**Experiment No. 7**

**Aim:** Use Ajax for user validation and to show the result on the same page below the submit button

**LO4:** Develop Rich Internet Application using AJAX

**Theory:**

**Code:**

**Index.html**

<!DOCTYPE html>

<html>

<head>

<title>AJAX Form Validation</title>

<style>

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

body { font-family: 'Poppins', sans-serif; background: #f5f5f5; display: flex; justify-content: center; align-items: center; min-height: 100vh; padding: 20px; color: #333; }

#mainform { background: white; padding: 30px; border-radius: 15px; box-shadow: 0 8px 32px rgba(0, 0, 0, 0.1); max-width: 500px; width: 100%; animation: slideIn 0.8s ease-in-out; }

@keyframes slideIn { from { opacity: 0; transform: translateY(-20px); } to { opacity: 1; transform: translateY(0); } }

h2 { font-size: 2rem; font-weight: 700; margin-bottom: 20px; text-align: center; color: #333; }

form { display: flex; flex-direction: column; gap: 15px; }

.form-group { display: flex; flex-direction: column; gap: 5px; }

label { font-size: 1rem; font-weight: 500; color: #555; }

input { width: 100%; padding: 10px; border: 2px solid #ddd; border-radius: 8px; font-size: 1rem; background: #f9f9f9; color: #333; outline: none; transition: all 0.3s ease-in-out; }

input:focus { border-color: #ffcc00; box-shadow: 0 0 10px rgba(255, 204, 0, 0.3); }

input[type="button"] { background: linear-gradient(90deg, #ffcc00, #ff8800); color: white; padding: 12px; font-size: 1rem; font-weight: bold; border: none; border-radius: 8px; cursor: pointer; transition: all 0.3s ease-in-out; box-shadow: 0 4px 8px rgba(255, 204, 0, 0.3); }

input[type="button"]:hover { box-shadow: 0 0 25px rgba(255, 204, 0, 0.5); transform: scale(1.05); }

#disply { margin-top: 20px; padding: 15px; background: #f9f9f9; border-radius: 8px; border-left: 5px solid #ffcc00; display: none; }

#disply h4 { font-size: 1.25rem; color: #333; margin-bottom: 10px; }

#disply h5 { font-size: 1rem; color: #555; margin-bottom: 5px; }

#disply p { font-size: 1rem; color: #777; }

@media (max-width: 768px) { #mainform { padding: 20px; } h2 { font-size: 1.75rem; } input { font-size: 0.9rem; } input[type="button"] { font-size: 0.9rem; } }

@media (max-width: 480px) { #mainform { padding: 15px; } h2 { font-size: 1.5rem; } input { font-size: 0.8rem; } input[type="button"] { font-size: 0.8rem; } }

</style>

</head>

<body>

<div id="mainform">

<h2>AJAX Form Validation</h2>

<form id="myForm" name="myForm">

<div class="form-group"><label for="username1">Username</label><input id="username1" name="username" type="text" placeholder="Enter your username"><div id="username"></div></div>

<div class="form-group"><label for="password1">Password</label><input id="password1" name="password" type="password" placeholder="Enter your password"><div id="password"></div></div>

<div class="form-group"><label for="email1">Email</label><input id="email1" name="email" type="text" placeholder="Enter your email"><div id="email"></div></div>

<input onclick="checkForm()" type="button" value="Submit">

</form>

<div id="disply">

<h4>Details you submitted:</h4>

<h5>Username: <p id="usrname"></p></h5>

<h5>Email ID: <p id="mail"></p></h5>

</div>

</div>

<script>

function checkForm() {

var name = document.getElementById("username1").value;

var password = document.getElementById("password1").value;

var email = document.getElementById("email1").value;

if (name == "" || password == "" || email == "") {

alert("Fill All Fields");

} else {

var xhr = new XMLHttpRequest();

xhr.open("POST", "/validate", true);

xhr.setRequestHeader("Content-Type", "application/json");

xhr.onreadystatechange = function () {

if (xhr.readyState == 4 && xhr.status == 200) {

var response = JSON.parse(xhr.responseText);

if (response.errors) {

document.getElementById("disply").innerHTML = "<h4>Validation Errors:</h4>";

response.errors.forEach(function (error) {

document.getElementById("disply").innerHTML += "<p>" + error + "</p>";

});

} else {

document.getElementById("disply").innerHTML =

"<h4>Details you submitted:</h4>" +

"<h5>Username: <p>" + response.username + "</p></h5>" +

"<h5>Email ID: <p>" + response.email + "</p></h5>";

}

document.getElementById("disply").style.display = "block";

}

};

var data = JSON.stringify({ username: name, password: password, email: email });

xhr.send(data);

}

}

</script>

</body>

</html>

**Server.js**

const express = require("express");

const path = require("path");

const app = express();

const port = 3000;

// Serve static files (HTML, CSS, JS)

app.use(express.static(path.join(\_\_dirname)));

// Parse JSON request bodies

app.use(express.json());

// Route for the root URL

app.get("/", (req, res) => {

res.sendFile(path.join(\_\_dirname, "index.html"));

});

// Validation endpoint

app.post("/validate", (req, res) => {

const { username, password, email } = req.body;

const errors = [];

// Validate username

if (username.length < 3) {

errors.push("Username must be 3+ letters.");

}

// Validate password

if (password.length < 6) {

errors.push("Password must be 6+ characters.");

}

// Validate email

if (!/^[^\s@]+@[^\s@]+\.[^\s@]+$/.test(email)) {

errors.push("Invalid email address.");

}

if (errors.length > 0) {

res.json({ errors });

} else {

res.json({ username, email });

}

});

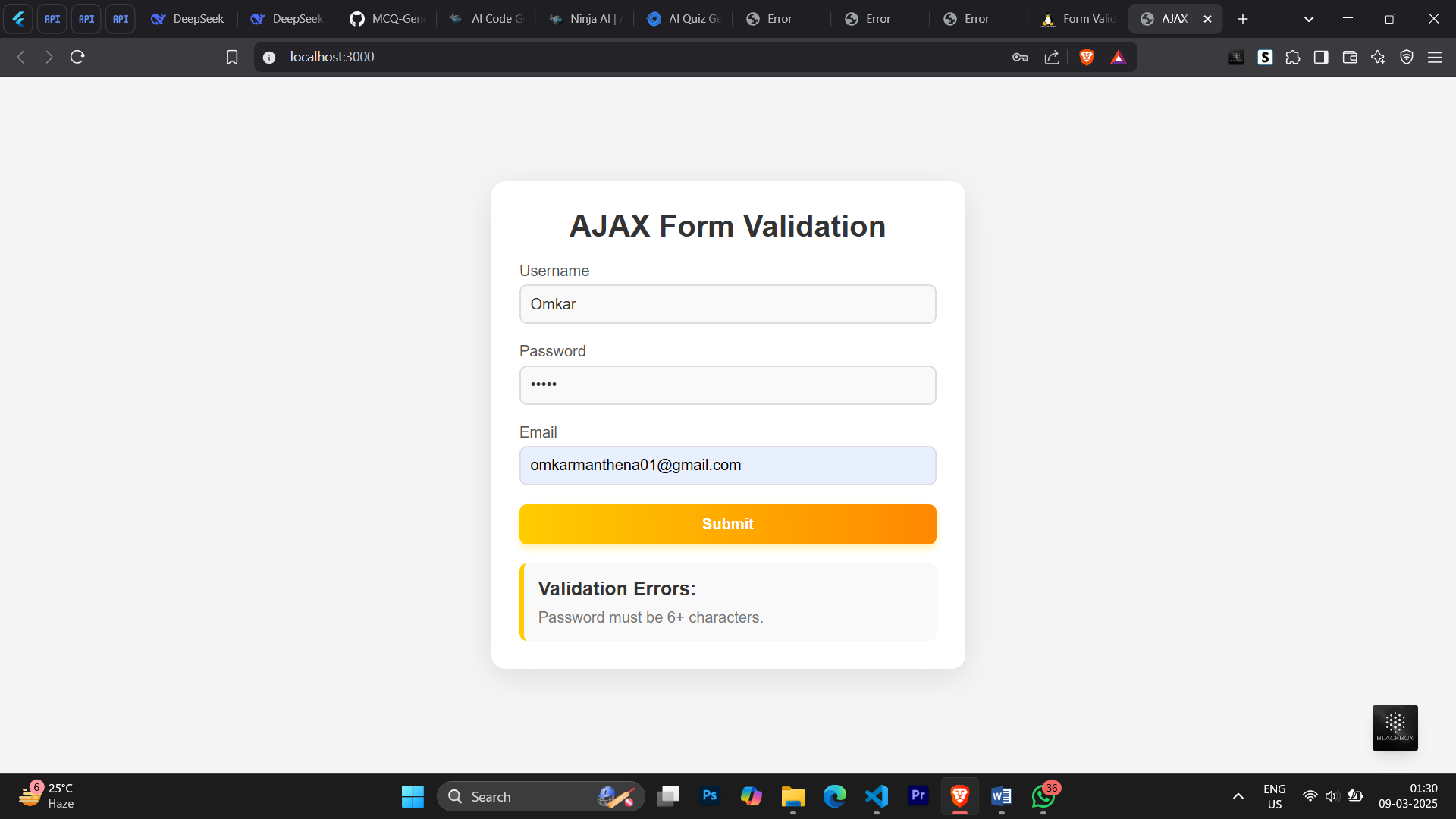
// Start the server

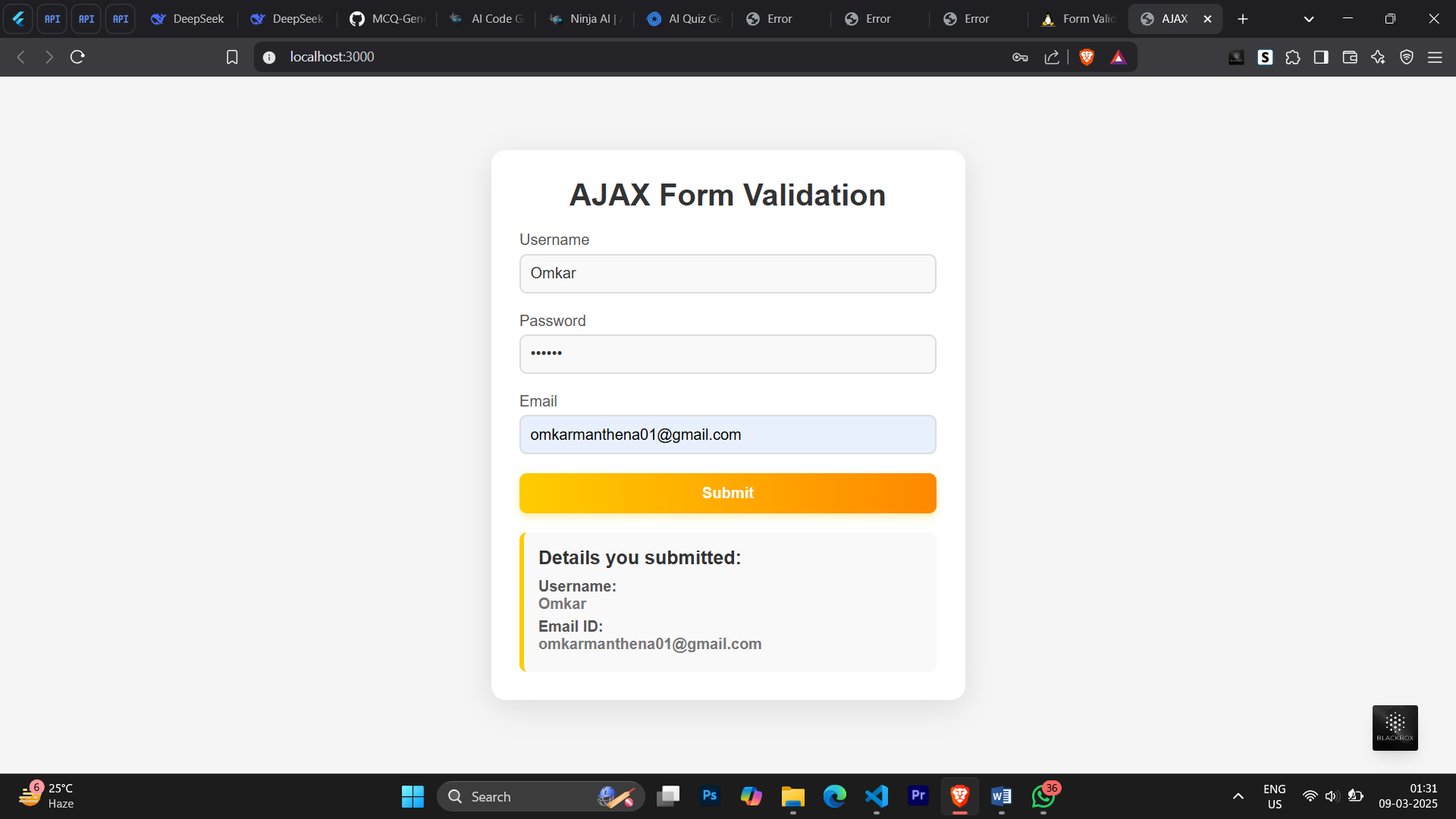
app.listen(port, () => {

console.log(`Server running at http://localhost:${port}`);

});

**Output:**





**Conclusion**: In this experiment, we successfully implemented an AJAX-based form validation system using HTML, CSS, JavaScript, and a Node.js backend. The project demonstrates how to create a user-friendly and responsive web form that validates user input in real-time without reloading the page.

**LO’s achieved:** LO4

**PO’s achieved:** PO1-PO5, PO8-PO12