

# Training Day 25 Report

14 July 2025

The twenty-fifth day of Web Development training focused on implementing admin authentication using login functionality. This ensures that only authorized users can access sensitive features such as viewing alumni records, feedback, or updating content.

## 1. Admin Login Form

Created a simple form for admin login with username and password fields.

Example Form:

```
<form method="POST" action="login.php">
  <input type="text" name="username" placeholder="Username" required>
  <input type="password" name="password" placeholder="Password" required>
  <input type="submit" value="Login">
</form>
```

## 2. Validating Login Credentials

Stored admin credentials in a database table (admin\_users)

Used SQL to verify credentials

Started a session on successful login and redirected to the dashboard

Example PHP Logic:

```
session_start();

$sql = "SELECT * FROM admin_users WHERE username='$username' AND
password='$password'";
```

Note: Basic implementation used plaintext passwords. Later improvement may include password hashing for better security.

### 3. Hands-on Practice

Designed and styled the login page

Connected form to backend for validation

Used PHP sessions to manage logged-in status

Added login check (`if(!isset($_SESSION['admin']))`) to protect admin pages

### 4. Relevance to Project

Admin authentication is critical for securing the portal. It prevents unauthorized access to alumni data, feedback messages, and dashboard controls—ensuring data integrity and privacy.

### 5. Learning Outcomes

Understood user authentication flow in PHP

Learned how to use sessions for access control

Strengthened backend login logic

Improved project security through protected admin routes

By: Rida

URN: 2302647

CRN: 2315192

D3(CSE-D)