

PASSWORD STRENGTH CHECKER

Description

This repository contains my practical work on password strength checker application that evaluates the strength of passwords entered by users.

Environment Used

- Visual Studio Code

Language Used

- HTML
- CSS
- JavaScript

Explanation

1. Criteria Checks:

- Length Criteria: Checks if the password is at least 8 characters long.
- Lowercase Criteria: Checks if the password contains at least one lowercase letter.
- Uppercase Criteria: Checks if the password contains at least one uppercase letter.
- Digit Criteria: Checks if the password contains at least one digit.
- Special Character Criteria: Checks if the password contains at least one special character. Unique character criteria: Ensures a good mix of unique characters.

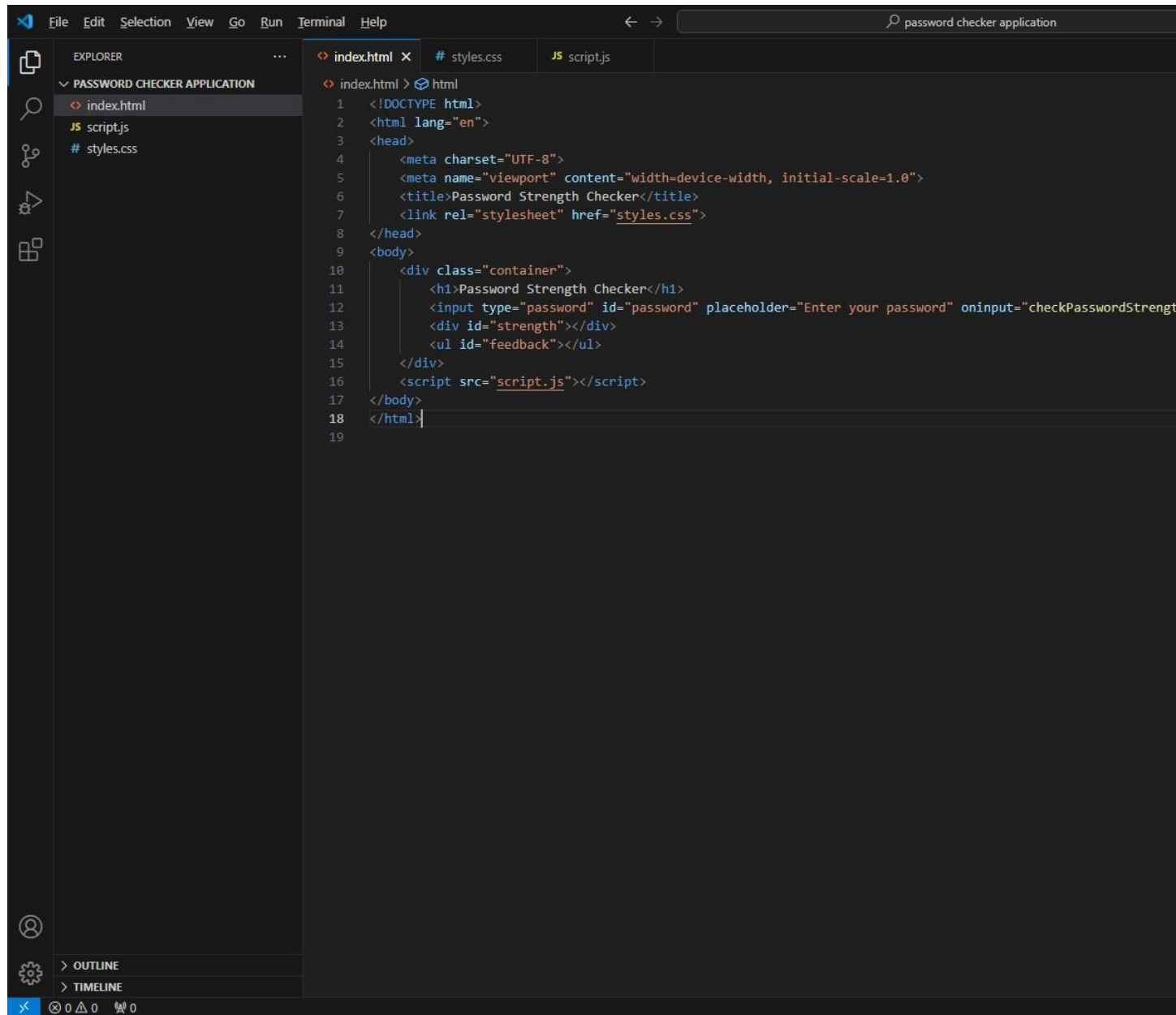
PASSWORD STRENGTH CHECKER INTERFACE

← → ↻ 📁 File C:/Users/USER/Desktop/password%20checker%20application/index.html

Password Strength Checker

Enter your password

HTML CODE

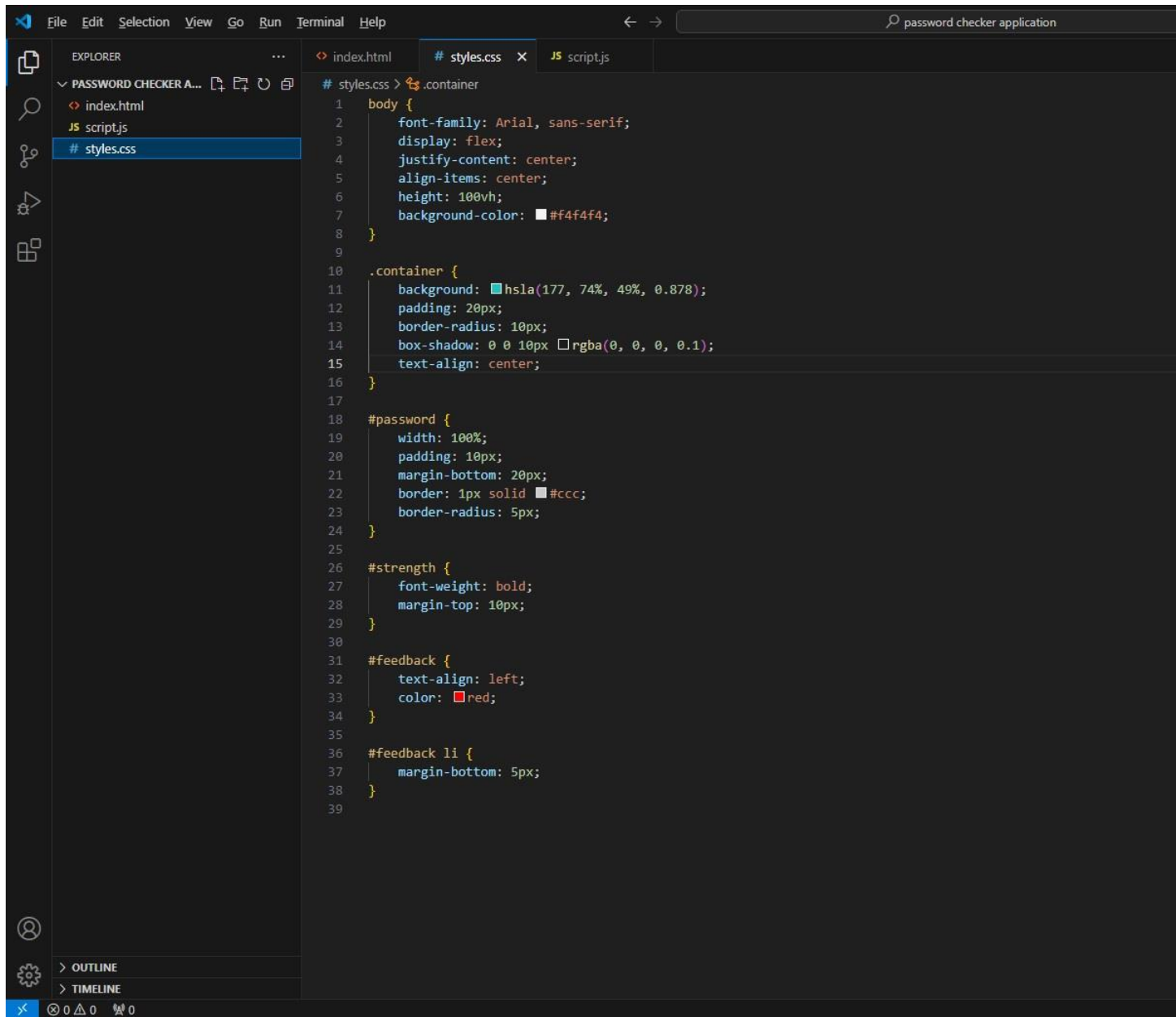


The image shows a Visual Studio Code editor window with a dark theme. The Explorer sidebar on the left shows a project named "PASSWORD CHECKER APPLICATION" containing three files: "index.html", "script.js", and "styles.css". The main editor area displays the "index.html" file, which contains the following HTML code:

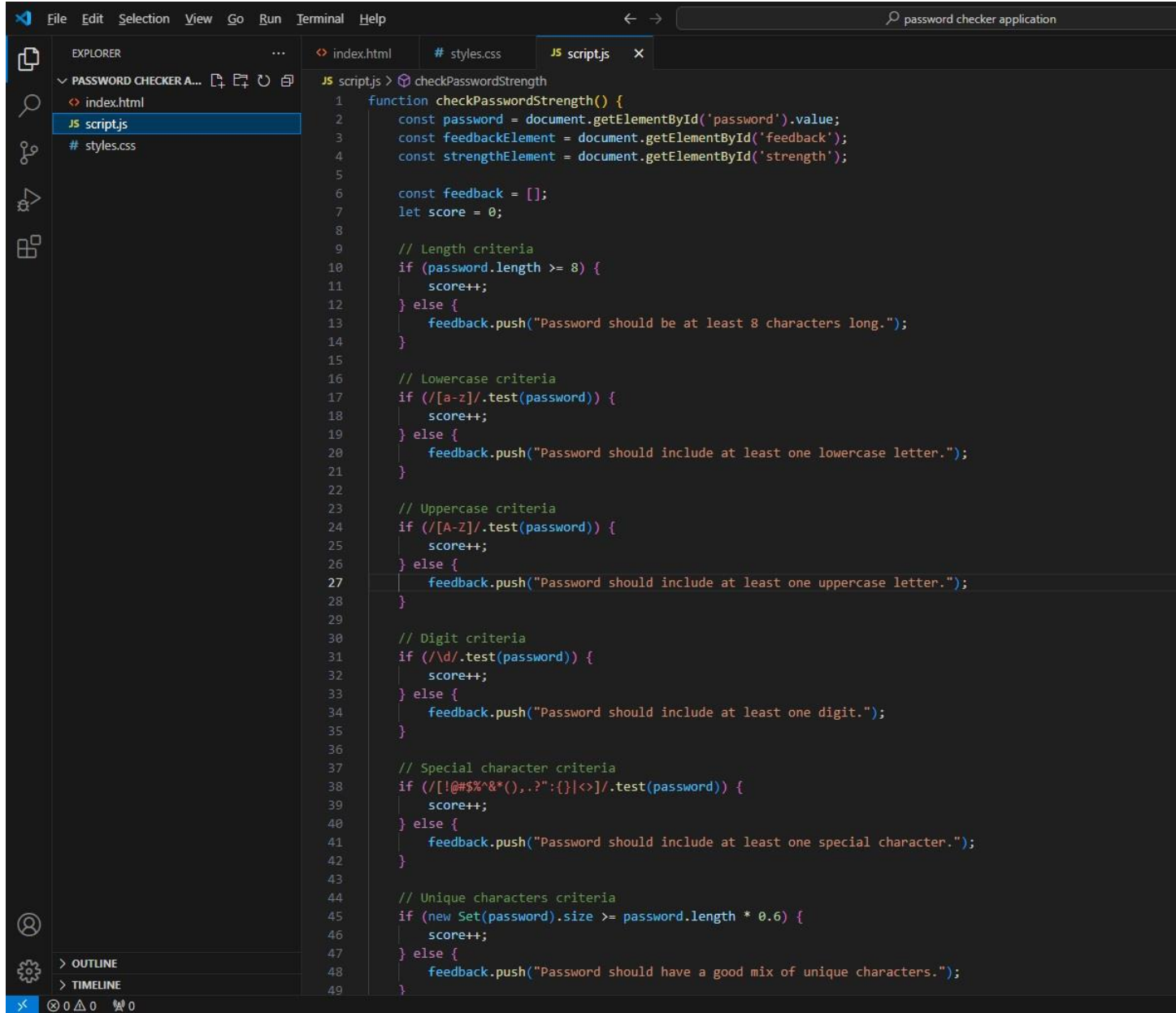
```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Password Strength Checker</title>
7   <link rel="stylesheet" href="styles.css">
8 </head>
9 <body>
10   <div class="container">
11     <h1>Password Strength Checker</h1>
12     <input type="password" id="password" placeholder="Enter your password" oninput="checkPasswordStrength">
13     <div id="strength"></div>
14     <ul id="feedback"></ul>
15   </div>
16   <script src="script.js"></script>
17 </body>
18 </html>
```

The status bar at the bottom left shows icons for Explorer, Search, Run and Debug, and Extensions, along with a search icon and a magnifying glass icon. The status bar at the bottom right shows the file name "index.html" and the file extension ".html".

CSS CODE



JAVASCRIPT



The image shows a screenshot of the Visual Studio Code editor interface. The Explorer panel on the left shows a project named 'PASSWORD CHECKER A...' with files 'index.html', 'script.js', and 'styles.css'. The 'script.js' file is selected and its content is displayed in the main editor. The code defines a 'checkPasswordStrength' function that evaluates a password based on several criteria: length (at least 8 characters), lowercase letters, uppercase letters, digits, special characters, and a mix of unique characters (at least 60% of the password length). Feedback messages are pushed into an array for each failed criterion.

```
1 function checkPasswordStrength() {
2   const password = document.getElementById('password').value;
3   const feedbackElement = document.getElementById('feedback');
4   const strengthElement = document.getElementById('strength');
5
6   const feedback = [];
7   let score = 0;
8
9   // Length criteria
10  if (password.length >= 8) {
11    score++;
12  } else {
13    feedback.push("Password should be at least 8 characters long.");
14  }
15
16  // Lowercase criteria
17  if (/[a-z]/.test(password)) {
18    score++;
19  } else {
20    feedback.push("Password should include at least one lowercase letter.");
21  }
22
23  // Uppercase criteria
24  if (/[A-Z]/.test(password)) {
25    score++;
26  } else {
27    feedback.push("Password should include at least one uppercase letter.");
28  }
29
30  // Digit criteria
31  if (/^\d/.test(password)) {
32    score++;
33  } else {
34    feedback.push("Password should include at least one digit.");
35  }
36
37  // Special character criteria
38  if (/[@#%&*(),.?":{}|<>]/.test(password)) {
39    score++;
40  } else {
41    feedback.push("Password should include at least one special character.");
42  }
43
44  // Unique characters criteria
45  if (new Set(password).size >= password.length * 0.6) {
46    score++;
47  } else {
48    feedback.push("Password should have a good mix of unique characters.");
49  }
```

```
1  function checkPasswordStrength() {
39      score++;
40  } else {
41      feedback.push("Password should include at least one special character.");
42  }
43
44      // Unique characters criteria
45      if (new Set(password).size >= password.length * 0.6) {
46          score++;
47      } else {
48          feedback.push("Password should have a good mix of unique characters.");
49      }
50
51      // Determine password strength
52      let strength;
53      if (score === 6) {
54          strength = "Strong";
55          strengthElement.style.color = 'green';
56      } else if (score >= 4) {
57          strength = "Medium";
58          strengthElement.style.color = 'orange';
59      } else {
60          strength = "Weak";
61          strengthElement.style.color = 'red';
62      }
63
64      // Update the UI
65      strengthElement.textContent = `Password Strength: ${strength}`;
66      feedbackElement.innerHTML = '';
67      feedback.forEach(item => {
68          const li = document.createElement('li');
69          li.textContent = item;
70          feedbackElement.appendChild(li);
71      });
72  }
73
```

2. Strength Calculation:

- The strength score is calculated by summing up the number of criteria met by the password.

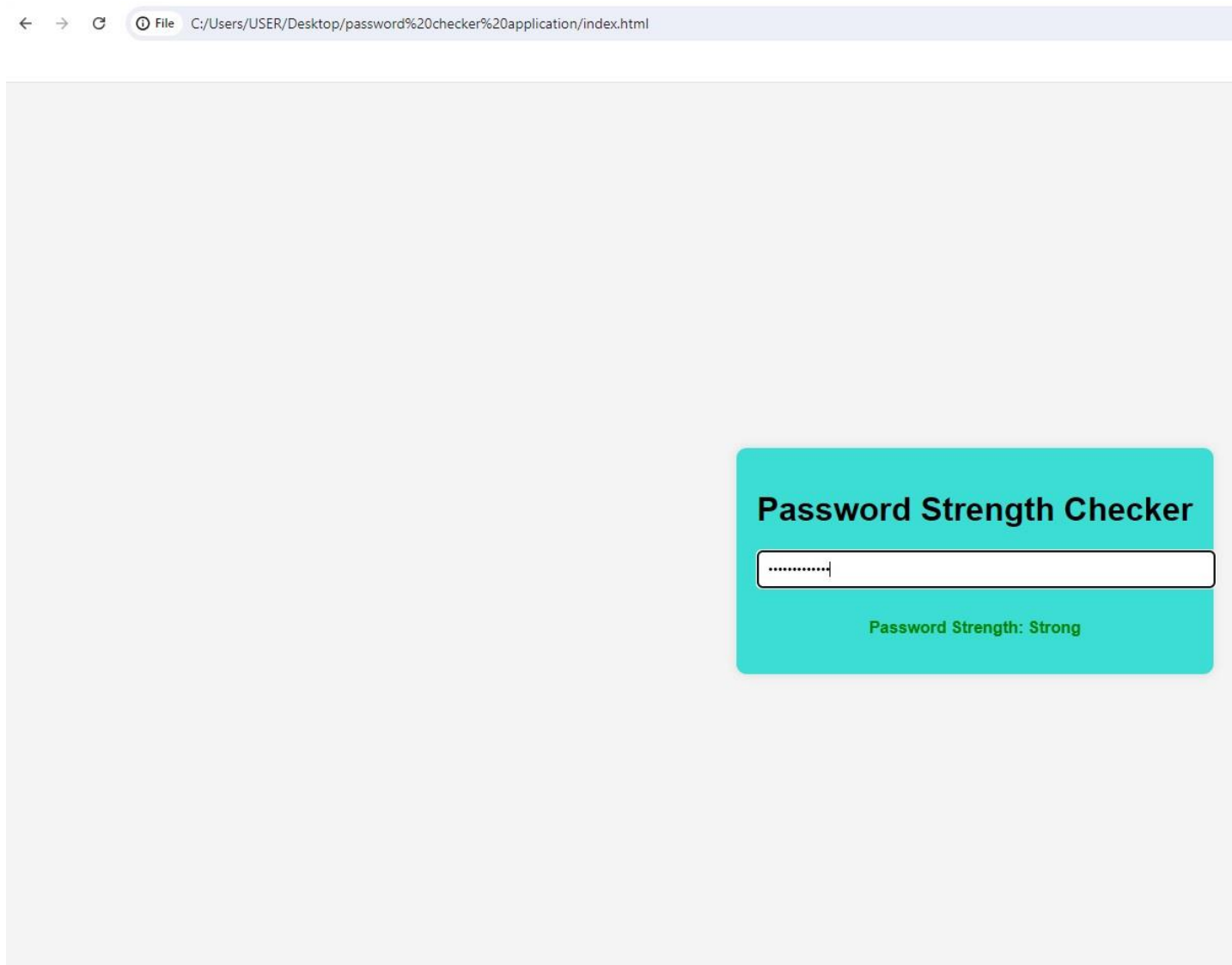
3. Feedback:

- Feedback is generated based on which criteria are not met.

4. Strength Determination:

- A password is considered "Strong" if all criteria are met, "Medium" if most criteria are met, and "Weak" if fewer criteria are met. The image below shows the result.

PASSWORD STRENGTH: STRONG



PASSWORD STRENGTH: MEDIUM

← → ↻ ⓘ File C:/Users/USER/Desktop/password%20checker%20application/index.html

Password Strength Checker

Password Strength: Medium

- Password should include at least one special character.

PASSWORD STRENGTH: WEAK

← → ↻ 📄 File C:/Users/USER/Desktop/password%20checker%20application/index.html

Password Strength Checker

Password Strength: Weak

- Password should be at least 8 characters long.
- Password should include at least one uppercase letter.
- Password should include at least one digit.
- Password should include at least one special character.

USAGE

- Run the script.
- Input the password when prompted
- Receive feedback on the strength of the password and suggestions for improvement.