



## Green University of Bangladesh



### Department of Computer Science and Engineering (CSE)

Faculty of Sciences and Engineering Semester:

(Fall, Year:2025), B.Sc. in CSE (Day)

**Lab Report NO #02**

**Course Title: Database Lab**

**Course Code: CSE 302    Section: 232\_D1**

**Lab Experiment Name:** Design and Development Login Page with Pattern Using HTML and CSS

#### **Student Details:**

Name	ID
Majharul Islam	232002256

**Lab Date** : 14 .10. 2025

**Submission Date** : 19.10. 2025

**Course Teacher's Name** : Tanpia Tasnim

[For Teachers use only: Don't Write Anything inside this box]

#### **Lab Report Status**

**Marks:** .....

**Comments:**.....

**Signature:**.....

**Date:**.....

# 1. TITLE OF THE LAB EXPERIMENT

Design and Development of a Login Page with Pattern Validation using HTML and CSS for user authentication with proper form structure and password validation rules.

## 2. OBJECTIVES

- To design a login form using HTML and CSS.
- To validate passwords using a regular expression pattern.
- To implement user-friendly design with CSS styling.
- To ensure the form structure follows accessibility and usability standards.

## 3. INTRODUCTION

In this experiment, a login page was developed using HTML and CSS, with additional password validation using a pattern attribute.

The project focuses on building secure and user-friendly authentication interfaces.

It ensures that users follow password rules (e.g., one uppercase, one lowercase, one number, one special character, and minimum eight characters).

The styling is handled through CSS for layout, alignment, and color themes to make the design professional and responsive.

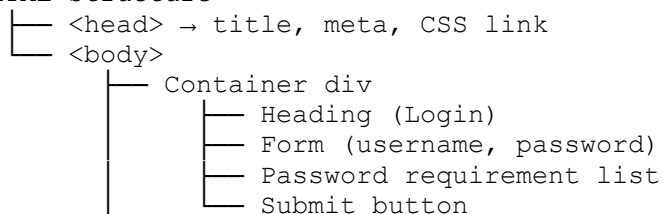
## 4. PROCEDURE

### Steps followed:

1. Created a new HTML file named `login_page.html`.
2. Linked an external stylesheet `login_page.css`.
3. Added form elements (username, password, and login button).
4. Used the pattern attribute to define strong password rules.
5. Used CSS for color scheme, text alignment, border-radius, and hover effects.
6. Added password toggle using JavaScript for better user experience.

### Design Flow:

#### HTML Structure



## 5. IMPLEMENTATION

### HTML CODE (Login\_page.html)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Login page</title>
  <link rel="stylesheet" href="login_page.css">
</head>
<body>
  <div class="container">
    <h1>Login</h1>
    <form action="/login" method="POST">
      <label for="username">Username:</label>
      <input type="text" id="username" name="username" required>
      <br><br>
      <label for="password">Password:</label>
      <div class="password-input-wrapper">
        <input type="password" id="password" name="password"
          pattern="^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@$!%*?&])[A-Za
z\d@$!%*?&]{8,}$"
          title="Password must be at least 8 characters long and contain at least one
uppercase letter, one lowercase letter, one number, and one special character
(@$!%*?&)"
          required>
        <span class="toggle-password" onclick="togglePassword()">
          <svg class="eye-icon" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 24
24" fill="none" stroke="currentColor" stroke-width="2" stroke-linecap="round"
stroke-linejoin="round">
            <path class="eye-open" d="M12s4-8 11-8 11 8-4 8-11 8-11-8-11-
8z"></path>
            <circle class="eye-open" cx="12" cy="12" r="3"></circle>
            <path class="eye-closed" d="M17.94 17.94A10.07 10.07 0 0 1 12 20c-7 0-11-
8-11-8a18.45 18.45 0 0 1 5.06-5.94M9.9 4.24A9.12 9.12 0 0 1 12 4c7 0 11 8 11 8a18.5 18.5 0
0 1-2.16 3.19m-6.72-1.07a3 3 0 1 1-4.24-4.24"></path>
```

```

        <line class="eye-closed" x1="1" y1="1" x2="23" y2="23"></line>

    </svg>

</span>
</div>
<div class="password-requirements">
    <p>Password must contain:</p>
    <ul>
        <li>At least 8 characters</li>
        <li>One uppercase letter</li>
        <li>One lowercase letter</li>
        <li>One number</li>
        <li>One special character (@$!%*?&)</li>
    </ul>
</div>
<button type="submit">Login</button>
</form>
</div>

<script>
    function togglePassword() {
        const passwordInput = document.getElementById('password');
        const eyeIcon = document.querySelector('.eye-icon');
        if (passwordInput.type === 'password') {
            passwordInput.type = 'text';
            eyeIcon.classList.add('show-password');
        } else {
            passwordInput.type = 'password';
            eyeIcon.classList.remove('show-password');
        }
    }
</script>
</body>
</html>

```

## CSS CODE (Login\_page.css)

```

* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}
body {

```

```
font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
display: flex;
justify-content: center;
align-items: center;
min-height: 100vh;
padding: 20px;
}
.container {
  background: white;
  padding: 40px;
  border-radius: 10px;
  box-shadow: 0 10px 25px rgba(0, 0, 0, 0.2);

  max-width: 400px;
  width: 100%;
}

h1 {
  color: #333;
  text-align: center;
  margin-bottom: 30px;
  font-size: 28px;
}

form {
  display: flex;
  flex-direction: column;
}

label {
  color: #555;
  font-weight: 600;
  margin-bottom: 8px;
  font-size: 14px;
}

.password-input-wrapper {
  position: relative;
  margin-bottom: 20px;
}

.password-input-wrapper input[type="password"],
.password-input-wrapper input[type="text"] {
  width: 100%;
  padding-right: 45px;
  margin-bottom: 0;
}

.toggle-password {
  position: absolute;
```

```
right: 12px;
top: 50%;
transform: translateY(-50%);
cursor: pointer;
display: flex;
align-items: center;
justify-content: center;
padding: 5px;
transition: all 0.3s ease;
}
.toggle-password:hover {
  opacity: 0.7;
}

.eye-icon {
  width: 20px;
  height: 20px;
  color: #667eea;
}
.eye-icon .eye-closed {
  display: none;
}
.eye-icon.show-password .eye-open {
  display: none;
}
.eye-icon.show-password .eye-closed {
  display: block;
}
input[type="text"],
input[type="password"] {
  padding: 12px 15px;
  border: 2px solid #e0e0e0;
  border-radius: 5px;
  font-size: 14px;
  transition: all 0.3s ease;
  margin-bottom: 20px;
}
input[type="text"]:focus,
input[type="password"]:focus {
  outline: none;
  border-color: #667eea;
  box-shadow: 0 0 3px rgba(102, 126, 234, 0.1);
}
input[type="text"]:invalid,
input[type="password"]:invalid {
```

```
border-color: #e74c3c;
}

input[type="text"]:valid,
input[type="password"]:valid {
border-color: #2ecc71;
}

button[type="submit"] {
background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
color: white;
padding: 12px 20px;
border: none;
border-radius: 5px;
font-size: 16px;
font-weight: 600;
cursor: pointer;
transition: transform 0.2s ease, box-shadow 0.2s ease;
margin-top: 10px;
}

button[type="submit"]:hover {

transform: translateY(-2px);
box-shadow: 0 5px 15px rgba(102, 126, 234, 0.4);

}
button[type="submit"]:active { transform: translateY(0)
}
.password-requirements {
background: #f8f9fa;
padding: 15px;
border-radius: 5px;
margin-bottom: 20px;
font-size: 12px;
color: #666;
}
.password-requirements p {
margin-bottom: 5px;
font-weight: 600;
color: #333;
}
.password-requirements ul {
margin-left: 20px;
margin-top: 8px;
```

```
}  
.password-requirements li {  
  margin-bottom: 5px;  
}
```

## 6. TEST RESULT / OUTPUT

### Observation:

- i. The login form loads properly in browsers.
- ii. Password validation works as expected.
- iii. Incorrect passwords show an alert before submission.
- iv. Eye icon toggles password visibility.

### Expected Output Screenshot:

**Login**

Username:

Password:

**Password must contain:**

- At least 8 characters
- One uppercase letter
- One lowercase letter
- One number
- One special character (@\$!%\*?&)

**Login**

## 7. ANALYSIS AND DISCUSSION

- i. **Analysis:** The use of the pattern attribute improves form validation directly in HTML.



- ii. **What went well:** Password validation and responsive layout.
- iii. **Challenges:** Ensuring pattern syntax matched all password requirements.
- iv. **Learning Outcomes:**
  - a. Learned password validation using regex pattern.
  - b. Improved understanding of CSS styling and HTML structure.
- v. **Achievement of Objective:** All objectives were successfully met.

## 8. SUMMARY

In this lab, a Login Page with Pattern Validation was successfully developed using HTML and CSS. The design was responsive, secure, and visually appealing. This experiment enhanced understanding of web form design, regex-based validation, and CSS layout principles.