GLOBAL CONSTRAINTS

✅ MANDATORY TECH STACK (NON-NEGOTIABLE)

Candidates **must use exactly the following stack**:

Core

- **React 18+**
- **TypeScript (strict mode enabled)**
  - `noImplicitAny`, `strictNullChecks`, `noUncheckedIndexedAccess` **must be ON**
- **Vite** (preferred) or **Next.js (App Router)**
  - No CRA, no legacy setups

Styling

- **Tailwind CSS (latest)**
- Must use:
  - Utility-first approach
  - CSS variables where applicable
  - Design-token–style abstractions (spacing, colors, radii)
- **NO inline styles** except for unavoidable dynamic calculations

Component Development

- **Storybook (latest)**
- **Chromatic** for publishing and visual regression
- All components **must be demonstrated ONLY via Storybook**

Tooling & Quality

- **ESLint + TypeScript ESLint**
- **Prettier**
- **Testing Library (React)** for interaction tests
- **axe-core / @storybook/addon-a11y** for accessibility checks

---

🚫 ABSOLUTE FORBIDDEN LIST (INSTANT DISQUALIFICATION)

Using **any item below = direct rejection**, no discussion.

❌ Component Libraries

You **CANNOT** use:

- MUI / Material UI
- Ant Design
- Chakra UI
- Mantine
- Radix UI
- Headless UI
- Blueprint
- ShadCN (or anything built on Radix)
- Fluent UI
- Carbon
- PrimeReact
- Any internal or external design system

👉 **All UI, primitives, and behavior must be built from scratch.**

❌ Prebuilt Hooks / Utilities

You **CANNOT** use:

- `react-table`
- `tanstack/table`
- `tanstack/virtual`
- `react-window`
- `react-virtualized`
- `downshift`
- `react-select`
- `floating-ui`
- `popper.js`
- `date-fns` picker components
- Any editor framework (Slate, ProseMirror, Quill, Lexical, TipTap, etc.)

👉 **Logic must be implemented manually.**

❌ CSS / Styling Shortcuts

- No Tailwind UI
- No paid/free UI kits
- No copy-pasting styles from:

    - GitHub repos
    - CodePen
    - Blogs
- No `@apply` abuse to recreate component libraries

❌ AI-Generated Code (Strict)

- You **may not** paste large AI-generated components
- You **must** be able to explain *every line* of logic
- If reviewers detect:
    - Pattern-level AI code
    - Unnecessary abstractions
    - “Too generic to be human” structure
      → **Instant rejection**

⚠️ ARCHITECTURE & QUALITY RULES

State Management

- **No Redux, Zustand, Jotai, Recoil**
- Use:
    - React state
    - Context **only where justified**
    - Custom hooks (written by you)

Accessibility (Non-Optional)

- Keyboard-first UX
- ARIA roles **must be correct**, not decorative
- Focus management **must be deliberate**
- Screen reader behavior must be testable and documented

Performance

- No unnecessary re-renders
- Memoization must be **intentional**, not blanket `useMemo`
- Virtualization logic must be **understood**, not imported

📦 STORYBOOK & CHROMATIC RULES

- Storybook **must be published publicly**
- Chromatic build **must pass**
- Stories must include:
  - Edge cases
  - Failure states
  - Loading states
  - High-contrast mode
  - Keyboard-only usage
- No “happy path only” stories

🧪 TESTING RULES

- Tests must:
  - Cover keyboard interactions
  - Validate a11y constraints
  - Assert failure behavior
- Snapshot-only testing = **invalid**

📛 DISQUALIFICATION CONDITIONS (NO EXCEPTIONS)

You will be **immediately disqualified** if:

1. Any forbidden library is detected
2. Component logic is outsourced to third-party utilities
3. Storybook is missing or private
4. TypeScript strict mode is disabled
5. Accessibility is treated as optional
6. Performance claims are not measured
7. Code ownership cannot be explained clearly