Multi-Step Form Implementation

Design Decisions

1. This form uses scalable components like Stepper, Progress Bar, Input Fields, and Navigation Buttons.
2. The Progress Bar provides visual feedback on form completion.
3. Step Indicator helps user to understand their current position.
4. Enhances user experience with animations.

Code Structure

1. Modular Javascript Functions: Functions for validation, progress updates, and navigation are separated for better maintainability.
2. CSS: Responsive design using media queries for mobile-friendliness.
3. Error Handling: Shows inline error messages for better feedback.

Error Handling Strategies: Client-side error handling ensures that users get immediate feedback on incorrect inputs before going to the next step.

1. Inline Validation checks as the user types, the form check if the input is valid or not. If incorrect, an error message appears, like enter 10-digit phone number.
2. Required Fields Check user cannot move to the next step if required fields is empty.
3. Step Validation Before moving to the next step, the form checks all fields in the current step.

Scalability & Usability Enhancements

1. Implement React, Vue, or Angular to manage the state efficiently.
2. Store form progress in localStorage or sessionStorage to prevent data loss.
3. Store form data in NoSQL(MongoDB, Firebase) or SQL database based on conditions.
4. Implement auto save to continue the form where we leave.
5. Add multi-language support to make the application accessible to a global audience.