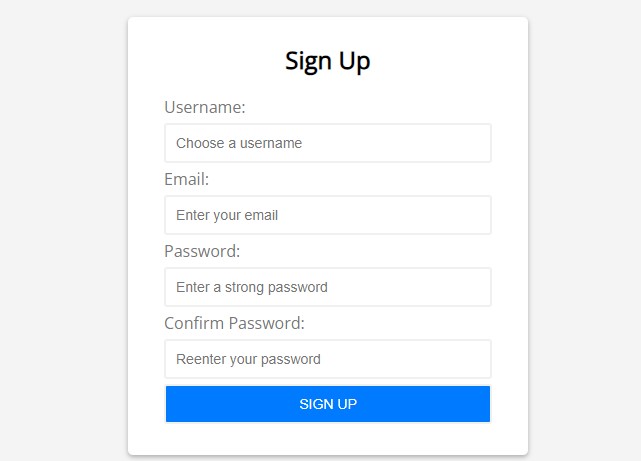
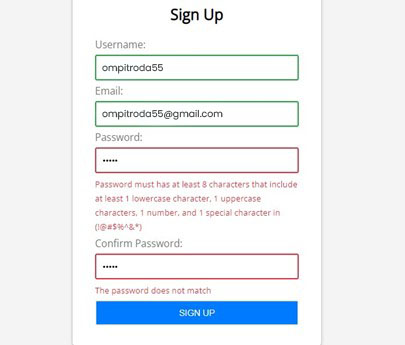
**Lab#2(a) Implement JavaScript Validations to the registration form using Event Listener and Events.**

|  |  |  |
| --- | --- | --- |
| **TAGS USED** | **ATTRIBUTES USED** | **DESCRIPTION** |
| <SCRIPT> |  | Denotes html code |
| <FUNCTION> |  | Initialize constant variable |
| <CONST> |  | To select input fields |
| <METHOD> | Username, email, password,  confirm password and sign  up | Check the field value |
| Regular expression |  | Check the field value |

**Final Outcome:**





**Code Snippet:**

<script>

const usernameEl = document.querySelector('#username'); const emailEl = document.querySelector('#email'); const passwordEl = document.querySelector('#password');

const confirmPasswordEl = document.querySelector('#confirm-password');

const form = document.querySelector('#signup');

const checkUsername = () => {

let valid = false;

const min = 3, max = 25;

const username = usernameEl.value.trim();

if (!isRequired(username)) { showError(usernameEl, 'Username cannot be blank.'); } else if (!isBetween(username.length, min, max)) {

showError(usernameEl, `Username must be between ${min} and ${max} characters.`)

} else {

showSuccess(usernameEl);

valid = true;

}

return valid;

};

const checkEmail = () => { let valid = false;

const email = emailEl.value.trim(); if (!isRequired(email)) {

showError(emailEl, 'Email cannot be blank.');

} else if (!isEmailValid(email)) {

showError(emailEl, 'Email is not valid.')

} else {

showSuccess(emailEl); valid = true;

}

return valid;

};

let valid = false;

const password = passwordEl.value.trim();

if (!isRequired(password)) {

showError(passwordEl, 'Password cannot be blank.');

} else if (!isPasswordSecure(password)) {

showError(passwordEl, 'Password must has at least 8 characters that include at least 1 lowercase character, 1 uppercase characters, 1 number, and 1 special character in (!@#$%^&\*)');

} else {

showSuccess(passwordEl);

valid = true;

}

return valid;

};

const checkConfirmPassword = () => {

let valid = false;

const confirmPassword = confirmPasswordEl.value.trim(); const password = passwordEl.value.trim();

if (!isRequired(confirmPassword)) {

showError(confirmPasswordEl, 'Please enter the password again');

} else if (password !== confirmPassword) {

showError(confirmPasswordEl, 'The password does not match');

} else {

showSuccess(confirmPasswordEl); valid = true;

}

return valid;

};

const isEmailValid = (email) => {

const re = /^(([^<>()\[\]\\.,;:\s@"]+(\.[^<>()\[\]\\.,;:\s@"]+)\*)|(".+"))@((\[[0-9]{1,3}\.[0-

9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/; return re.test(email);

};

const isPasswordSecure = (password) => {

const re = new RegExp("^(?=.\*[a-z])(?=.\*[A-Z])(?=.\*[0-9])(?=.\*[!@#\$%\^&\\*])(?=.{8,})"); return re.test(password);

};

const isRequired = value => value === '' ? false : true;

const isBetween = (length, min, max) => length < min || length > max ? false : true;

const showError = (input, message) => {

const formField = input.parentElement;

formField.classList.remove('success');

formField.classList.add('error');

const error = formField.querySelector('small'); error.textContent = message;

};

const showSuccess = (input) => {

const formField = input.parentElement;

formField.classList.remove('error'); formField.classList.add('success');

const error = formField.querySelector('small');

error.textContent = '';

}

form.addEventListener('submit', function (e) {

e.preventDefault();

let isUsernameValid = checkUsername(), isEmailValid = checkEmail(), isPasswordValid = checkPassword(),

isConfirmPasswordValid = checkConfirmPassword();

let isFormValid = isUsernameValid &&

isEmailValid && isPasswordValid && isConfirmPasswordValid;

if (isFormValid) {

}

});

const debounce = (fn, delay = 500) => {

let timeoutId;

return (...args) => {

if (timeoutId) {

clearTimeout(timeoutId);

}

timeoutId = setTimeout(() => { fn.apply(null, args)

}, delay);

};

};

form.addEventListener('input', debounce(function (e) {

switch (e.target.id) { case 'username': checkUsername(); break; case 'email': checkEmail(); break; case 'password': checkPassword(); break; case 'confirm-password': checkConfirmPassword();

break;

}

}));

</script>