



COLLEGE CODE :9623
COLLEGE NAME :AMRITA COLLEGE OF ENGINEERING AND
TECHNOLOGY
DEPARTMENT :COMPUTER SCIENCE AND ENGINEERING
STUDENT NM-ID :752E80CE0530AB3B286425E0EC83A2E0
ROLL NO :962323104018
DATE :12/09/2025

COMPLETED THE PROJECT NAMED AS
PHASE2 TECHNOLOGY PROJECT
NAME:INTERACTIVE FORM VALIDATION

SUBMITTED BY,

M.S.Anush kumari

MOBILE NO:9790054171



Edit with WPS Office

 **Project title: Interactive form validation**

Phase 2 – Solution Design & Architecture

◆ **Tech Stack Selection**

Frontend: React.js (for form interactivity), Tailwind CSS (for styling), React Hook Form / Formik (form handling)

Backend: Node.js + Express.js (REST API)

Database: MongoDB or PostgreSQL

Validation Libraries: Yup / Joi (schema-based validation)

Auth (optional for signup form): JWT + bcrypt

◆ **UI Structure / API**



Schema Design

UI Components:

FormContainer (main form wrapper)

InputField (with validation feedback)

ErrorTooltip (inline error messages)

SubmitButton

SuccessMessage / ErrorMessage

API Schema (example for registration form):

```
// User Submission Model
{
  id: String,
  name: String,
```



```
email: String,  
passwordHash: String,  
phone: String,  
createdAt: Date,  
isValid: Boolean  
}
```

◆ Data Handling Approach

Frontend Validation: Handled by React Hook Form + Yup (fast, interactive feedback).

Backend Validation: Extra layer with Joi (ensures no bypass of frontend rules).

Data Security: Passwords hashed before storage.

Storage: Validated submissions stored in MongoDB/Postgres.

◆ Component / Module Diagram



Frontend:

Validation Module (field checks)

Error Handling Module (tooltips/messages)

Submission Module (form submit handler)

Backend:

Validation Middleware (Joi schema checks)

Submission Controller

Admin Controller (fetch submissions)

- Basic Flow Diagram

User → Frontend (React Form) → REST API (Express) → Database (MongoDB/Postgres)

