

Building a User Registration System using PHP and JSON

PHP CODE:

```
registration.php X {} users.json
C: > xampp >htdocs > PHP > registration.php

1  {?php
2  // Function to validate email address format
3  function validateEmail($email) {
4  |   return filter_var($email, FILTER_VALIDATE_EMAIL);
5  | }
6
7  // Function to validate password strength
8  function validatePassword($password) {
9  |   // Add your custom password validation rules here
10 |   // Example: Check minimum length, presence of special characters, etc.
11 |   return (strlen($password) >= 8);
12 | }
13
14 // Function to check if email already exists in users.json
15 function isEmailExists($email) {
16 |   $users = getUsersFromJson();
17 |   foreach ($users as $user) {
18 |       if ($user['email'] === $email) {
19 |           return true;
20 |       }
21 |   }
22 |   return false;
23 | }
24
25 // Function to get users array from users.json
26 function getUsersFromJson() {
27 |   $usersData = file_get_contents('users.json');
28 |   return json_decode($usersData, true);
29 | }
30
31 // Function to get users array from users.json
32 function getUsersFromJson() {
33 |   $usersData = file_get_contents('users.json');
34 |   return json_decode($usersData, true);
35 | }
36
37 // Function to save users array to users.json
38 function saveUsersToJson($users) {
39 |   $usersData = json_encode($users);
40 |   file_put_contents('users.json', $usersData);
41 | }
42
43 // Initialize variables
44 $nameErr = $emailErr = $passwordErr = $confirmPasswordErr = '';
45 $name = $email = '';
46
47 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
48 |   // Handle form submission
49 |
50 |   // Retrieve form data
51 |   $name = $_POST['name'];
52 |   $email = $_POST['email'];
53 |   $password = $_POST['password'];
54 |   $confirmPassword = $_POST['confirm_password'];
55 |
56 |   // Perform form validation
57 |   if (empty($name)) {
58 |       $nameErr = 'Name is required';
59 |   }
60 | }
```

```

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Handle form submission

    // Retrieve form data
    $name = $_POST['name'];
    $email = $_POST['email'];
    $password = $_POST['password'];
    $confirmPassword = $_POST['confirm_password'];

    // Perform form validation
    if (empty($name)) {
        $nameErr = 'Name is required';
    }

    if (empty($email)) {
        $emailErr = 'Email is required';
    } elseif (!validateEmail($email)) {
        $emailErr = 'Invalid email format';
    } elseif (isEmailExists($email)) {
        $emailErr = 'Email is already registered';
    }

    if (empty($password)) {
        $passwordErr = 'Password is required';
    } elseif (!validatePassword($password)) {
        $passwordErr = 'Password must be at least 8 characters long';
    }
}

```

registration.php X

{ } users.json

C:\> xampp > htdocs > PHP > registration.php

```

59     } elseif (isEmailExists($email)) {
60         $emailErr = 'Email is already registered';
61     }
62
63     if (empty($password)) {
64         $passwordErr = 'Password is required';
65     } elseif (!validatePassword($password)) {
66         $passwordErr = 'Password must be at least 8 characters long';
67     }
68
69     if ($password !== $confirmPassword) {
70         $confirmPasswordErr = 'Passwords do not match';
71     }
72
73     // If there are no errors, proceed with registration
74     if (empty($nameErr) && empty($emailErr) && empty($passwordErr) && empty($confirmPasswordErr)) {
75         // Hash the password
76         $hashedPassword = password_hash($password, PASSWORD_DEFAULT);
77
78         // Create user array
79         $user = [
80             'name' => $name,
81             'email' => $email,
82             'password' => $hashedPassword
83         ];
84
85         // Get existing users and add the new user
86         $users = getUsersFromJson();
87         $users[] = $user;
88

```

```
registration.php X users.json
C: > xampp > htdocs > PHP > registration.php
64 | $passwordErr = 'Password is required';
65 | } elseif (!validatePassword($password)) {
66 |     $passwordErr = 'Password must be at least 8 characters long';
67 | }
68 |
69 | if ($password !== $confirmPassword) {
70 |     $confirmPasswordErr = 'Passwords do not match';
71 | }
72 |
73 | // If there are no errors, proceed with registration
74 | if (empty($nameErr) && empty($emailErr) && empty($passwordErr) && empty($confirmPasswordErr)) {
75 |     // Hash the password
76 |     $hashedPassword = password_hash($password, PASSWORD_DEFAULT);
77 |
78 |     // Create user array
79 |     $user = [
80 |         'name' => $name,
81 |         'email' => $email,
82 |         'password' => $hashedPassword
83 |     ];
84 |
85 |     // Get existing users and add the new user
86 |     $users = getUsersFromJson();
87 |     $users[] = $user;
88 |
89 |     // Save updated users array to users.json
90 |     saveUsersToJson($users);
91 |
92 |     // Display success message
93 |     $successMessage = 'Registration successful!';
94 | }
```

HTML CONTATING PHP CODE:

```
registration.php X {} users.json
C: > xampp > htdocs > PHP > registration.php
98 | <!DOCTYPE html>
99 | <html>
100 | <head>
101 | | <title>User Registration</title>
102 | </head>
103 | <body>
104 | | <h2>User Registration</h2>
105 | | <form method="POST" action="<?php echo htmlspecialchars($_SERVER['PHP_SELF']); ?>">
106 | | | <label for="name">Name:</label>
107 | | | <input type="text" name="name" value="<?php echo $name; ?>">
108 | | | <span style="color: red;"><?php echo $nameErr; ?></span>
109 | | | <br><br>
110 | | | <label for="email">Email:</label>
111 | | | <input type="email" name="email" value="<?php echo $email; ?>">
112 | | | <span style="color: red;"><?php echo $emailErr; ?></span>
113 | | | <br><br>
114 | | | <label for="password">Password:</label>
115 | | | <input type="password" name="password">
116 | | | <span style="color: red;"><?php echo $passwordErr; ?></span>
117 | | | <br><br>
118 | | | <label for="confirm_password">Confirm Password:</label>
119 | | | <input type="password" name="confirm_password">
120 | | | <span style="color: red;"><?php echo $confirmPasswordErr; ?></span>
121 | | | <br><br>
122 | | | <input type="submit" value="Register">
123 | | </form>
124 | | <?php if (!empty($successMessage)): ?>
125 | | | <div style="color: green;"><?php echo $successMessage; ?></div>
126 | | <?php endif; ?>
127 | </body>
128 | </html>
```

JASON CODE:

```
registration.php {} users.json X
C: > xampp > htdocs > PHP > {} users.json > {} 1 > password
1 [{"name":"bibek","email":"bibek44@gmail.com","password":"$2y$10$R17WTUNjz5e05Fv3NRG5VOHgWek46q19P87XP6o\7\shoUwWpD\5"},{"name":"sunil","email"
```

RUNNING AND DEBUGGING OF CODE:

← → ↻ ⓘ localhost/PHP/registration.php

User Registration

Name:

Email:

Password:

Confirm Password:
