# Dynamic Form Generator

The Dynamic Form Generator is a versatile and user-friendly tool designed to empower users in creating customizable forms effortlessly.

## Key Features

1. Form Customization

Field Types: Add various field types such as text inputs, text areas, dropdowns, checkboxes, radio buttons, and file uploads.

Labeling: Easily label each field for clear user understanding.

1. Conditional Fields

Dynamically show or hide fields based on the selected qualification.

1. Validation

Real-time Validation: Receive instant feedback on form validation to ensure data accuracy.

Error Messages: Clearly identify and rectify any errors in form entries.

1. Form Saving and Loading

JSON Configuration: Save and Load form configurations as JSON data for future use.

1. Submission and Editing

Submission: Submit forms with confidence after ensuring accurate data entry.

Editing Mode: Easily switch between form generation and editing modes.

1. User-Friendly Interface

Intuitive Design: Enjoy a sleek and user-friendly interface for a seamless form creation experience.

## Installation

1. Prerequisites:   
     
   Before getting started, ensure that you have the following installed on your machine:  
     
   A) Node.js: The JavaScript runtime.  
   B) npm: The Node.js package manager.
2. Steps to install and run:

A) Clone the Repository  
  
 <https://github.com/Vnay12/Dynamic-Form-Generator.git>

B) Navigate to the Project Directory

C) Install Dependencies   
  
 npm install

D) Start the application   
  
 npm run dev

## How to Use

1. Form Generation:
2. Provide a form name.
3. Select the qualification type.
4. Add various fields using the provided buttons.
5. Qualification-based Fields:
6. Choose a qualification to reveal additional fields specific to that qualification.
7. Form Validation:
8. Address any validation errors before submitting the form.
9. Submission:
10. Submit the form after ensuring all data is accurately entered.
11. Editing Mode:
12. Switch to editing mode to make adjustments to the form configuration.
13. Save and Load:
14. Save the form configuration as JSON for future use.
15. Load a previously saved configuration when needed.