|  |  |
| --- | --- |
| REST API pagination project in Codeigniter 3 | |
| Controller | C:\xampp\htdocs\1\_api\pgination\_CI\application\controllers\UsersController.php |
|  | <?php  defined('BASEPATH') OR exit('No direct script access allowed');  class UsersController extends CI\_Controller {      public function \_\_construct() {          parent::\_\_construct();          // Load the model          $this->load->model('UsersModel');      }      // Function to return paginated users for the view      public function viewPaginatedUsers() {          // Get pagination parameters from query string          $limit = $this->input->get('limit', TRUE) ?: 4; // default to 4          $page = $this->input->get('page', TRUE) ?: 1;    // default to page 1          $offset = ($page - 1) \* $limit;          // Get total number of users          $total\_users = $this->UsersModel->getTotalUsersCount();            // Get paginated users          $users = $this->UsersModel->getPaginatedUsers($limit, $offset);          // Prepare pagination metadata          $total\_pages = ceil($total\_users / $limit);          $current\_page = (int)$page;          // Prepare data for the view          $data = array(              'users' => $users,              'current\_page' => $current\_page,              'total\_pages' => $total\_pages          );          // Load the view          $this->load->view('users\_view', $data);      }      // Function to return paginated users as JSON      public function getPaginatedUsers() {          // Get pagination parameters from query string          $limit = $this->input->get('limit', TRUE) ?: 4; // default to 4          $page = $this->input->get('page', TRUE) ?: 1;    // default to page 1          $offset = ($page - 1) \* $limit;          // Get total number of users          $total\_users = $this->UsersModel->getTotalUsersCount();            // Get paginated users          $users = $this->UsersModel->getPaginatedUsers($limit, $offset);          // Prepare pagination metadata          $total\_pages = ceil($total\_users / $limit);          $current\_page = (int)$page;          // Prepare the response          $response = array(              'status' => true,              'total\_users' => $total\_users,              'total\_pages' => $total\_pages,              'current\_page' => $current\_page,              'data' => $users          );          // Output the response in JSON format          $this->output               ->set\_content\_type('application/json')               ->set\_output(json\_encode($response));      }  } |
| View part | C:\xampp\htdocs\1\_api\pgination\_CI\application\views\users\_view.php |
|  | <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Paginated Users</title>      <style>          table {              width: 50%;              margin: 20px auto;              border-collapse: collapse;          }          table, th, td {              border: 1px solid black;          }          th, td {              padding: 8px;              text-align: center;          }      </style>  </head>  <body>      <h2 style="text-align:center;">Paginated Users</h2>      <?php if (!empty($users)) : ?>      <table>          <thead>              <tr>                  <th>User ID</th>                  <th>User Name</th>                  <th>Email</th>              </tr>          </thead>          <tbody>              <?php foreach ($users as $user) : ?>              <tr>                  <td><?= $user['userId'] ?></td>                  <td><?= $user['userName'] ?></td>                  <td><?= $user['password'] ?></td>              </tr>              <?php endforeach; ?>          </tbody>      </table>        <div style="text-align:center;">          <p>Page <?= $current\_page ?> of <?= $total\_pages ?></p>          <!-- Pagination Links -->          <a href="?page=<?= $current\_page - 1 ?>" <?= $current\_page <= 1 ? 'style="pointer-events: none; color: grey;"' : '' ?>>Previous</a>          <a href="?page=<?= $current\_page + 1 ?>" <?= $current\_page >= $total\_pages ? 'style="pointer-events: none; color: grey;"' : '' ?>>Next</a>      </div>      <?php else : ?>      <p style="text-align:center;">No users found.</p>      <?php endif; ?>  </body>  </html> |
| Model | C:\xampp\htdocs\1\_api\pgination\_CI\application\models\UsersModel.php |
|  | <?php  defined('BASEPATH') OR exit('No direct script access allowed');  class UsersModel extends CI\_Model {      public function getPaginatedUsers($limit, $offset) {          // Fetch paginated data          $query = $this->db->select('userId, userName, password')                            ->limit($limit, $offset)                            ->get('lfy\_users');          return $query->result\_array();      }      public function getTotalUsersCount() {          // Get total number of users for pagination          return $this->db->count\_all('lfy\_users');      }  } |
| Route create | $route['api/users'] = 'UsersController/getPaginatedUsers';  $route['api/users/view'] = 'UsersController/viewPaginatedUsers'; |
| Database create | $active\_group = 'default';  $query\_builder = TRUE;  $db['default'] = array(      'dsn'   => '',      'hostname' => 'localhost',      'username' => 'root',      'password' => '',      'database' => 'busbooking',      'dbdriver' => 'mysqli',      'dbprefix' => '',      'pconnect' => FALSE,      'db\_debug' => (ENVIRONMENT !== 'production'),      'cache\_on' => FALSE,      'cachedir' => '',      'char\_set' => 'utf8',      'dbcollat' => 'utf8\_general\_ci',      'swap\_pre' => '',      'encrypt' => FALSE,      'compress' => FALSE,      'stricton' => FALSE,      'failover' => array(),      'save\_queries' => TRUE  ); |
| Call the page |  |
| Create database | -- phpMyAdmin SQL Dump  -- version 5.2.1  -- https://www.phpmyadmin.net/  --  -- Host: 127.0.0.1  -- Generation Time: Sep 11, 2024 at 09:12 AM  -- Server version: 10.4.28-MariaDB  -- PHP Version: 8.0.28  SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";  START TRANSACTION;  SET time\_zone = "+00:00";  /\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;  /\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;  /\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;  /\*!40101 SET NAMES utf8mb4 \*/;  --  -- Database: `busbooking`  --  -- --------------------------------------------------------  --  -- Table structure for table `lfy\_users`  --  CREATE TABLE `lfy\_users` (    `userId` int(11) NOT NULL,    `userName` varchar(255) NOT NULL,    `password` varchar(255) NOT NULL  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;  --  -- Dumping data for table `lfy\_users`  --  INSERT INTO `lfy\_users` (`userId`, `userName`, `password`) VALUES  (1, 'john\_doe', 'password123'),  (2, 'jane\_smith', 'pass1234'),  (3, 'mike\_jones', 'mikepass'),  (4, 'emma\_watson', 'emma456'),  (5, 'will\_smith', 'will\_pass'),  (6, 'julia\_roberts', 'julia789'),  (7, 'tom\_cruise', 'tom987'),  (8, 'kevin\_durant', 'kdpass'),  (9, 'stephen\_curry', 'curry123'),  (10, 'lebron\_james', 'lebron789'),  (11, 'elon\_musk', 'musk123'),  (12, 'bill\_gates', 'billpass'),  (13, 'mark\_zuckerberg', 'mark789'),  (14, 'sundar\_pichai', 'google123'),  (15, 'tim\_cook', 'applepass'),  (16, 'jeff\_bezos', 'amazonpass'),  (17, 'warren\_buffett', 'investor123'),  (18, 'larry\_page', 'search456'),  (19, 'sergey\_brin', 'search789'),  (20, 'jack\_ma', 'alibaba123');  --  -- Indexes for dumped tables  --  --  -- Indexes for table `lfy\_users`  --  ALTER TABLE `lfy\_users`    ADD PRIMARY KEY (`userId`);  --  -- AUTO\_INCREMENT for dumped tables  --  --  -- AUTO\_INCREMENT for table `lfy\_users`  --  ALTER TABLE `lfy\_users`    MODIFY `userId` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=21;  COMMIT;  /\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;  /\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;  /\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/; |
|  |  |
| Call the page | http://localhost/1\_api/pgination\_CI/index.php/api/users |
|  |  |
|  |  |
| Call the view page | http://localhost/1\_api/pgination\_CI/index.php/api/users/view |
|  |  |
| Check through postman | <http://localhost/ci3/index.php/userscontroller/getPaginatedUsers?limit=4&page=1>  <http://localhost/1_api/pgination_CI/index.php/api/users?limit=4&page=1> |
|  |  |
|  |  |
|  | To call a specific user ID, you need to set up a method in your UsersController to handle requests for a specific user. Here's how you can do it: |
| C:\xampp\htdocs\1\_api\pgination\_CI\application\controllers\UsersController.php | public function getUserById($userId) {          // Load the model          $this->load->model('UsersModel');            // Get the user data          $user = $this->UsersModel->getUserById($userId);            // Check if user exists          if ($user) {              // Prepare the response              $response = array(                  'status' => true,                  'data' => $user              );          } else {              // User not found              $response = array(                  'status' => false,                  'message' => 'User not found'              );          }            // Output the response in JSON format          $this->output               ->set\_content\_type('application/json')               ->set\_output(json\_encode($response));      } |
| C:\xampp\htdocs\1\_api\pgination\_CI\application\models\UsersModel.php | public function getUserById($userId) {          $query = $this->db->where('userId', $userId)->get('lfy\_users');          return $query->row\_array(); // Fetch a single row      } |
| Route create | $route['api/userbyid/(:num)'] = 'UsersController/getUserById/$1'; |
| Call the view page | <http://localhost/1_api/pgination_CI/index.php/api/userbyid/1> |
|  |  |
| CALL THROGH POSTMAN | <http://localhost/1_api/pgination_CI/index.php/api/userbyid/10> |
|  |  |
|  |  |
| share data between two CodeIgniter projects using a REST API | |
|  | ****Step 1: Install CodeIgniter RESTful API Library**** You can use the [CodeIgniter Rest Server](https://github.com/chriskacerguis/codeigniter-restserver) library to easily set up RESTful APIs. Download or clone it into your first project:  bash  Copy code  git clone https://github.com/chriskacerguis/codeigniter-restserver.git  Place the files in your application/libraries directory. |
|  | To share data between two CodeIgniter projects using a REST API, you'll need to:   1. **Set up one project as the API server** that exposes specific data. 2. **Access the API from the second project** by sending HTTP requests to the API server.   Here’s a guide to help you set it up: **1. Set Up the API Server (First CodeIgniter Project)******Step 1: Install CodeIgniter RESTful API Library**** You can use the [CodeIgniter Rest Server](https://github.com/chriskacerguis/codeigniter-restserver) library to easily set up RESTful APIs. Download or clone it into your first project:  bash  Copy code  git clone https://github.com/chriskacerguis/codeigniter-restserver.git  Place the files in your application/libraries directory. ****Step 2: Create a Controller for the API**** Create an API controller that exposes the required data.  php  Copy code  <?php  defined('BASEPATH') OR exit('No direct script access allowed');  require APPPATH . '/libraries/REST\_Controller.php';  class UsersAPI extends REST\_Controller {  public function \_\_construct() {  parent::\_\_construct();  $this->load->model('UsersModel');  }  // API to get all users  public function users\_get() {  $users = $this->UsersModel->getAllUsers();  $this->response($users, REST\_Controller::HTTP\_OK); // Send the data in JSON format  }  // API to get a user by ID  public function user\_get($id) {  $user = $this->UsersModel->getUserById($id);  if ($user) {  $this->response($user, REST\_Controller::HTTP\_OK);  } else {  $this->response(['status' => false, 'message' => 'User not found'], REST\_Controller::HTTP\_NOT\_FOUND);  }  }  } ****Step 3: Update Routing**** Add the following routes to your application/config/routes.php file:  php  Copy code  $route['api/users'] = 'usersAPI/users';  $route['api/users/(:num)'] = 'usersAPI/user/$1'; ****Step 4: Set Up the Model**** php  Copy code  class UsersModel extends CI\_Model {  public function getAllUsers() {  $query = $this->db->get('users');  return $query->result\_array();  }  public function getUserById($id) {  $query = $this->db->get\_where('users', ['userId' => $id]);  return $query->row\_array();  }  } ****Step 5: Test Your API**** Your API is now available at:   * **Get all users**: http://your-server/api/users * **Get a specific user**: http://your-server/api/users/{id}   Test these endpoints using Postman or curl. **2. Access the API from the Second Project (Second CodeIgniter Project)** In the second project, you’ll need to consume the API by making HTTP requests to the first project’s API. ****Step 1: Create an API Client**** You can use cURL in CodeIgniter to make requests to the API.  php  Copy code  class ApiClient extends CI\_Controller {  private $apiUrl = 'http://first-server/api/'; // The API base URL of the first CodeIgniter project  // Function to get all users from the API  public function getUsers() {  $url = $this->apiUrl . 'users';  // Initialize cURL  $curl = curl\_init($url);  curl\_setopt($curl, CURLOPT\_RETURNTRANSFER, true);  $response = curl\_exec($curl);  curl\_close($curl);  // Convert JSON response to an array  $users = json\_decode($response, true);  // Use the data as needed  print\_r($users);  }  // Function to get a user by ID from the API  public function getUserById($id) {  $url = $this->apiUrl . 'users/' . $id;  // Initialize cURL  $curl = curl\_init($url);  curl\_setopt($curl, CURLOPT\_RETURNTRANSFER, true);  $response = curl\_exec($curl);  curl\_close($curl);  // Convert JSON response to an array  $user = json\_decode($response, true);  // Use the data as needed  print\_r($user);  }  } ****Step 2: Add Routes for API Client**** In your second project, add the following routes in application/config/routes.php:  php  Copy code  $route['api-client/users'] = 'ApiClient/getUsers';  $route['api-client/users/(:num)'] = 'ApiClient/getUserById/$1'; ****Step 3: Test API Calls**** In your second project, access the API using these URLs:   * **Get all users**: http://your-second-server/api-client/users * **Get a user by ID**: http://your-second-server/api-client/users/{id}  **3. Summary**  * The **first project** exposes data via a REST API using UsersAPI controller. * The **second project** consumes this data using ApiClient controller by making HTTP requests to the first project’s API.   This way, you can share data between two CodeIgniter projects through a REST API. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |