

IBM Interactive Form Validation Project Documentation

Phase 5: Final Demonstration & Documentation

# Table of Contents

* Introduction

* Project Overview

* Objectives

* Final Demo Walkthrough

* System Architecture

* Technologies Used

* Project Report

* API Documentation

* Screenshots

* Challenges and Solutions

* Testing and Validation

* Deployment Process

* GitHub Repository and Setup Guide

* Future Enhancements

* Conclusion

# Introduction

Interactive Form Validation is a web-based application developed as part of the IBM FrontEnd Development Project. The goal of this project is to build a responsive and user-friendly form validation system that ensures accurate data collection using JavaScript, HTML5, and CSS3. The form prevents incorrect or incomplete user inputs through real-time validation, error messages, and dynamic UI updates.

The project was executed in multiple phases, from requirement gathering to implementation, testing, and deployment. This document provides the final comprehensive documentation required in Phase 5: Final Demonstration & Documentation.

# Project Overview

The Interactive Form Validation system allows users to enter data into a form with built-in validation rules. It ensures data integrity by validating fields such as username, email, password, gender selection, phone number, and address.

The system also features:

Real-time validation messages

Error highlights

Responsive design

Input sanitization

# Objectives

The main objectives of the project were:

To develop a user-friendly form validation system

To implement client-side validation using HTML, CSS, and JavaScript

To ensure data is entered correctly and securely

To enhance the user experience with interactivity

To deploy the project on GitHub Pages for public access

# Final Demo Walkthrough

The final project demo includes a detailed walkthrough of user interactions with the system. Below are the demo steps:

Step 1: Navigate to Home Page

The user lands on the Home Page which provides an introduction to the project and navigation options.

Step 2: Access the Registration Form

The user clicks on the Register button to open the form page.

Step 3: Enter User Details

The form contains the following fields:

Full Name

Email Address

Password

Confirm Password

Gender (Radio Button)

Phone Number

Address (Textarea)

Step 4: Real-Time Validation

As the user enters data, the system performs validation in real-time:

Email must follow standard format

Password must be strong (min 8 chars, combination of letters & numbers)

Phone number must be 10 digits

Required fields cannot be empty

Step 5: Form Submission

On successful validation, the form is submitted and a success message is displayed.

# System Architecture

The system follows a simple client-side architecture:

User Interface (HTML + CSS)

|

Validation Logic (JavaScript)

No backend server is used in this phase of the project. All validation is handled on the client side.

# Technologies Used

Technology Purpose

HTML5 Structure

CSS3 Design & Styling

JavaScript Form Validation

Git & GitHub Version Control & Hosting

GitHub Pages Deployment

# Project Report

This section contains a detailed analysis of project implementation including design decisions, use cases, and workflows.

7.1 Use Case Diagram

The system identifies the following actors:

User – Inputs data into the form

7.2 Features Implemented

Input validation

Responsive UI

Clean error messages

JavaScript DOM manipulation

# API Documentation

This section contains JavaScript function documentation and validation logic. (To be expanded)

# Screenshots

Screenshots of the application will be included here.

# Challenges and Solutions

Challenges faced during development and their solutions are explained here.

1. Testing and Validation

Testing and validation are essential phases in ensuring that the Interactive Form Validation system functions accurately, efficiently, and securely. The goal is to verify that form inputs are properly validated before submission and that user interactions are smooth and intuitive.

1.1 Types of Testing Conducted

1.2

The following types of testing were performed:

Type of Testing Purpose Result

Unit Testing Testing individual validation functions like email, password, mobile number Successfully executed using JavaScript test data

Integration Testing Checked interaction between UI, validation logic, and error messages All components worked together without failure

System Testing End-to-end testing of the full form submission workflow Positive result

Usability Testing Ensured user-friendly error messages and design Enhanced clarity and feedback messages

Compatibility Testing Tested on Chrome, Firefox, Edge & Android mobile browser Fully compatible

Validation Testing Verified that input rules work correctly Strict validation achieved

1.3 Test Cases

1.4

Test Case ID Field Tested Input Expected Output Result

TC001 Email keer@com Show error: Invalid email Pass

TC002 Password 12345 Show error: Weak password Pass

TC003 Mobile 98765 Show error: Minimum 10 digits Pass

TC004 Submit Empty form Prevent submission Pass

1.3 Bug Fix Summary

Duplicate error messages removed

Improved mobile responsiveness

Added real-time validation

Password strength meter integrated

1. Deployment Process

The deployment process converts the locally developed project into a publicly accessible application. The project was deployed using GitHub Pages for hosting.

2.1 Prerequisites

2.2

Git installed

GitHub Account

HTML, CSS, and JavaScript project files

2.3 Deployment Steps

2.4

1. Initialize Git in project folder

Git init

Git add .

Git commit -m “Initial commit”

1. Create GitHub repository and push project

Git remote add origin <https://github.com/username/interactive-form-validation.git>Git push -u origin main

1. Enable GitHub Pages

Go to Repository → Settings → Pages

Source: main branch → root

1. Get live project link

➝<https://username.github.io/interactive-form-validation/>

2.3 Post Deployment Verification

Checked responsiveness

Checked performance

Validated hosting stability

Verified browser compatibility

3. GitHub README Setup Guide

# Interactive Form Validation

## Project Overview

This project focuses on validating user inputs in real-time using JavaScript.

## Features

Real-time error messages

Email and password validation

Responsive design

Error highlight feedback

## Technologies Used

* HTML
* CSS
* JavaScript

## Project Structure

/index.html

/style.css

/script.js

## ▶ How to Run

1. Download or clone the repo
2. Open index.html in browser

## Contribution

Feel free to fork and contribute!

## Contact

Email: your-email@gmail.com

* 1. Future Enhancements

To improve the system, the following features can be added:

Feature Description

OTP Verification Add email/mobile OTP

Multi-Step Forms Form wizard navigation

API Integration Connect with backend

Database Storage Save form details

Theme Mode Light/Dark theme support

Advanced Security Google reCAPTCHA

* 1. Conclusion

The IBM Interactive Form Validation project successfully demonstrates real-time form validation using JavaScript. It ensures accurate data entry, reduces user errors, enhances usability, and prevents invalid form submissions. The project is efficiently structured and hosted online using GitHub Pages for easy accessibility and scalability. With additional enhancements such as backend integration and security features, it can evolve into a complete web application validation module.

This project helped improve front-end development skills, use of Git for version control, and web deployment experience using GitHub. Overall, it provides a strong foundation for learning client-side validation and web development best practices.

# GitHub Repository

[**:https://github.com/Keerthana34191/Iterative-form-validation.git**](https://github.com/Keerthana34191/Iterative-form-validation.git)

**Netlify Repository:**

[**https://app.netlify.com/teams/keerthana34191/projects**](https://app.netlify.com/teams/keerthana34191/projects) **Screenshot:**



