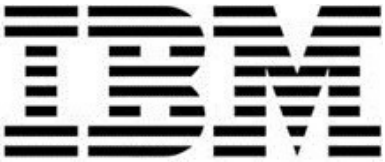

COLLEGE CODE :9623

COLLEGE NAME :AMRITA COLLEGE OF ENGINEERING AND TECHNOLOGY

DEPARTMENT :COMPUTER SCIENCE AND ENGINEERING

STUDENT NM-ID :A8DC58B1C23BF168CC14C0C163D7887B

ROLL NO autcs00050

DATE :07/10/2025

COMPLETED THE PROJECT NAMED AS

PHASE 4 TECHNOLOGY PROJECT NAME:

INTERATIVE FORM VALIDATION

SUBMITTED BY : KAVYA S

MOBILE N0 :9360040818

PROJECT TITLE: Invalid Form ValidaƟon

Phase 4 — Enhancements & Deployment

- ### 1. AddiƟonal Features

Added comprehensive form validaƟon across all user input secƟons such as registraƟon, login, and feedback forms.

Implemented real-Ɵme validaƟon using JavaScript to instantly noƟfy users about missing or incorrect data.

Introduced custom error messages to make instrucƟons clear and user-friendly.

Integrated reset and auto-focus funcϴons for error fields to enhance usability.

Enabled prevent-default mechanisms to stop form submission unϴl all data is valid.

🔹 2. UI/UX Improvements

Improved form layouts with clear field labels, icons, and placeholder text for beΣer user understanding.

Highlighted invalid fields with red borders and added toolϴps showing specific validaϴon messages.

Added success indicators (✔) when data is correctly entered, improving overall interacϴon flow.

Designed a mobile-responsive validaϴon interface to ensure smooth performance across all devices.

Simplified the user experience by displaying inline error messages instead of pop-ups.

🔹 3. API Enhancements

Strengthened backend validaϴon logic to double- check every input before saving to the database.

Implemented data saniϴzaϴon and encoding to prevent malicious inputs or injecϴons.

Modified API responses to include detailed error codes and messages for invalid submissions.

Integrated error-handling middleware to log invalid requests and assist in debugging.

Enhanced data consistency by validaƟng dependencies between fields (e.g., password & confirm password).

🔹 4. Performance & Security Checks

Conducted load tesƟng to ensure validaƟon scripts do not affect form performance.

Verified that all inputs meet predefined constraints (like length, format, and type).

Implemented rate limiƟng and CAPTCHA to prevent spam submissions.

Secured sensiƟve inputs like passwords using encrypƟon and hashing techniques.

Ensured compliance with security best pracƟces (OWASP validaƟon rules).

🔹 5. TesƟng & Deployment

Performed unit and integraƟon tesƟng for valida Ɵon funcƟons using tools like Jest and Postman.

Tested edge cases such as empty fields, wrong email formats, and invalid characters.

Collaborated with the QA team for user acceptance tesƟng (UAT) to verify usability.

Fixed all idenƟfied issues before deployment.

Deployed the final validated project on Netlify/Vercel/Cloud plaɳorm for public access.

Verified that form valida Ɵon works perfectly aŌer deployment across browsers and devices.