

Plan to make Tailwind layouts consistent

Table of Contents

Plan to make Tailwind layouts consistent and fast to ship.....	1
- Foundations (today).....	1
- Reusable UI primitives (Blade components).....	1
- Migration approach (page by page, low risk).....	1
- Consistency guardrails.....	2
- Timeline.....	2
- Why this over migrating to Bootstrap.....	2

Foundations (today)

- Tailwind tokens: unify spacing, radii, colors via `tailwind.config.js`.
- Prettier + Tailwind plugin: enforce class order and formatting across views.
- Short “UI conventions” doc: 1-page checklist for responsive patterns and actions.

Reusable UI primitives (Blade components)

- ``x-page.header``: back-chevron + title (+ optional subtitle/actions).
- ``x-filters.panel``: mobile-collapsible, desktop-visible filter container.
- ``x-table.wrapper``: ``overflow-x-auto`` + ``min-w-max`` table shell with consistent head/body paddings and alignments.
- ``x-list.card``: mobile card item with slots for title, meta, and footer actions.
- ``x-form.actions``: stacked buttons on mobile, inline on sm+ with consistent spacing.
- Button utilities: ``.btn``, ``.btn-primary``, ``.btn-edit``, ``.btn-delete``, etc. implemented once with ``.@apply`` and reused everywhere.
- Location:
 - Blade: ``resources/views/components/...``
 - CSS utilities: ``resources/css/utilities/buttons.css`` (imported in `app.css`)

Migration approach (page by page, low risk)

- Priority 1 (Orders): index (done), create (done), edit (done), show (next).
- Priority 2 (Sales): index/create/show.
- Priority 3 (Inventory): index/low-stock.
- Each page:
 - Replace ad-hoc headers with ``x-page.header``.
 - Wrap filters in ``x-filters.panel``.
 - Use ``x-table.wrapper`` for desktop tables.
 - Add ``x-list.card`` for mobile lists and hide table on mobile.
 - Replace raw action buttons with ``.btn*`` utilities or ``x-form.actions`` where relevant.

Consistency guardrails

- PR checklist (Definition of Done):
 - Uses `x-page.header` (or explains why not).
 - Filters are collapsible on mobile, visible on desktop.
 - Mobile cards present; desktop tables have overflow wrappers.
 - Actions are reachable on mobile; buttons use `.btn*` utilities.
- Pre-commit formatting with Prettier + Tailwind plugin.

Timeline

- Day 1: Add components/utilities + retrofit Orders Show; update docs.
- Day 2–3: Retrofit Sales and Inventory priority screens.
- Ongoing: Convert remaining lists as we touch them.

Why this over migrating to Bootstrap

- No framework switch risk/cost; we keep Tailwind only as you prefer [[memory:8676888]].
- We still deliver an AdminLTE-like, modern look via shared components [[memory:8676899]].
- Faster, safer path to consistency without sweeping rewrites.

Want me to start by implementing the base components and retrofitting Orders Show now?