

Eligibility Form Validation

Solution Approach:

1. Defining the Form Schema (Validation with Zod)

- We use **Zod** to define and enforce validation rules on user input.
 - The schema ensures:
 - **Full Name:** At least 2 characters.
 - **Age:** Between 17 and 24.
 - **Education Level:** Must be selected.
 - **Student Status:** Must be a boolean.
 - **University Name:** Required if the user is a student.
 - **GPA:** Between 0 and 4.
-

2. Setting Up the Form with React Hook Form

- **React Hook Form** is used for form handling, integrated with zodResolver for validation.
 - The form has default values to ensure proper state management.
-

3. Conditional Rendering for Student Fields

- If the user selects "**I am currently an active student**", additional fields (University and GPA) appear dynamically.
-

4. Eligibility Criteria Logic

- The user is considered **eligible** if:
 - They are between **17 and 24 years old**.
 - Their **GPA is at least 3.0**.
 - They have marked themselves as a **student**.
 - Their **education level is undergraduate**.
-

5. Providing Feedback via Toast Notifications

- If eligible, the user sees a **success toast**.
- If ineligible, a **destructive toast** appears, explaining why they are not eligible.

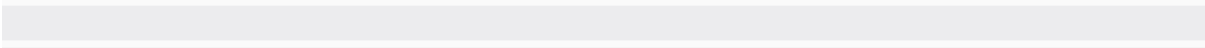
Advantages of this Approach

- ➡ **Scalable & Maintainable:** Uses zod for validation and react-hook-form for efficient form state management.
 - ➡ **User-Friendly:** Provides real-time feedback and error messages.
 - ➡ **Dynamic:** Conditionally renders fields based on user selections.
 - ➡ **Reusable Components:** Uses modular UI components from shadcn/ui.
-

Potential Enhancements

- **Improved Accessibility:** Use better ARIA attributes for screen readers.
 - **Form Persistence:** Save user input in local storage to prevent data loss.
 - **API Integration:** Submit form data to a backend for further processing
-

Interface



Eligibility Check

Please fill out this form to check your eligibility for the program.

Full Name

Age

You must be between 17 and 24 years old.

Education Level

Select your education level

☐ I am currently an active student

Check Eligibility

Conclusion

This eligibility form effectively validates user input and dynamically adjusts based on their selections. Using Zod for validation and React Hook Form for efficient state management ensures a seamless user experience. The form provides immediate feedback through toast notifications, making it intuitive and user-friendly. This approach ensures scalability, maintainability, and adaptability for future enhancements.