

CodeIgniter - Implementing user authentication and authorization

Topics : [Codeigniter](#)

Written on [March 01, 2024](#)

Implementing user authentication and authorization in CodeIgniter involves verifying user identities and controlling access to different parts of your application based on user roles or permissions.

1. Database Setup:

- Create a database table to store user information, including usernames, passwords, and roles/permissions.
- You can use CodeIgniter's migration feature or manually create the database table.

2. User Model:

- Create a model to handle interactions with the user database table (e.g., `User_model`).
- Implement methods for user authentication and fetching user information based on credentials.

3. User Authentication:

- Implement a login form where users can enter their credentials (username/email and password).
- Create a controller method to handle the login form submission.
- In the controller method, validate the user's credentials against the database using the User Model.
- If the credentials are valid, create a session for the authenticated user.

4. User Authorization:

- Define roles or permissions for different user types (e.g., admin, user).
- Create middleware or functions to check the user's role/permission before allowing access to certain parts of the application.
- Restrict access to specific controllers or methods based on the user's role/permission.

5. Protecting Routes:

- Use middleware or hooks to protect routes that require authentication or specific roles/permissions.
- Redirect unauthenticated users to the login page or display an error message.

6. Logout:

- Implement a logout functionality where users can log out of their sessions.
- Destroy the session when the user logs out to invalidate the session data.

7. Password Hashing:

- Hash user passwords before storing them in the database to enhance security.
- Use a strong hashing algorithm such as bcrypt to securely hash passwords.

8. Remember Me Functionality:

- Implement a "Remember Me" functionality using cookies to allow users to stay logged in across sessions.
- Use a secure and unique token to identify users and validate their sessions.

© Copyright **Aryatechno**. All Rights Reserved. Written tutorials and materials by [Aryatechno](#)