**Report on Password Strength Validator**

**Project Title: Password Strength Validator**

**Introduction:**

This project is a web-based password strength validator designed to evaluate the strength of user passwords. It provides a visual indication of whether the password is **weak**, **medium**, or **strong**. The strength evaluation is determined using specific password criteria such as length, the inclusion of uppercase letters, numeric values, and special characters.

The project also features a toggle option to show or hide the password during input.

**Key Features:**

1. **Password Strength Check**
   * Evaluates password strength in real-time.
   * Strength levels:
     + Weak
     + Medium
     + Strong
2. **Visual Feedback**
   * Displays a strength meter with color-coded levels.
   * Provides textual feedback indicating the password's strength.
3. **Toggle Password Visibility**
   * Users can toggle between showing and hiding the password input.
4. **Responsive Design**
   * The interface adapts to different screen sizes and remains user-friendly.

**Implementation Details:**

**Technologies Used**

1. **HTML5** - For structuring the web page.
2. **CSS3** - For styling the page and the dynamic strength meter.
3. **JavaScript** - For implementing password validation logic and interactivity.

**Code Details:**

**1. HTML File (index.html)**

The HTML file provides the structure of the application, including:

* A container with input fields for email and password.
* A strength meter to display the password strength dynamically.
* A button to toggle password visibility.

**2. CSS File (style.css)**

The CSS file enhances the visual appeal of the application by:

* Styling the container and inputs.
* Creating animations and transitions for the strength meter.
* Adding color-coded feedback for weak, medium, and strong password levels.

**3. JavaScript File (app.js)**

The JavaScript file handles the application's logic:

* Defines the password strength evaluation logic through the Strength() function.
* Implements the dynamic behavior of the strength meter.
* Toggles password visibility.

**Password Evaluation Logic:**

The Strength() function uses the following criteria to assess the password:

1. Password length > 6 characters (adds 1 point).
2. Password length ≥ 10 characters (adds 1 point).
3. Includes at least one uppercase letter (adds 1 point).
4. Includes at least one numeric digit (adds 1 point).
5. Includes a combination of letters and numbers (adds 1 point).

Based on the cumulative score, the password is classified as:

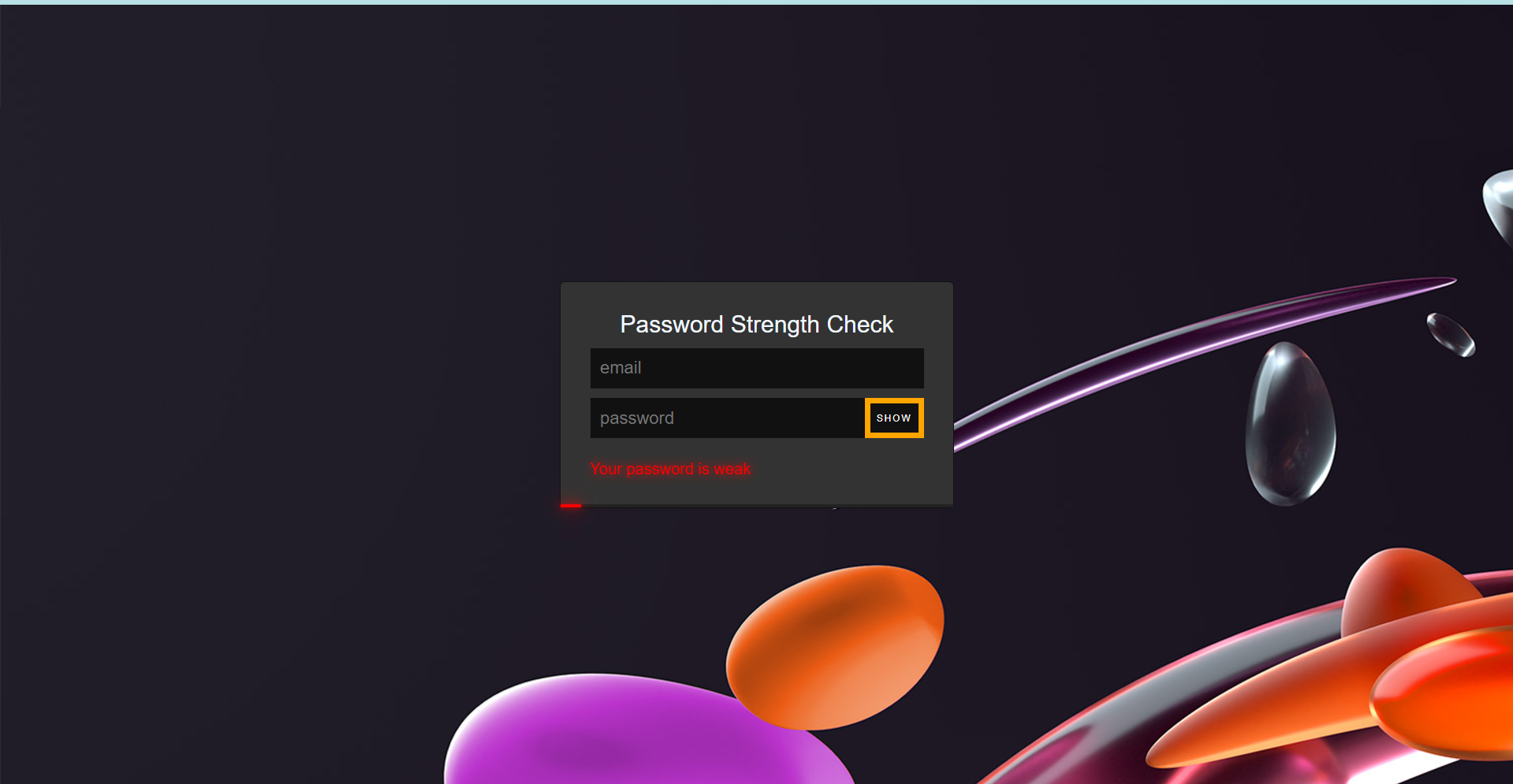
* Weak: ≤ 2 points
* Medium: 3–4 points
* Strong: 5 points

**How to Use the Application:**

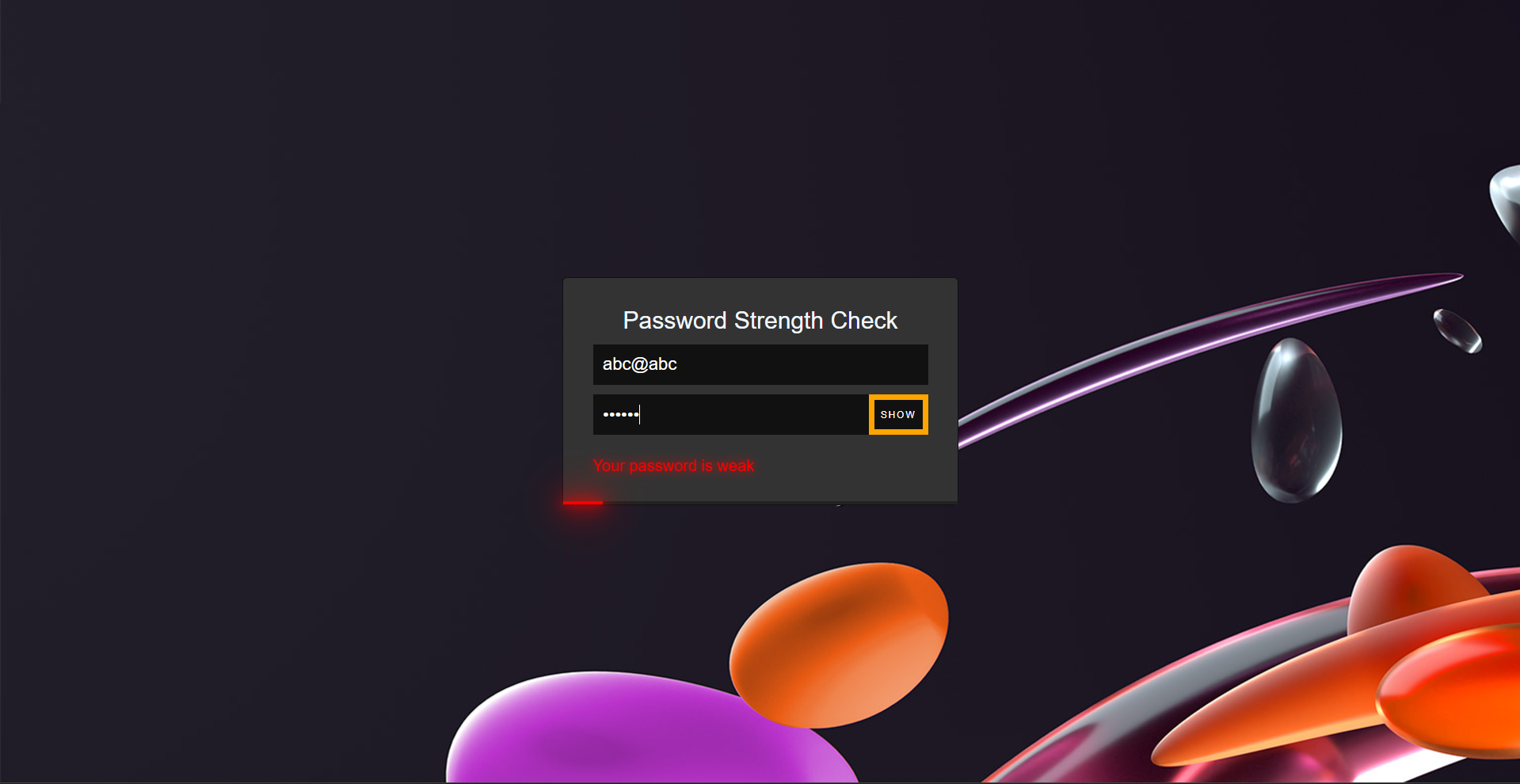
1. Open the HTML file in a browser.
2. Enter an email address in the designated field (optional).
3. Type a password in the password field.
   * Observe the dynamic feedback on the strength meter.
4. Toggle the visibility of the password by clicking the "Show" button.

**Screenshots:**

**Initial Screen:**

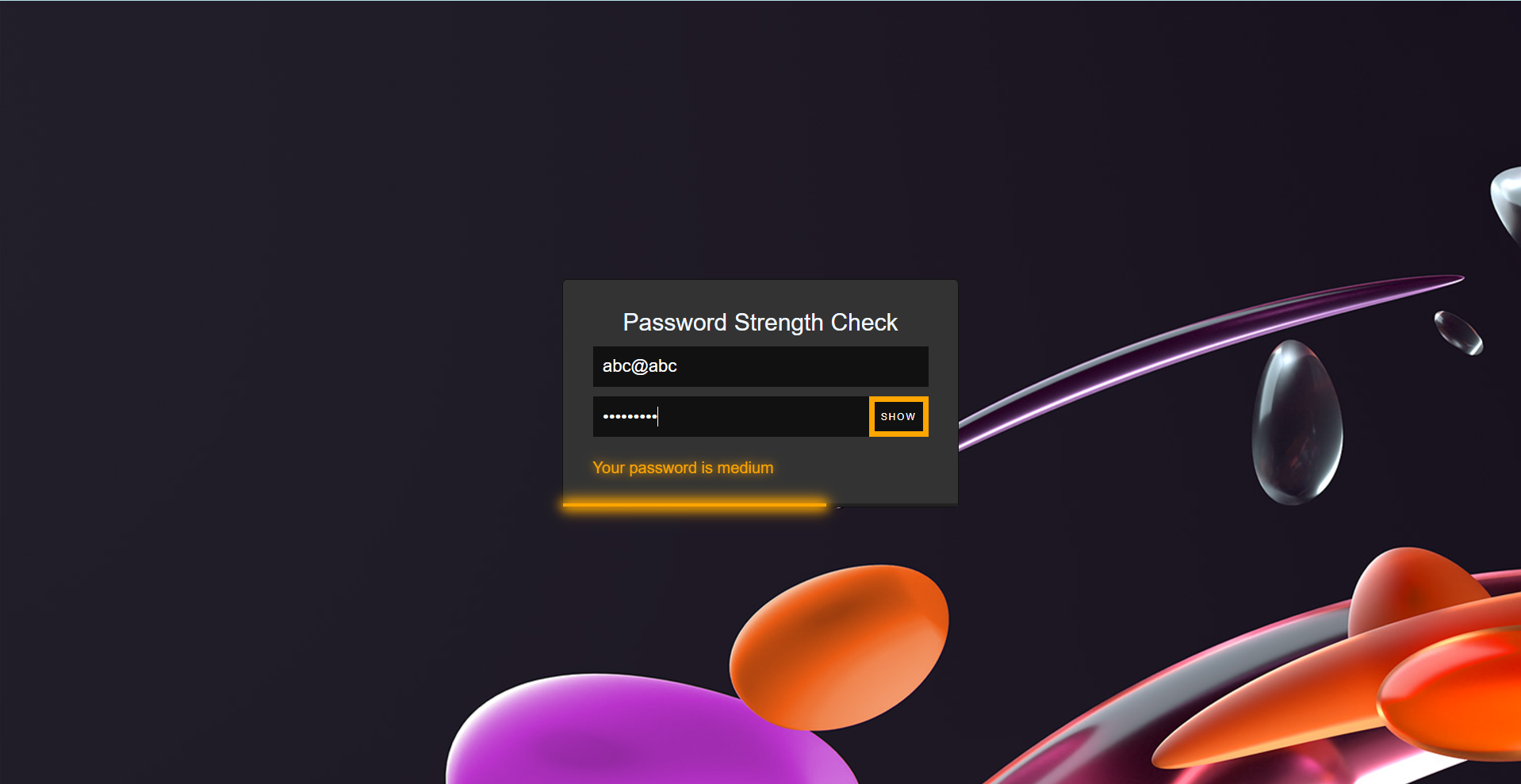
****

**Weak Password Example:**

****

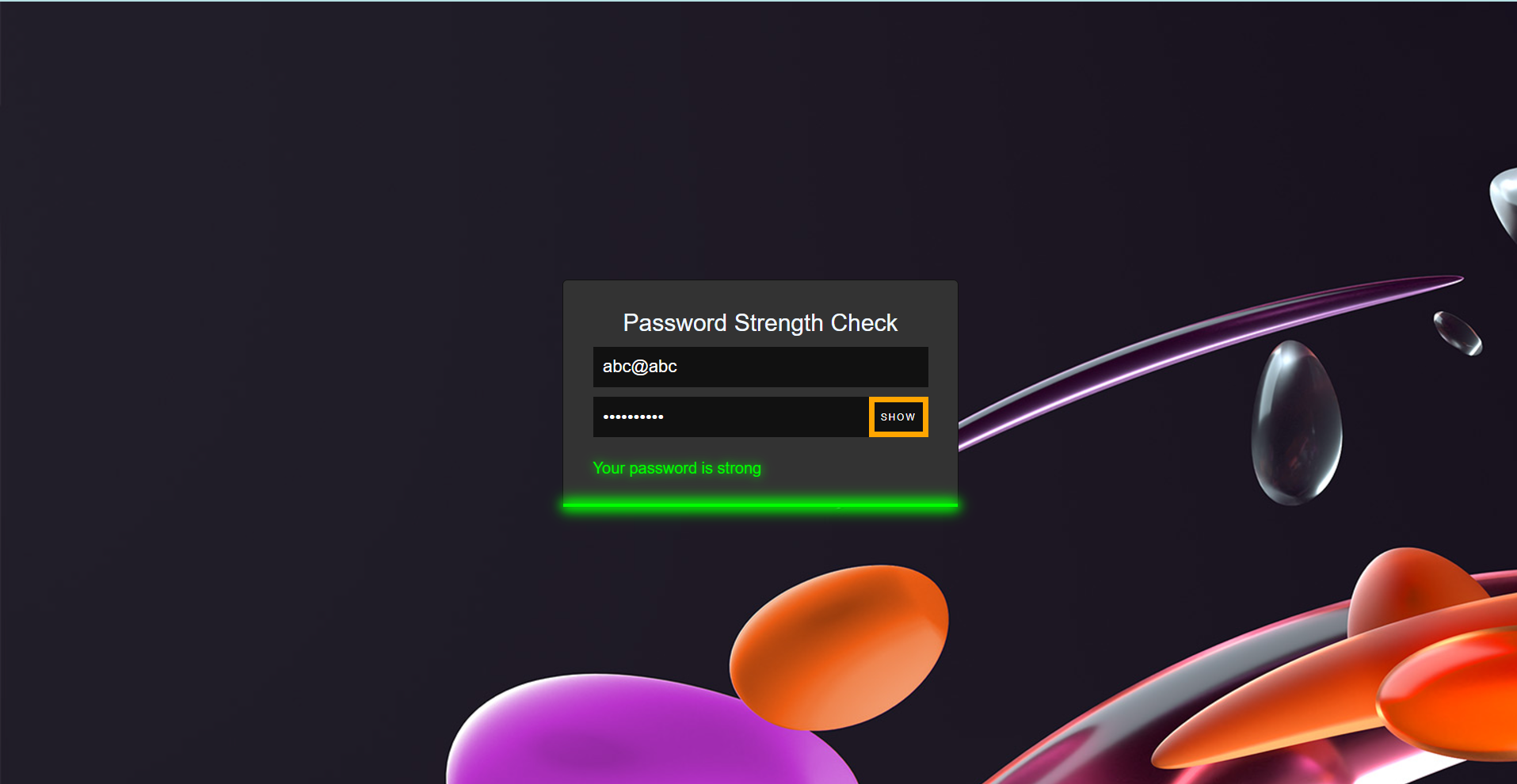
* Password: "abc123"
* Strength level: Weak
* Visual Feedback: Red strength meter.

**Medium Password Example:**

****

* Password: "abc123XYZ"
* Strength level: Medium
* Visual Feedback: Orange strength meter.

**Strong Password Example:**

****

* Password: "Abc123!@#"
* Strength level: Strong
* Visual Feedback: Green strength meter.

**Testing:**

The application was tested across multiple scenarios to validate the functionality:

1. Different password combinations to confirm accurate strength evaluation.
2. Responsive design on various screen sizes.
3. Toggle functionality for password visibility.

**Conclusion:**

The Password Strength Validator is a functional and interactive application that enhances user experience by providing immediate feedback on password security. It is a practical tool that encourages users to create stronger passwords.

**Submitted By:**  
Ankur