controllers/user.php

<?php

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

class User extends CI\_Controller {

public function \_\_construct() {

parent::\_\_construct();

$this->load->model('userModel');

$this->load->model('hotelModel');

$this->load->model('bookingModel');

}

public function login() {

if ($this->input->get\_post('btn\_Login')) {

if ($this->loginValid()) {

if ($this->input->get\_post('username') == 'admin' && $this->input->get\_post('password') == 'admin') {

redirect('admin/inputHotel', 'refresh');

} else {

if ($this->userModel->checkUserExist($this->input->get\_post('username'), $this->input->get\_post('password')) > 0) {

$this->userModel->setLogged($this->input->get\_post('username'));

if ($this->input->get\_post('rememberMe')) {

set\_cookie('username', $this->input->get\_post('username'), 86400\*20);

set\_cookie('password', $this->input->get\_post('password'), 86400\*20);

} else {

delete\_cookie('username');

delete\_cookie('password');

}

redirect('user/searchHotel', 'refresh');

}

}

} else {

$this->session->set\_flashdata('errorLogin', validation\_errors());

redirect('user/login');

}

}

$this->load->view('templates/header');

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

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

}

function loginValid() {

$this->form\_validation->set\_rules(

'username',

'Username',

'required|callback\_checkUsername'

);

$this->form\_validation->set\_rules(

'password',

'Password',

'required|callback\_checkPassword'

);

if ($this->form\_validation->run()) {

return true;

} else {

return false;

}

}

function checkUsername($username) {

if ($this->userModel->checkUsernameExist($username) > 0 || $username == 'admin') {

return true;

} else {

$this->form\_validation->set\_message('checkUsername', 'Username not registered yet!');

return false;

}

}

function checkPassword($password) {

if ($this->userModel->checkPasswordExist($password) > 0 || $password == 'admin') {

return true;

} else {

$this->form\_validation->set\_message('checkPassword', 'Password not registered yet!');

return false;

}

}

public function register() {

$data['success']= '';

if ($this->input->get\_post('btn\_Register')) {

if ($this->registerValid()) {

$new= [

'name' => $this->input->get\_post('name'),

'username' => $this->input->get\_post('username'),

'email' => $this->input->get\_post('email'),

'phone' => $this->input->get\_post('phone'),

'role' => $this->input->get\_post('role'),

'password' => $this->input->get\_post('password')

];

if ($this->userModel->insert($new) > 0) {

$data['success']= 'Success registered!';

} else {

$data['success']= 'Register failed!';

}

} else {

$this->session->set\_flashdata('errorRegister', validation\_errors());

redirect('user/register');

}

}

$this->load->view('templates/header');

$this->load->view('user/register', $data);

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

}

function registerValid() {

$this->form\_validation->set\_rules(

'name',

'Name',

'required|alpha'

);

$this->form\_validation->set\_rules(

'username',

'Username',

'required|callback\_checkNewUsername'

);

$this->form\_validation->set\_rules(

'email',

'E-mail',

'required|callback\_checkNewEmail'

);

$this->form\_validation->set\_rules(

'phone',

'Phone Number',

'required|callback\_checkNewPhoneNumber'

);

$this->form\_validation->set\_rules(

'password',

'Password',

'required|alpha\_numeric|min\_length[6]|max\_length[10]|matches[cpassword]'

);

$this->form\_validation->set\_rules(

'cpassword',

'Confirm - Password',

'required|matches[password]'

);

if ($this->form\_validation->run()) {

return true;

} else {

return false;

}

}

function checkNewUsername($username) {

if ($this->userModel->checkUsernameExist($username) == 0) {

return true;

} else {

$this->form\_validation->set\_message('checkNewUsername', 'Username already used!');

return false;

}

}

function checkNewEmail($email) {

if ($this->userModel->checkEmailExist($email) == 0) {

return true;

} else {

$this->form\_validation->set\_message('checkNewEmail', 'E-mail already used!');

return false;

}

}

function checkNewPhoneNumber($phone) {

if ($this->userModel->checkPhoneNumberExist($phone) == 0) {

return true;

} else {

$this->form\_validation->set\_message('checkNewPhoneNumber', 'Phone number already used!');

return false;

}

}

public function searchHotel($state= '') {

$data['logged']= $this->userModel->getLogged()[0];

$data['hotels']= $this->hotelModel->getIsActiveHotels();

if ($state == 'booked') $data['state']= 'Booking berhasil!';

else if ($state == 'added') $data['state']= 'Tambah ke cart!';

else if ($state == 'checkout') $data['state']= 'Check out berhasil!';

$this->load->view('templates/header');

$this->load->view('user/searchHotel', $data);

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

}

public function bookingHotel($id= '') {

$data['rooms']= $this->hotelModel->getAllRoom();

if ($id != '') {

$this->session->set\_userdata('id\_hotel', $id);

redirect('user/bookingHotel');

} else {

$id= $this->session->userdata('id\_hotel');

}

if ($this->input->get\_post('btn\_Next')) {

$booking['logged']= $this->userModel->getLogged()[0];

$booking['header']= [

'id\_hotel' => $id,

'hotel' => $this->hotelModel->getHotel($id)[0]->name,

'amount' => $this->input->get\_post('amount'),

'orderDate' => $this->input->get\_post('orderDate'),

'id\_room' => $this->input->get\_post('id\_room'),

'roomType' => $this->hotelModel->getRoom($this->input->get\_post('id\_room'))[0]->type,

'total' => $this->hotelModel->getHotelPrice($id)[0]->price

];

$this->session->set\_userdata('booking', $booking);

redirect('user/inputBookingData', 'refresh');

}

$this->load->view('templates/header');

$this->load->view('user/booking/bookingHotel', $data);

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

}

public function inputBookingData() {

$success= false;

if (!is\_null($this->session->userdata('booking'))) {

$data['booking']= $this->session->userdata('booking');

if ($this->input->get\_post('btn\_Confirm')) {

$data['booking']['logged']->name= $this->input->get\_post('name1');

$data['booking']['logged']->email= $this->input->get\_post('email1');

$data['booking']['logged']->phone= $this->input->get\_post('phone1');

if ($this->validBooking($data['booking']['header']['amount'])) {

$data['booking']['detail']['customer']= $this->getCustomerDataField($data['booking']['header']['amount']);

$this->session->set\_userdata('booking', $data['booking']);

$success= true;

} else {

$this->session->set\_flashdata('errorBooking', validation\_errors());

redirect('user/inputBookingData');

}

}

if ($this->input->get\_post('btn\_AddToCart')) {

if (!is\_null($this->session->userdata('carts'))) {

$carts= $this->session->userdata('carts');

}

$username= $data['booking']['logged']->username;

$header= $data['booking']['header'];

$detail= $data['booking']['detail'];

$carts[$username][]= [

'header' => $header,

'detail' => $detail

];

$this->session->set\_userdata('carts', $carts);

$state= 'added';

redirect('user/searchHotel/'.$state);

}

}

$this->load->view('templates/header');

if ($success) {

$this->load->view('user/booking/bookingDetail', $data);

} else {

$this->load->view('user/booking/inputBookingData', $data);

}

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

}

function validBooking($amount) {

for ($i= 0; $i<$amount; $i++) {

$this->form\_validation->set\_rules(

'name'.($i+1),

'Name',

'required'

);

$this->form\_validation->set\_rules(

'phone'.($i+1),

'No. HP',

'required|numeric'

);

$this->form\_validation->set\_rules(

'email'.($i+1),

'E-mail',

'required'

);

}

if ($this->form\_validation->run()) {

return true;

} else {

return false;

}

}

function getCustomerDataField($amount) {

for ($i= 0; $i<$amount; $i++) {

$customer[($i+1)]= [

'name'.($i+1) => $this->input->get\_post('name'.($i+1)),

'phone'.($i+1) => $this->input->get\_post('phone'.($i+1)),

'email'.($i+1) => $this->input->get\_post('email'.($i+1)),

'breakfast'.($i+1) => $this->input->get\_post('breakfast'.($i+1)),

];

}

return $customer;

}

public function shoppingCart($removed= '') {

$username= $this->userModel->getLogged()[0]->username;

$data['carts']= '';

if (!is\_null($this->session->userdata('carts')) && array\_key\_exists($username, $this->session->userdata('carts'))) {

$data['carts']= $this->session->userdata('carts')[$username];

if ($removed != '') {

unset($data['carts'][$removed]);

$data['carts']= array\_values($data['carts']);

$carts= $this->session->userdata('carts');

$carts[$username]= $data['carts'];

$this->session->set\_userdata('carts', $carts);

redirect('user/shoppingCart');

}

if ($this->input->get\_post('btn\_CheckOut')) {

$insertHeader= false;

$insertDetail= false;

foreach ($data['carts'] as $key => $i) {

$header= [

'id' => $this->bookingModel->autoGenID(),

'id\_user' => $this->userModel->getLogged()[0]->id,

'id\_hotel' => $i['header']['id\_hotel'],

'id\_room' => $i['header']['id\_room'],

'order\_date' => $i['header']['orderDate'],

'total' => $i['header']['total']

];

if ($this->bookingModel->insertHeader($header) > 0) {

$insertHeader= true;

}

foreach ($i['detail']['customer'] as $key => $j) {

$detail= [

'id\_booking' => $header['id'],

'name' => $j['name'.$key],

'phone' => $j['phone'.$key],

'email' => $j['email'.$key],

'breakfast' => $j['breakfast'.$key]

];

if ($this->bookingModel->insertDetail($detail) > 0) {

$insertDetail= true;

}

}

}

if ($insertHeader && $insertDetail) {

$carts= $this->session->userdata('carts');

unset($carts[$username]);

$this->session->set\_userdata('carts', $carts);

$state= 'checkout';

redirect('user/searchHotel/'.$state);

}

}

}

$this->load->view('templates/header');

$this->load->view('user/shoppingCart', $data);

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

}

public function bookingHistory() {

$data['history']['header']= $this->bookingModel->getAllBookingHistoryOf($this->userModel->getLogged()[0]->id);

foreach ($data['history']['header'] as $key => $row) {

$data['history']['detail'][$row->id]= $this->bookingModel->getBookingDetail($row->id);

}

$this->load->view('templates/header');

$this->load->view('user/booking/bookingHistory', $data);

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

}

public function profile() {

$data['logged']= $this->userModel->getLogged()[0];

$this->load->view('templates/header');

$this->load->view('user/profile', $data);

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

}

}

models/userModel.php

<?php

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

class UserModel extends CI\_Model {

public function insert($new) {

$this->db->query(

"INSERT INTO user (name, username, email, phone, role, password)

VALUES ('$new[name]', '$new[username]', '$new[email]', $new[phone], '$new[role]', '$new[password]')"

);

return $this->db->affected\_rows();

}

public function checkUserExist($username, $password) {

$this->db->query(

"SELECT username

FROM user

WHERE username = '$username' AND password = '$password'"

);

return $this->db->affected\_rows();

}

public function checkUsernameExist($username) {

$this->db->query(

"SELECT username

FROM user

WHERE username = '$username'"

);

return $this->db->affected\_rows();

}

public function checkEmailExist($email) {

$this->db->query(

"SELECT email

FROM user

WHERE email = '$email'"

);

return $this->db->affected\_rows();

}

public function checkPhoneNumberExist($phone) {

$this->db->query(

"SELECT phone

FROM user

WHERE phone = '$phone'"

);

return $this->db->affected\_rows();

}

public function checkPasswordExist($password) {

$this->db->query(

"SELECT password

FROM user

WHERE password = '$password'"

);

return $this->db->affected\_rows();

}

public function getLogged() {

$query= $this->db->query(

"SELECT \*

FROM user

WHERE status = 1"

);

return $query->result();

}

public function setLogout($username) {

$this->db->query(

"UPDATE user

SET status = 0

WHERE username = '$username'"

);

}

public function setLogged($username) {

$this->db->query(

"UPDATE user

SET status = 1

WHERE username = '$username'"

);

}

}

views/user/register.php

<div class="container" style="margin-top: 20px;">

<div class="row">

<div class="col s6 offset-s3">

<div class="card center">

<div class="card-title">

<h4>Register</h4>

</div>

<div class="card-content">

<?php

echo $this->session->flashdata('errorRegister');

echo form\_open();

echo 'Name: ';

echo form\_input('name');

echo 'Username: ';

echo form\_input('username');

echo 'E-mail: ';

echo form\_input('email');

echo 'Phone Number: ';

echo form\_input('phone');

echo 'Role: ';

$options= [

'customer' => 'Customer',

'merchant' => 'Merchant'

];

echo form\_dropdown('role', $options, 'customer');

echo 'Password: ';

echo form\_password('password');

echo 'Confirm - Password: ';

echo form\_password('cpassword').'<br> <br>';

echo form\_submit('btn\_Register', 'Register', "class='btn blue darken-3'");

echo form\_close();

echo $success;

?>

</div>

</div>

</div>

</div>

</div>

views/user/login.php

<div class="container" style="margin-top: 20px;">

<div class="row">

<div class="col s6 offset-s3">

<div class="card center">

<div class="card-title">

<h4>Login</h4>

</div>

<div class="card-content">

<?php

echo $this->session->flashdata('errorLogin');

echo form\_open();

echo 'Username: ';

if (!is\_null(get\_cookie('username'))) {

echo form\_input('username', $this->input->cookie('username'));

} else {

echo form\_input('username');

}

echo 'Password: ';

echo form\_password('password');

echo '<br> <br>';

echo '<label style="margin-right: 25px;">';

if (!is\_null(get\_cookie('username')) && !is\_null(get\_cookie('password'))) {

echo form\_checkbox(['name' => 'rememberMe', 'value' => 'accept', 'checked' => true]);

} else {

echo form\_checkbox(['name' => 'rememberMe', 'value' => 'accept']);

}

echo '<span>Remember Me</span>';

echo '</label>';

echo form\_submit('btn\_Login', 'Login', "class='btn blue darken-3'");

echo form\_close();

?>

</div>

</div>

</div>

</div>

</div>

views/user/shoppingCart.php

<div class="container" style="margin-top: 20px;">

<div class="section">

<a href="<?php echo base\_url('index.php/user/searchHotel'); ?>" class="btn blue darken-3">Back</a>

</div>

<?php if (!empty($carts)): ?>

<h5>Your Shopping Cart</h5>

<div class="section">

<table class="striped centered">

<thead>

<tr>

<th>Tanggal Pemesanan</th>

<th>Hotel</th>

<th>Kamar</th>

<th>Harga</th>

<th>Action</th>

</tr>

</thead>

<tbody>

<?php $total= 0; ?>

<?php foreach ($carts as $key => $value): ?>

<tr>

<td><?php echo $value['header']['orderDate']; ?></td>

<td><?php echo $value['header']['hotel']; ?></td>

<td><?php echo $value['header']['roomType']; ?></td>

<td><?php echo 'Rp. '.number\_format($value['header']['total'], 2, ",", "."); ?></td>

<td>

<a href="<?php echo base\_url('index.php/user/shoppingCart/'.$key); ?>" class="btn blue darken-3">Remove</a>

</td>

<?php $total+= $value['header']['total']; ?>

</tr>

<?php endforeach; ?>

</tbody>

<tfoot style="font-size: 16pt; font-weight: bold;">

<tr>

<td colspan="4"><?php echo 'Total: Rp. '.number\_format($total, 2, ",", "."); ?></td>

</tr>

</tfoot>

</table>

<div class="right" style="margin-bottom: 20px;">

<?php

echo form\_open();

echo form\_submit('btn\_CheckOut', 'Check Out', 'class="btn blue darken-3"');

echo form\_close();

?>

</div>

</div>

<?php else: ?>

<h5>Belum ada keranjang.</h5>

<?php endif; ?>

</div>

views/user/inputBookingData.php

<div class="container" style="margin-top: 20px;">

<div class="section">

<a href="<?php echo base\_url('index.php/user/bookingHotel'); ?>" class="btn blue darken-3">Back</a>

</div>

<div class="center">

<?php echo $this->session->flashdata('errorBooking'); ?>

<?php echo form\_open(); ?>

<?php for ($i= 0; $i<$booking['header']['amount']; $i++): ?>

<div class="row">

<div class="col s6 offset-s3">

<div class="card center">

<div class="card-title">

<h4>Customer Data</h4>

</div>

<div class="card-content">

<?php

if ($i == 0) {

echo 'Nama';

echo form\_input(['name' => 'name'.($i+1), 'value' => $booking['logged']->name]);

echo 'No. HP';

echo form\_input(['name' => 'phone'.($i+1), 'value' => $booking['logged']->phone]);

echo 'E-mail';

echo form\_input(['name' => 'email'.($i+1), 'value' => $booking['logged']->email]);

} else {

echo 'Nama';

echo form\_input(['name' => 'name'.($i+1)]);

echo 'No. HP';

echo form\_input(['name' => 'phone'.($i+1)]);

echo 'E-mail';

echo form\_input(['name' => 'email'.($i+1)]);

}

echo 'Breakfast';

$options= [

'Yes' => 'Yes',

'No' => 'No'

];

echo form\_dropdown(

[

'name' => 'breakfast'.($i+1),

'options' => $options,

'selected' => 'yes'

]

);

?>

</div>

</div>

</div>

</div>

<?php endfor; ?>

<?php

echo form\_submit('btn\_Confirm', 'Confirm', 'class="btn blue darken-3"');

echo form\_close().'<br>';

?>

</div>

</div>

views/user/bookingDetail.php

<div class="container" style="margin-top: 20px;">

<div class="row">

<div class="col s6 offset-s3">

<div class="card">

<div class="card-title center">

<h4>Rincian Booking</h4>

</div>

<div class="card-content">

Tanggal Pemesanan:

<div class="right">

<?php echo $booking['header']['orderDate']; ?>

</div> <br>

Hotel:

<div class="right">

<?php echo $booking['header']['hotel']; ?>

</div> <br>

Jenis Kamar:

<div class="right">

<?php echo $booking['header']['roomType']; ?>

</div> <br>

Jumlah Orang:

<div class="right">

<?php echo $booking['header']['amount']; ?>

</div> <br> <br>

Data Tamu <br> <hr>

<?php foreach ($booking['detail']['customer'] as $key => $value): ?>

Nama:

<div class="right">

<?php

echo $value['name'.($key)];

?>

</div> <br>

No. HP:

<div class="right">

<?php

echo $value['phone'.($key)];

?>

</div> <br>

Email:

<div class="right">

<?php

echo $value['email'.($key)];

?>

</div> <br>

Breakfast:

<div class="right">

<?php

echo $value['breakfast'.($key)];

?>

</div> <hr>

<?php endforeach; ?> <br>

Harga Total:

<div class="right">

<?php echo 'Rp. '.number\_format($booking['header']['total'], 2, ",", "."); ?> </div>

</div>

</div>

</div>

</div>

<div class="center">

<?php

echo form\_open();

echo form\_submit('btn\_AddToCart', 'Add to Cart', 'class="btn blue darken-3"');

echo form\_close();

?>

</div> <br>

</div>