly."
* Wiegers, Karl E. (2003). [*Software Requirements*](http://www.processimpact.com) (2nd ed.). Redmond, WA: Microsoft Press. [ISBN](http://en.wikipedia.org/wiki/International_Standard_Book_Number) [0-7356-1879-8](http://en.wikipedia.org/wiki/Special:BookSources/0-7356-1879-8).
* Phillip A. Laplante (2007) *What Every Engineer Should Know about Software Engineering*. Page 44.[**^**](http://en.wikipedia.org/wiki/Systems_analysis#cite_ref-0)[SYSTEMS ANALYSIS](http://web.archive.org/web/20070822025602/http:/pespmc1.vub.ac.be/ASC/SYSTEM_ANALY.html)
* Requirements Engineering A good practice guide, Ian Sommerville and Pete Sawyer, John Wiley and Sons, 1997
* Company, People's Computer (1987). ["Dr. Dobb's journal of software tools for the professional programmer"](http://books.google.com/?id=7RoIAAAAIAAJ). *Dr. Dobb's journal of software tools for the professional programmer* (M&T Pub) **12** (1–6): 116.
* Kolawa, Adam; Huizinga, Dorota (2007). [*Automated Defect Prevention: Best Practices in Software Management*](http://www.wiley.com/WileyCDA/WileyTitle/productCd-0470042125.html). Wiley-IEEE Computer Society Press. pp. 41–43.
* Laycock, G. T. (1993) ([PostScript](http://en.wikipedia.org/wiki/PostScript)). [*The Theory and Practice of Specification Based Software Testing*](http://www.mcs.le.ac.uk/people/gtl1/thesis.ps.gz). Dept of Computer Science, Sheffield University, UK. Retrieved 2008-02-13

**References:**

* [Systems Analysis, Modelling and Prediction (SAMP), University of Oxford](http://www.eng.ox.ac.uk/samp)
* [Software Requirement Analysis using UML](http://www.slideshare.net/dhirajmusings/software-requirement-analysis-using-uml) article by Dhiraj Shetty.
* [*Introduction to Social Macrodynamics*](http://urss.ru/cgi-bin/db.pl?cp=&page=Book&id=34250&lang=en&blang=en&list=Found)
* [A useful set of guides and a case study about the practical application of business and systems analysis methods](http://www.cilco.co.uk/briefing-studies/index.html)
* [Complete online tutorial for system analysis and design](http://www.systemsanalysis.co.nr)
* [A comprehensive description of the discipline of systems analysis from Simmons College, Boston, MA, USA (www.simmons.edu)](http://web.simmons.edu/%7Ebenoit/LIS486/SystemsAnalysis.html)
* [Systems Analysis: An Executive Summary (www.systemsanalyst.com)](http://www.systemsanalyst.com/systems-analyst-executive-summary/)
* <http://en.wikipedia.org/wiki/Requirements_elicitation>
* <http://en.wikipedia.org/wiki/Requirements_analysis>