Information Assurance and Security 1
Laboratory Activity 5
Authentication and Username Requirements

Mark Ian E. Ballesca
BSIT3 - BLK1

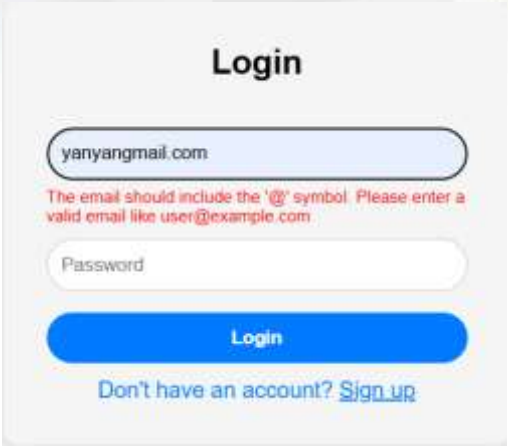
Information Assurance and Security 1
Th 7:30-8:30, F 7:30-8:30, 10:30-2:00

Source code

```php
<?php
session_start();
require 'db.php';

$error_message = "";

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $username = trim($_POST['username']);
    $password = $_POST['password'];


    if (!filter_var($username, FILTER_VALIDATE_EMAIL)) {
        $error_message = "Invalid email format. Please enter a valid email like user@example.com.";
    } else {

        $stmt = $conn->prepare("SELECT id, password FROM users WHERE username = ?");
        $stmt->bind_param("s", $username);
        $stmt->execute();
        $stmt->store_result();

        if ($stmt->num_rows == 1) {
            $stmt->bind_result($user_id, $hashedPassword);
            $stmt->fetch();

            if (password_verify($password, $hashedPassword)) {
                $_SESSION['user_id'] = $user_id;
                $_SESSION['username'] = $username;
                header("Location: dashboard.php");
                exit;
            } else {
                $error_message = "Invalid password.";
            }
        } else {
            $error_message = "User not found.";
        }
    }
}
```

Source Code

Sample

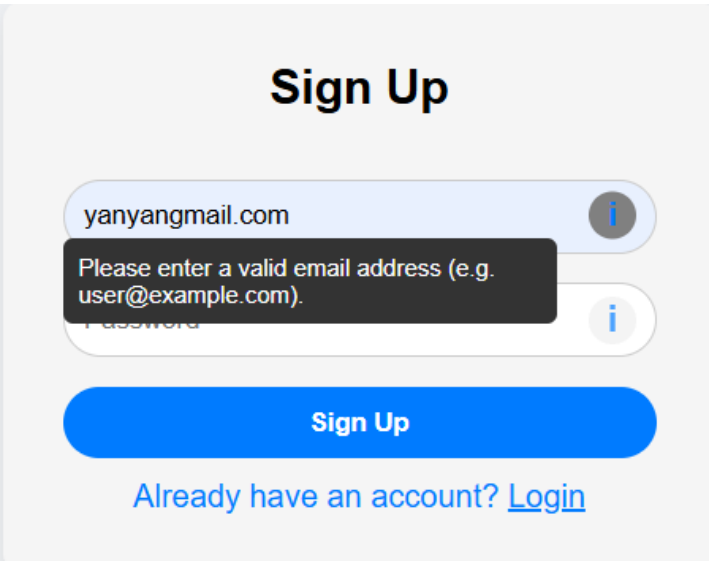


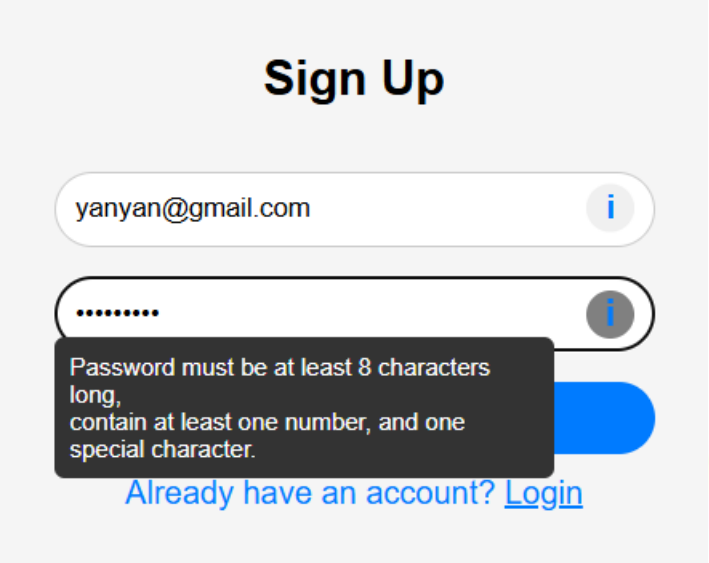
Sample

Source Code

```php
<?php
require 'db.php';

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $username = trim($_POST['username']);
    $password = $_POST['password'];

    // Basic password validation
    if (strlen($password) < 8 || !preg_match('/\d/', $password) || !preg_match('/[^a-zA-Z\d]/', $password)) {
        echo "Password must be at least 8 characters long and contain a number and special character.";
        exit;
    }

    $hashedPassword = password_hash($password, PASSWORD_DEFAULT);

    // Prepare and insert user
    $stmt = $conn->prepare("INSERT INTO users (username, password) VALUES (?, ?)");
    $stmt->bind_param("ss", $username, $hashedPassword);

    try {
        $stmt->execute();
        // Redirect to login page
        header("Location: login.html?success=1");
        exit;
    } catch (mysqli_sql_exception $e) {
        if ($e->getCode() === 1062) {
            echo "Username already exists!";
        } else {
            echo "Error: " . $e->getMessage();
        }
    }
}
?>
```

SAMPLE





Answer these questions

Explain the importance of authentication in securing systems and describe the key elements that make a password-based authentication method secure.

- Authentication is very important in keeping systems safe because it helps make sure that only the right people can access the system or information. It works like a security check that confirms who you are. A common way to do this is by using passwords. For a password-based method to be secure, the password should be long, hard to guess, and include a mix of letters, numbers, and special symbols. It should not be something easy like "123456" or your name. Also, passwords should be kept private and not shared with others. These simple steps help protect accounts and keep personal or important data safe from hackers.

Provide examples of common authentication requirements, and explain how they protect user data.

- Common authentication requirements include using a **strong password**, **two-factor authentication (2FA)**, and **security questions**. A strong password is one that includes uppercase and lowercase letters, numbers, and special characters, which makes it harder for hackers to guess. Two-factor authentication adds an extra layer of protection by requiring a second step, like entering a code sent to your phone, after typing your password. Security questions ask personal questions that only the user should know, adding another way to confirm identity. These methods help protect user data by making it much harder for unauthorized people to break into accounts, even if they know or guess the password.