Uttara University

Report

On

Course Name: Web Programming

Course Code: 0613301

Submitted to,

Md. Harun-Ar-Rashid
Course Teacher
CSE, UU

Department of Computer Science
& Engineering
Uttara University

Submitted by,

Moloy Chandra Das
ID: 2233091183
Batch- 60 (D)

Department of Computer Science
& Engineering
Uttara University

Submission Date: 06/05/2025

Title: Implementation of a User Authentication System in PHP with OTP Verification

---

Objective

To develop a user authentication system using PHP that includes:

- User Registration with email and mobile OTP verification
- User Login using email or mobile number
- Password Recovery through identity verification and OTP
- Management of basic and extra user information

Software and Tools Used

- **Backend**: PHP 8.x
- **Database**: MySQL / MariaDB
- **Frontend**: HTML, CSS, Bootstrap (optional)
- **Web Server**: Apache (XAMPP/LAMP/WAMP)
- **Mailer**: PHPMailer or SMTP

System Requirements

- Email and phone number verification with OTP
- Unique username
- Secure password handling (hashed)
- Forms for registration, login, and password recovery
- Basic and extra user information fields
- Identity verification during password reset

Modules and Functionality

1. User Registration

Inputs:

- Email → Send OTP (via email)
- Mobile Number → Send OTP (via SMS or mock)
- Password

- Basic Information (Fast Name, Last Name, Address, etc.)

Process:

1. User fills out the registration form.
2. System sends OTP to both email.
3. User enters OTPs for verification.
4. After validation, account is created with a unique username.
5. Password is hashed before storing in the database.

---

2. User Login

Inputs:

- Email or Mobile Number
- Password

Process:

1. User enters credentials.
2. System verifies login using email/mobile and password.
3. On success, user is redirected to the dashboard.
4. Username must be unique and associated with the account.

---

3. Password Recovery

Inputs:

- Identity Input (Email or Mobile)
- OTP Verification
- New Password

Process:

1. User clicks "Forgot Password".
2. Enters email or mobile number.
3. System sends OTP to verify identity.
4. Once verified, user can enter a new password.
5. Password is updated after hashing.

Security Measures

- Passwords are hashed using `password_hash()`.
- Email and phone OTPs expire after a short duration.
- Unique constraints on username, email, and phone.
- SQL injection protection with prepared statements.

Folder – Authentication System Project Structure

```
│
├── config/
│   └── db.php              ← Database connection
│
├── helpers/
│   └── functions.php       ← Functions: send OTP, verify, utilities
│
├── register.php            ← Registration form + Send OTP
├── verify_otp.php          ← OTP verification and final registration
├── login.php               ← User login (email or phone + password)
├── forgot_password.php     ← Request OTP for password reset
├── reset_password.php      ← Verify OTP and set new password
├── dashboard.php           ← Logged-in user dashboard (after login)
└── init.sql                ← SQL script to create necessary tables
```

helpers/functions.php

```
$mail->Host = 'smtp.gmail.com';
$mail->Username = 'your_email@gmail.com';    // Your email address
$mail->Password = 'your_app_password';       // Gmail App Password
$mail->Port = 587;
$mail->SMTPSecure = PHPMailer::ENCRYPTION_STARTTLS;
$mail->SMTPAuth = true;
```