UI Standards

for a Web Application Page

Abstract

UI (User Interface) standards for software development are a set of guidelines that ensure consistency, usability, and accessibility in the user interfaces.

Help developers to create interfaces that are user-friendly, intuitive, and effective across different platforms and devices.

# TABLE OF CONTENTS

# **Check list: Must-Have Components in a Modern Web Application UI**

1. Navigation **3**
2. Page Header **4**
3. Main Content Area **5**
4. Actions & CTAs (Call to Actions) **7**
5. Feedback Elements **8**
6. Spacing and Layout **9**
7. Footer (Optional in App Pages) **10**
8. Search & Filters (if applicable) **11**

* **Example: Page Layout 12**
* **Do's and Don'ts 13**

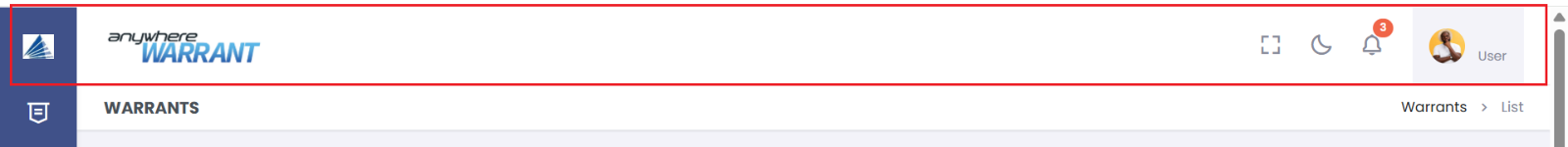
# **UI/UX Design Principles and Best Practices**

1. Clarity and Simplicity **14**
2. Consistency **15**
3. Color and Contrast **16**
4. Accessibility **17**
5. Interaction **18**
6. Responsive Design **19**
7. Performance and Efficiency **20**

* **Checklist: Must-Have Components in a Modern Web Application UI**

# Navigation

* **Top Navigation Bar** (if applicable)
* It should include   
  1) **Logo** (linked to Home or Dashboard)  
  2) **Global Actions** (e.g., search, settings, help)  
  3) **User Profile** (name, avatar, dropdown for more user specific actions)  
  4) **Notifications** (bell icon with badge)
* Place at the top of the viewport, horizontally aligned
* Stick or fix it on scroll if needed.
* Right-align profile and notifications.
* Ensure the logo is clickable and leads to a landing page.
* Use icons for compact global actions; include tooltips or labels on hover.



* **Sidebar/Left Navigation Menu**
* Position on the left side of the screen.
* The sidebar provides primary module navigation**.** It's typically persistent and can be collapsible for better screen usage.
* Use vertical layout.
* Group related navigation items.
* Show icons alongside text for clarity.
* Highlight the active route visually.

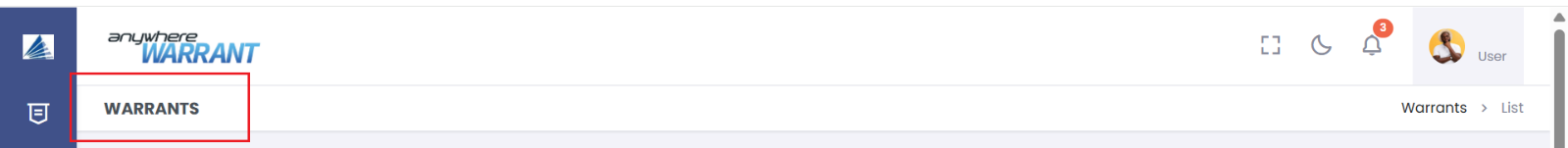
A screenshot of a calendar

AI-generated content may be incorrect.

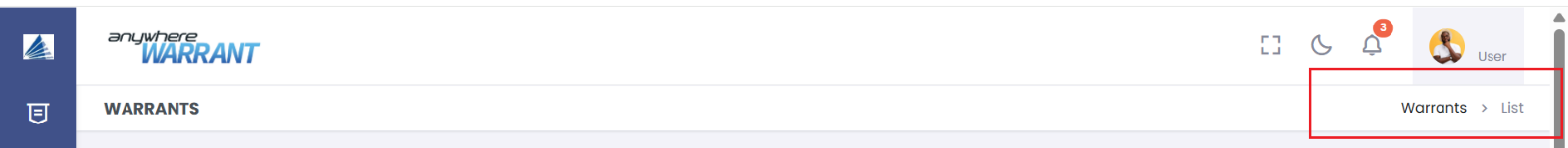
* **Active State Indicators**
* Shows users where they are in the application. Improves orientation and usability.
* Apply a distinct background color, bold text, or an indicator bar.
* Automatically highlight based on the current route/path.

# Page Header

* **Page Title**
* The **Page Header** is the top section of a web page that immediately informs the user of their current location within the application and provides context for the page’s purpose.
* It should be **clear**, **descriptive**, and **contextual**, accurately representing the page's content.
* Place it left-aligned within the page header section.
* Keep the font size larger than body text (e.g., text-2xl or 24px).
* Capitalize consistently (e.g., Title Case).
* Example: “USER PROFILE”



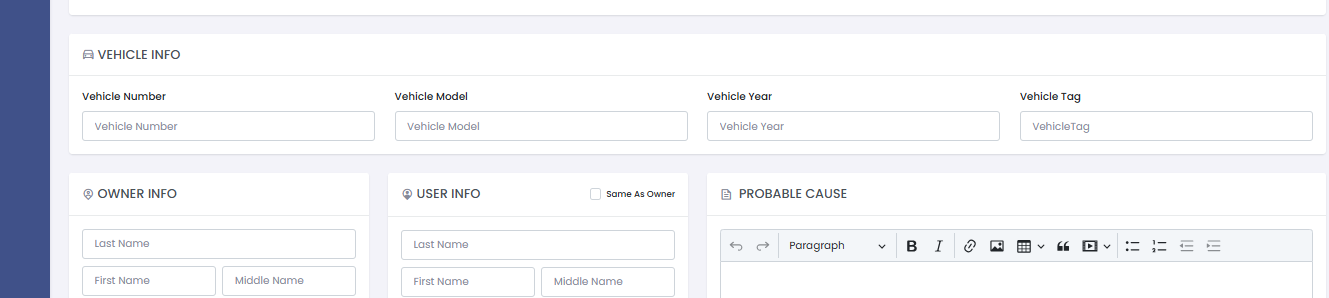
* **Breadcrumbs**
* Breadcrumbs are a navigational aid showing the user's path in the hierarchy.
* Place below or beside the page title (typically top-right of the page).
* Link all breadcrumb items **except** the current page.
* Use smaller font size and muted color
* Example: “Home > Users > John Doe”



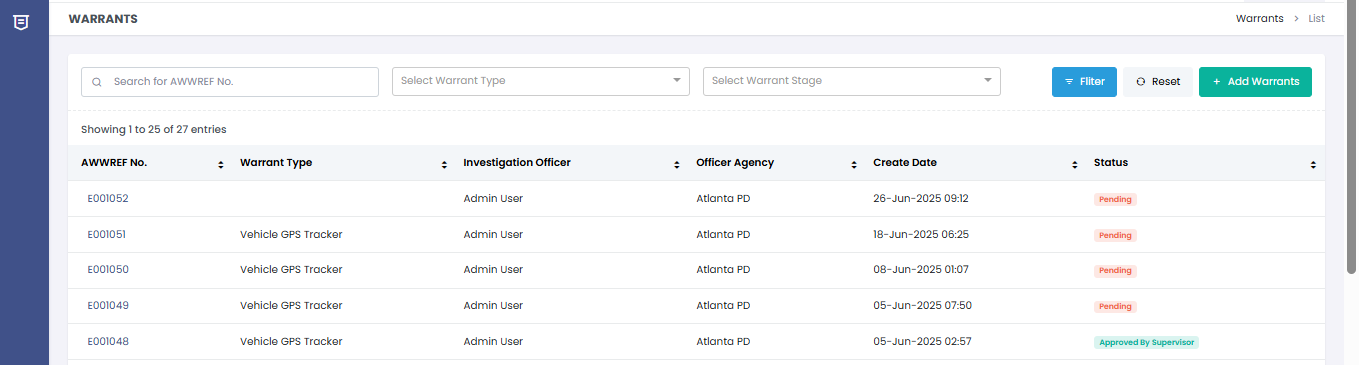
# Main Content Area

The central area where page-specific information is displayed. Its structure depends on the page's purpose but follows modular patterns:

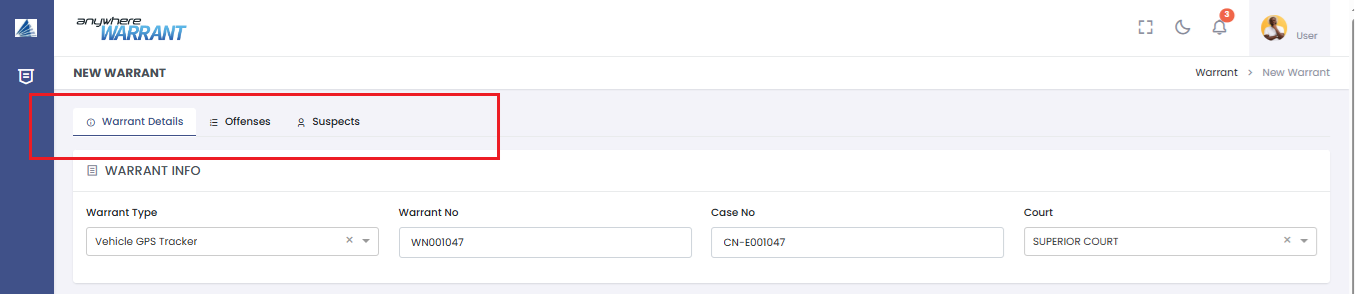
* **Section Headers**
* Clearly label content areas (e.g., “User Details”).
* Use consistent font size and weight (e.g., text-xl font-semibold).
* Separate sections with spacing or dividers.
* **Cards/Panels**
* Group related data visually.
* Use padding, border, and subtle shadows.
* Can contain text, tables, forms, etc.
* **Forms or Inputs:**
* Use standard spacing and labels.
* Include required indicators and inline validation



* **Tables/Lists**
* Ideal for structured, itemized data.
* Must support sorting, filtering, pagination.
* Add hover and active row indicators.



* **Tabs (optional):**
* Useful for switching between subviews or categories.



# Actions & CTAs (Call to Actions)

* **Primary Buttons:**
* One per page or card (e.g., “Save”, “Submit”).
* Use brand/theme primary color consistently.

A screenshot of a computer

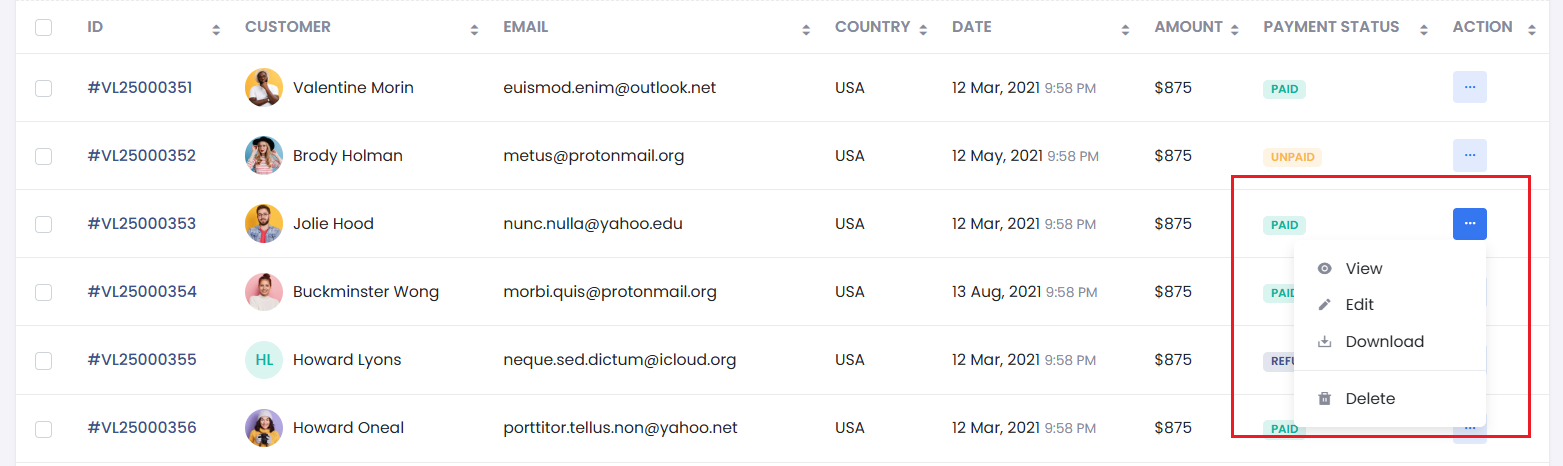
AI-generated content may be incorrect.

* **Secondary Buttons**
* For secondary or cancel actions.
* Use a neutral color like gray.

A screenshot of a computer

AI-generated content may be incorrect.

* **Dropdown Menus or Action Menus**
* Useful in tables or cards for multiple actions (Edit, Delete, View).



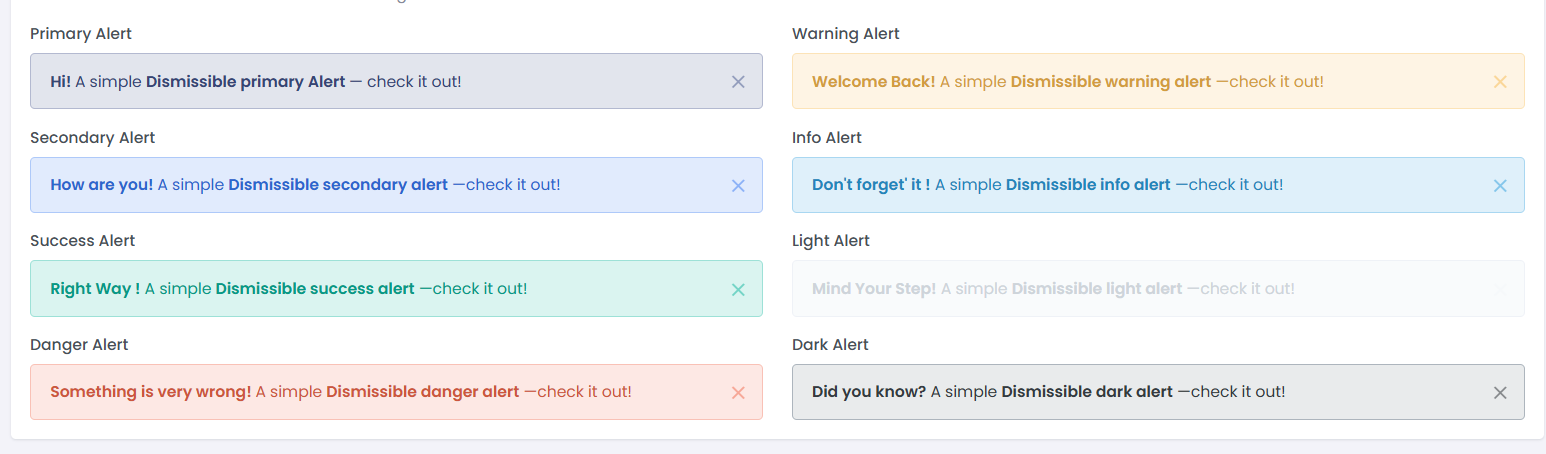
# Feedback Elements

* **Loading Indicators/Spinners**
* Show while data is being fetched or action is in the process.

A group of icons on a white background

AI-generated content may be incorrect.

* **Toast/Alert Messages**
* Provide feedback after actions (success, error, or warning). It should be auto dismissed or closable.



* **Validation Messages**
* Display below the inputs.
* e.g., “This field is required”

A screenshot of a computer screen

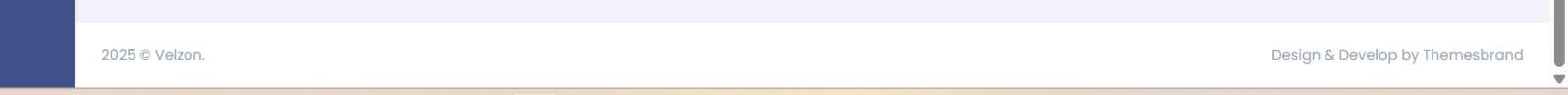
AI-generated content may be incorrect.

# Spacing and Layout

* **Consistent Padding and Margins**
* Use spacing tokens (e.g., p-4, mb-6, space-y-4) to avoid clutter.
* **Grid System**
* Use CSS Grid or Flexbox for layout.
* Example: Two-column layout on desktop, single column on mobile
* **Whitespace**
* Allow space around groups of elements to improve scanning and readability.

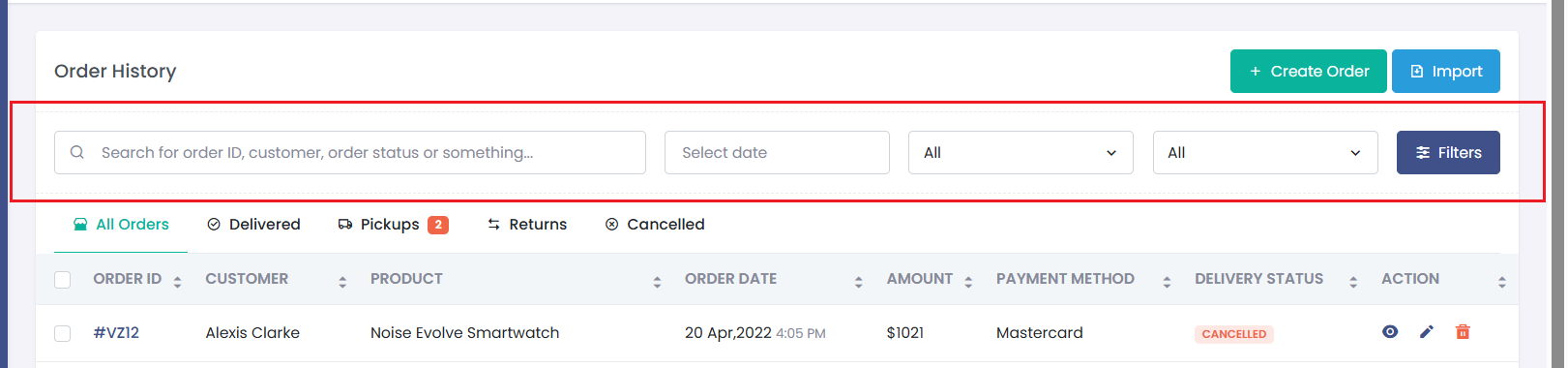
# Footer (Optional in App Pages)

* Can be used to include to add legal info or version numbers.

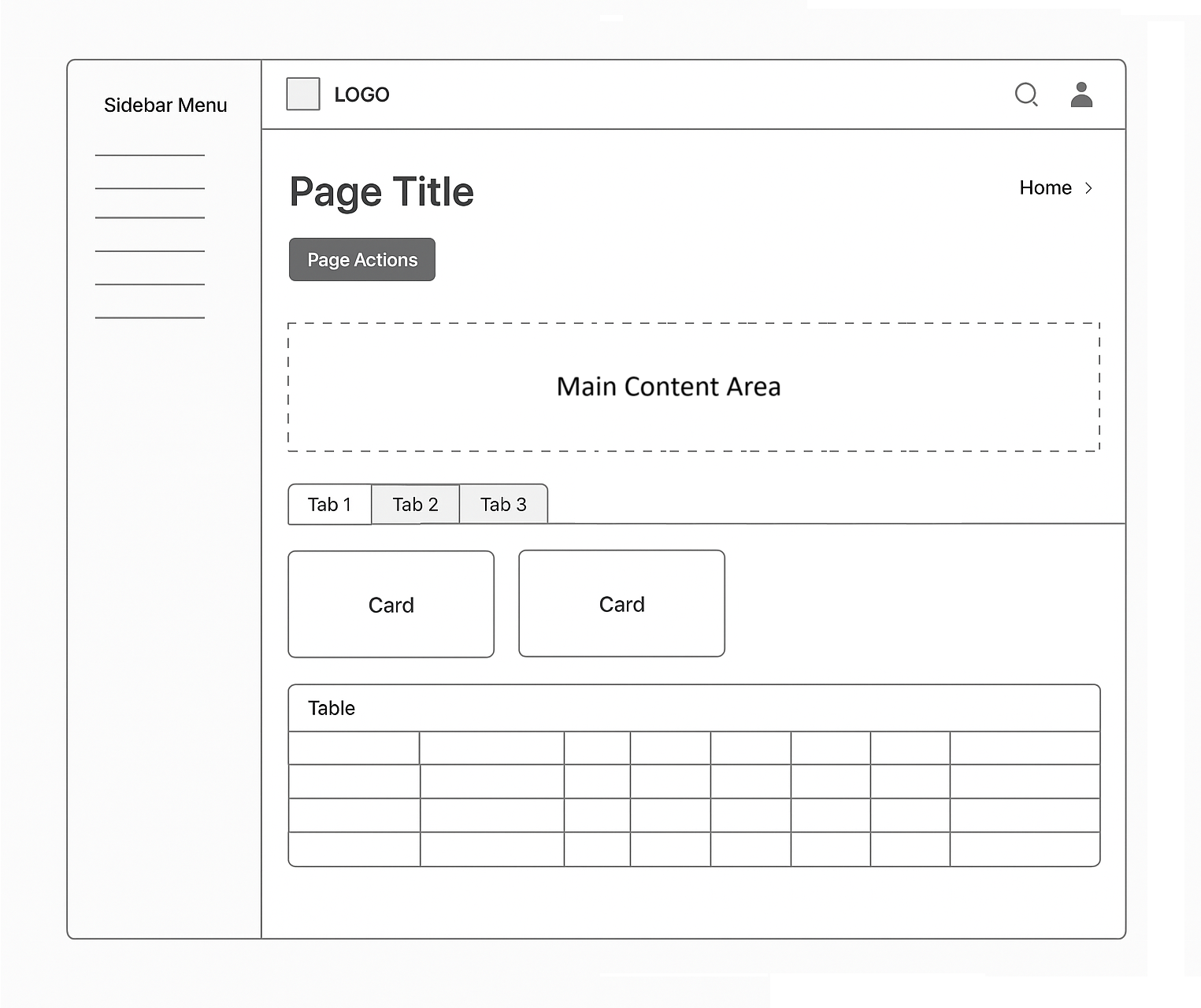


# Search & Filters (if applicable)

* For pages with large data sets (e.g., user lists, reports, logs).
* Should be placed at top of data.
* Include advanced filters or tag-based filters if necessary.



# **Example: Page Layout** **High-Level -**



A screenshot of a computer

AI-generated content may be incorrect.

# **Do’s and Don’ts**

**✅ Do**

* Use a consistent theme for all UI elements including fonts, buttons, labels, and tables.
* Follow a standardized color palette for buttons (e.g., primary, secondary, danger).
* Use page titles and breadcrumbs to clearly indicate the user's location and page purpose.
* Structure content using cards or panels with proper headers for each module.
* Align content in grid layouts with consistent margins and padding.
* Make primary table columns clickable (e.g., Order ID, Name) to access detailed views.
* Include dropdown action menus (3-dot menu) in rows for secondary actions.
* Implement a left sidebar for primary navigation with icons and labels.
* Ensure active state highlighting in both the sidebar and topbar for better orientation.
* Use theme-aligned labels, inputs, and validation messaging.
* Show inline validation messages and required indicators.
* Present complex data using tabs to switch between related categories or sub views.
* Use toast or alerts for confirmation, success, and error messages.
* Add loading indicators to show action progress.
* Use Modal Popups only to show read only data or form submission with 2 or 3 fields.

**❌ Don’t**

* Mix or overrides of custom styles with theme components.
* Apply custom fonts, sizes, or unstyled elements - they break visual consistency.
* Overload rows or headers with multiple buttons or icons.
* Use Modal Popups for lengthy form submission.
* **UI/UX Design Principles and Best Practices**

# Clarity and Simplicity

* **User-Centered Design:**  
  Begin with user research—identify your users’ needs, pain points, and behavior patterns. Design flows and interfaces around real user goals, not just business logic.
* **Minimalistic Design:**  
  Remove unnecessary elements. Prioritize content and actions that users care about most. Use whitespace to guide focus and reduce visual noise.
* **Readability:**  
  Use legible fonts with proper sizing and line height. Stick to a maximum of 2–3 font families. Ensure text color contrasts well with its background.
* **Line Length:**  
  Keep line lengths between 50–75 characters per line for body text. This prevents eye strain and improves reading speed.
* **Text Hierarchy:**  
  Use a clear typographic scale:
* H1 for page titles
* H2 for section headings
* H3 or bold text for subpoints
* Standard body text for content  
  This helps users scan and understand content faster.

# Consistency

* **Visual Consistency:**

Use a predefined design system or theme. Apply the same color palette, icon style, typography, and button appearance throughout the app.

* **Behavioural Consistency:**Ensure interactions behave the same way everywhere. Actions, such as button clicks or gestures, should behave in a consistent way across the application.

# Color and Contrast

* **Follow Brand Guidelines:**  
  Use brand-approved colors for primary buttons, highlights, and headers. Align the color scheme with brand identity (logo, marketing material).
* **Contrast:**   
  Ensure sufficient contrast between foreground (text) and background for readability.
* **Limit Palette**
* Stick to 3–5 core colors:

**Primary:** For buttons, highlights

**Secondary:** Accents, badges, or alternative actions

**Neutral:** Grays for structure

**Text:** Black or dark gray

**Background:** White/light

* **Semantic colors**:
* ✅ Green: Success
* ❌ Red: Errors
* ⚠️ Yellow/Orange: Warnings
* ℹ️ Blue: Info
* **Support Dark Mode (Optional but Valuable)**
* Plan a dark theme with proper inverted colors and consistent tone.
* Use CSS variables for color themes
* e.g., --primary-color, --text-color

# Accessibility

* **Keyboard Navigation:**  
  All interactive elements (links, buttons, form fields) must be reachable via the Tab key and activated with Enter or Space.
* **Alt Text for Images:**  
  Provide meaningful alt attributes for all informative images. Decorative images should have alt="" to be ignored by screen readers.
* **Tab Indexing:**  
  Use the tabindex attribute responsibly to define the focus order, ensuring logical and intuitive navigation, especially in custom components or modals.

# Interaction

Users should always understand what is happening in response to their actions. Lack of feedback leads to confusion or repeated inputs.

* **Instant Feedback:**  
  Show visual cues like button states (pressed, loading), or display messages after form submissions or actions.
* **Error Handling:**  
  Use **inline error messages** near the field with the issue. Be specific and actionable:

“Email is required” is better than “Invalid input.”

* **Loading Indicators:**  
  Display spinners, progress bars, or skeleton loaders when data is being fetched or processes are in progress.

# Responsive Design

Users access apps on a variety of devices. Your interface should adapt seamlessly across screen sizes and orientation.

* **Flexible Layouts:**  
  Use CSS Grid or Flexbox to create layouts that adjust based on screen width.
* **Mobile-First Design:**  
  Design for smaller screens first, then scale up. Ensure content stacks vertically and inputs/buttons remain touch-friendly.
* **Orientation Handling:**  
  The interface should work correctly when switching between portrait and landscape on phones and tablets.

# Performance and Efficiency

* **Fast Load Times:**  
  To achieve Fast load Times, try to compress images, lazy-load content, and minimize third-party scripts. Use browser caching and CDN services.
* **Smooth Transitions:**  
  Use subtle animations for modal openings, tab transitions, and dropdowns to enhance interactivity—but avoid overly complex or distracting effects.