

















<?php

namespace Mini\Model;

use Mini\Core\Model;

use PDO;

class Register extends Model

{

public function checkEmail($email){

$query = $this->db -> prepare("SELECT email FROM customer\_tbl WHERE email = ? LIMIT 1");

$query -> execute(array($email));

if($query->rowCount()>0){

return true;

}else{

return false;

}

}

public function registerCustomer($name,$email,$phoneNumber,$address,$password){

if($this->checkEmail($email)){

$this->msg->error('User with that email already exist.', $\_SERVER['HTTP\_REFERER']);

}else{

$password = sha1($password);

$queryInsert = $this->db->prepare("INSERT INTO customer\_tbl (id,name,email,phone,address,password,created\_at) VALUES(?,?,?,?,?,?,now())");

$queryInsert->execute(array('',$name,$email,$phoneNumber,$address,$password));

$userId = $this->db->lastInsertId();

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

//print\_r($this->db->errorInfo());

if($queryInsert){

$this->msg->success('You are registered and logged in.', URL.'home/book');

}else{

$this->msg->error('Unable to register customer at this time, please try again later.', $\_SERVER['HTTP\_REFERER']);

}

}

}

public function customerLogIn($email,$password){

$password = sha1($password);

$query = $this->db->prepare("SELECT \* FROM customer\_tbl WHERE email=? AND password=? LIMIT 1");

$query -> execute(array($email,$password));

if ($query->rowCount()>0){

$result = $query->fetch(PDO::FETCH\_ASSOC);

$email= $result["email"];

$userId = $result["id"];

$passwordDb = $result["password"];

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

return $this->msg->success('You are logged in.', URL.'home/book');

}else{

return $this->msg->error('Invalid email and password combination.', $\_SERVER['HTTP\_REFERER']);

}

}

public function getCustomerInfo($id){

$query = $this->db -> prepare("SELECT \* FROM customer\_tbl WHERE id = ? LIMIT 1");

$query -> execute(array($id));

$result = $query->fetch();

return $result;

}

public function getServiceInfo($id){

$query = $this->db -> prepare("SELECT \* FROM service\_tbl WHERE id = ? LIMIT 1");

$query -> execute(array($id));

$result = $query->fetch();

return $result;

}

public function bookedRecord(){

$query = $this->db -> prepare("SELECT \* FROM booking\_tbl WHERE customer\_id=? ORDER BY id DESC");

$query->execute(array($\_SESSION['userId']));

if($query->rowCount()>0){

?>

<h2 class=""> Your Booked Record</h2>

<div class="table-responsive">

<table class="table table-hover table-striped">

<thead>

<tr>

<th>Booking ID</th>

<th>Customer</th>

<th>Room Type</th>

<th># of rooms</th>

<th>Arrival</th>

<th>Departure</th>

<th>Days</th>

<th>Amount Payable</th>

<th>Status</th>

<th>Created at</th>

</tr>

</thead>

<?php

while($row = $query->fetch(PDO::FETCH\_ASSOC)){

?>

<tr class="text-left">

<td> <?php echo $row['booking\_id'];?> </td>

<td><?php echo $this->getCustomerInfo($row['customer\_id'])->name;?></td>

<td><?php echo $this->getServiceInfo($row['service\_id'])->room\_name;?></td>

<td> <?php echo $row['number\_of\_room'];?> </td>

<td> <?php echo $row['arrival'];?> </td>

<td> <?php echo $row['departure'];?> </td>

<td> <?php echo $row['number\_of\_days'];?> </td>

<td> <?php echo $row['amount\_payable'];?> </td>

<td>

<input type="hidden" name="bookingId" value="<?php echo $row['id'];?>">

<div class="form-group">

<?php //if($row['status'] === 'Pending'){

echo '<span class="badge">'. $row['status'].'</span>';

//}

?>

</div>

</td>

<td><?php echo $row['created\_at'];?></td>

</tr>

<?php

}

?>

</table>

</div>

<?php

}else{

echo '<div>

<a class="list-group-item">No Record Found.</a>

</div>';

}

}

public function saveBooking($bookingId,$customerId,$serviceId,$numberOfRoom,$arrival,$departure,$amountPayable,$numberOfDays,$status)

{

// Update number of roomnbooked

$oldBookedVal = $this->getServiceInfo($serviceId)->number\_of\_room\_booked; //get the old val

$newlyBookedVal = $numberOfRoom + $oldBookedVal; //add with th ene val

$this->updatenumberOfRoomsBooked($serviceId,$newlyBookedVal); //call update functon

// die($bookingId.'customerid:'.$customerId.$serviceId.$numberOfRoom.$arrival.$departure.'amount:'.$amountPayable.$numberOfDays.$status);

$queryInsert = $this->db->prepare("INSERT INTO booking\_tbl (id,booking\_id,customer\_id,service\_id,number\_of\_room,arrival,departure,amount\_payable,number\_of\_days,status,created\_at) VALUES(?,?,?,?,?,?,?,?,?,?,now())");

$queryInsert->execute(array('',$bookingId,$customerId,$serviceId,$numberOfRoom,$arrival,$departure,$amountPayable,$numberOfDays,$status));

unset($\_SESSION['serviceId']);

unset($\_SESSION['numberOfRooms']);

unset($\_SESSION['arrival']);

unset($\_SESSION['departure']);

unset($\_SESSION['amountPayable']);

unset($\_SESSION['days']);

print\_r($this->db->errorInfo());

if($queryInsert){

$this->msg->success('Booking successful made.', URL);

}else{

$this->msg->success('Unable to book at this time, please try again later.', $\_SERVER['HTTP\_REFERER']);

}

}

public function updatenumberOfRoomsBooked($serviceId,$newlyBookedVal)

{

$queryUpdate = $this->db->prepare("UPDATE service\_tbl set number\_of\_room\_booked = ? WHERE id = ?");

$queryUpdate->execute(array($newlyBookedVal, $serviceId));

if($queryUpdate){

// $this->msg->success('Booking status successfully updated.', $\_SERVER['HTTP\_REFERER']);

}else{

$this->msg->error('Unable to update booking status at this time, please try again later.', $\_SERVER['HTTP\_REFERER']);

}

}

public function listServices(){

$query = $this->db->prepare("SELECT \* FROM service\_tbl ");

$query->execute();

if($query->rowCount()>0){

while($row=$query->fetch(PDO::FETCH\_ASSOC)){

$id = $row['id'];

?>

<div class="media list-group-item">

<div class="media-left">

<a href="#">

<img class="media-object" height="100" width="100" src="<?php echo URL.'images/'.$row['image']?>" alt="Image">

</a>

</div>

<div class="media-body">

<h4 class="media-heading"><?php echo $row['room\_name']; ?></h4>

<form action="<?php $\_SERVER['PHP\_SELF'];?>" method="POST">

<div class="row">

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

<?php echo $row['description']; ?>

</div>

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

Number Of Rooms

<input type="number" style="width:80px;" name="numberOfRooms" value="1">

<input type="hidden" name="serviceId" value="<?php echo $row['id']; ?>">

<input type="hidden" name="price" value=" <?php echo $row['price']; ?>">

</div>

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

Per Night<br>

<span class="text-primary">&#8358;<?php echo $row['price']; ?> </span><br>

<button class="btn btn-primary" name="bookBtn" type="submit">Book now</button>

</div>

</div>

</form>

</div>

</div>

<?php

}

}else{

echo '

<a class="list-group-item">Not reframed yet. </a>

';

}

}

}

public function adminLogIn($email,$password){

if(!empty($email) or !empty($password)){

$query = $this->db->prepare("SELECT \* FROM admin\_tbl WHERE email=? LIMIT 1");

$query -> execute(array($email));

if ($query->rowCount()>0){

$result = $query->fetch(PDO::FETCH\_ASSOC);

$email= $result["email"];

$adminId = $result["id"];

$adminId = $result["id"];

$passwordDb = $result["password"];

$verify = password\_verify($password,$passwordDb);

if ($verify){

$\_SESSION["adminId"] = $adminId;

header('location:'.URL.'register/member');

}else{

return $this->msg->error('Invalid username and password combination.', URL);

}

}else{

return $this->msg->error('Invalid username and password combination.', URL);

}

}else{

return $this->msg->error('Invalid username and password combination.', URL);

}

}

<?php

$msg = new \Mini\Libs\FlashMessages();

use Mini\Model\Service;

$Service = (new Service);

?>

<div class="container">

<div class="row">

<div class="col-md-5">

<div class="panel">

<div class="panel-heading">

<?php $msg->display(); ?>

<h2 class="text-primary text-center">Add Room Category </h2>

</div>

<div class="panel-body " >

<form action="<?php $\_SERVER['PHP\_SELF'];?>" method="POST" role="form" enctype="multipart/form-data">

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

<div class="form-group text-left">

<label for="">Room Category Name</label>

<input type="text" name="roomName" class="form-control" id="" placeholder="Enter Room Name">

</div>

</div>

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

<div class="form-group text-left">

<label for="">Price</label>

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

</div>

</div>

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

<div class="form-group text-left">

<label for="">Number Of Rooms</label>

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

</div>

</div>

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

<div class="form-group text-left">

<label for="">Room Image</label>

<input type="file" name="file" class="form-control">

</div>

</div>

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

<div class="form-group text-left">

<label for=""> Description</label>

<textarea name="description" id="input" class="form-control" ></textarea>

</div>

</div>

<button type="submit" name="addServiceBtn" class="btn btn-primary ">Add Service</button>

</form>

</div>

</div>

</div>

<div class="col-md-7">

<div class="row">

<div class="panel panel-body ">

<?php

$Service->serviceRecord();

?>

</div>

</div>

</div>

</div>

</div>