**🛡️ Authentication Dashboard**

*A responsive full-stack authentication system with user signup/login, validation, hashed passwords, and a dynamic dashboard. Built with* ***Node.js****,* ***Express****,* ***MySQL****, and modern* ***HTML/CSS/JS****.*

**🚀 Features**

* ✅ User Signup & Login Forms
* ✅ Duplicate Username/Email Check
* ✅ Password Hashing using bcryptjs
* ✅ Dashboard with logout
* ✅ Categorized quiz & learning resources
* ✅ Responsive design with animations

This project is a full-featured Authentication Dashboard built using HTML, CSS, JavaScript, Node.js, Express.js, and MySQL. It features a clean, responsive frontend for user signup and login, with backend validation and secure password hashing using bcrypt. The system checks for duplicate usernames and emails, offers smart suggestions, and securely stores user credentials in a MySQL database. After logging in, users are redirected to a dashboard that includes logout functionality, user profile display, and a categorized list of learning and quiz resources. This project is a practical example of full-stack integration, with plans to use MongoDB in future versions.

🔧 Technologies & Basic Syntax

✅ HTML

<!-- Basic form -->

<form onsubmit="handleSignup(event)">

<input type="text" placeholder="Name" />

<input type="email" placeholder="Email" />

<input type="password" placeholder="Password" />

<button type="submit">Sign Up</button>

</form>

🎨 CSS

/\* Button style \*/

.logout-btn {

background: #e74c3c;

color: white;

padding: 0.5rem 1rem;

border-radius: 8px;

cursor: pointer;

transition: background 0.3s;

}

.logout-btn:hover {

background: #c0392b;

}

🧠 JavaScript

// Handle form submission

async function handleSignup(event) {

event.preventDefault();

const name = document.querySelector('input[placeholder="Name"]').value;

const res = await fetch('/api/users/signup', {

method: 'POST',

headers: { 'Content-Type': 'application/json' },

body: JSON.stringify({ name })

});

const data = await res.json();

alert(data.message);

}

🚀 Node.js + Express.js

// Basic Express setup

const express = require('express');

const app = express();

app.use(express.json());

app.post('/api/users/signup', (req, res) => {

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

res.send('Signup successful');

});

app.listen(3000, () => console.log('Server running'));

🛢️ MySQL (with mysql2)

const mysql = require('mysql2');

const db = mysql.createConnection({

host: 'localhost',

user: 'root',

password: 'your\_password',

database: 'auth\_db'

});

db.query('SELECT \* FROM users WHERE email = ?', [email], (err, results) => {

if (err) throw err;

});

🔐 bcryptjs

const bcrypt = require('bcryptjs');

const hashed = bcrypt.hashSync(password, 10);

const isMatch = bcrypt.compareSync(inputPassword, hashed);