

Komponenty

Komponenty

Architektura UI opiera się na komponentach shadcn/ui (headless + Tailwind), rozszerzonych o własną logikę raportową.

Hierarchia komponentów

```
main.tsx
├─ ThemeProvider      # kontekst motywu (dark/light/system)
  └─ TooltipProvider  # globalny provider tooltipów (Radix)
    └─ App            # główna logika: selektor listy/miesiąca, sekcje raportu
      ├── DarkModeToggle
      ├── ReportTable (T1)
      ├── KpiMonthlyTable (T2)
      └─ ReportTable (T5)
```

ThemeProvider

Plik: `src/components/theme-provider.tsx`

Zarządza motywem całej aplikacji.

API

```
// Odczyt i zmiana motywu w dowolnym komponencie:
const { theme, setTheme } = useTheme()

setTheme("dark")    // zawsze ciemny
setTheme("light")   // zawsze jasny
```

```
setTheme("system") // podążaj za OS
```

Zachowania

- Motyw persystowany w `localStorage` (klucz: `"theme"`)
- Przy starcie odczytuje zapisaną wartość; brak → `"system"`
- Reaguje na zmianę `prefers-color-scheme` w czasie rzeczywistym (gdy `theme === "system"`)
- Reaguje na `StorageEvent` — synchronizacja między zakładkami
- **Skrót klawiszowy** `D` — przełącza dark/light bez użycia myszki (działa gdy focus nie jest na polu tekstowym)
- Przełączenie motywu tymczasowo wyłącza CSS transitions (brak migania)

DarkModeToggle

Komponent `Switch` powiązany z `useTheme`. Wyświetla aktualny stan i pozwala go przełączyć kliknięciem.

ReportTable (T1, T5)

Tabela wolumenowa/YTD renderowana przez **TanStack Table v8**.

Props / dane wejściowe

- Dane: `ReportRow[]` ładowane z pliku JSON (lub hardcoded `reportData` w T1)
- Kolumny generowane dynamicznie na podstawie miesięcy (z `MONTHS`)
- Każda komórka może mieć `highlight` (kolor tekstu) lub `highlightBg` (tło)

Funkcje ID

Funkcja	Opis
<code>getDisplayRowId(label, fallback)</code>	Mapuje rok na alias: <code>2026→TY</code> , <code>2025→LY</code> , <code>2024→AY</code> , <code>vs→VS1/VS2</code>
<code>getBaseYearTwoDigits(label, fallback)</code>	Zwraca dwuznakowy prefiks dla ID komórki
<code>getT1MonthCellId(base, monthId)</code>	Buduje ID komórki: <code>TY+02→TYLM</code> , <code>TY+03→TYTM</code> , <code>TY+04→TYNM</code>

Atrybuty DOM

Atrybut	Przykład	Opis
---------	----------	------

data-table-id	data-table-id="T1"	Na kontenerze sekcji
data-row-id	data-row-id="TY"	Na wierszu
data-cell-id	data-cell-id="TYLM"	Na komórce

KpiMonthlyTable (T2)

Tabela KPI z 13 stałymi kolumnami miesięcznymi i 11 wskaźnikami.

Kolumny miesi?czne

Zdefiniowane jako stała KPI_MONTH_COLUMNS — od Mar 2026 wstecz do Mar 2025. Kolumna Mar 2026 (bieżący miesiąc) wyświetla zawsze wartość TM (placeholder — dane live).

Props KpiMonthlyTable

Prop	Typ	Domyślnie	Opis
data	KpiRow[]	kpiMonthlyData	Dane tabeli
showIds	boolean	false	Pokazuje techniczne ID kolumn i wierszy
hidePercent	boolean	false	Ukrywa znak % w wartościach
hidePln	boolean	false	Ukrywa sufiks zł

KpiLabelCell

Komponent renderujący etykietę wskaźnika z kolorowaniem składni:

- **tekst główny** — font-medium
- **(opis w nawiasach)** — kolor text-syntax-opisy
- **[slug w nawiasach kwadratowych]** — font-mono, kolor text-syntax-slug

CellContent

Wspólny komponent komórki dla T1/T5:

```
<CellContent cell={{ value: "88 045", highlight: true }} hidePercent={false} />
```

- `highlight: true` → `text-amber-500 dark:text-amber-400`
- `highlightBg: true` → `bg-amber-500/15 dark:bg-amber-500/20 rounded px-1`
- `hidePercent: true` → usuwa znak `%` z wartości

Komponenty UI (shadcn)

Wszystkie w `src/components/ui/`. Dodane przez CLI `npx shadcn@latest add <nazwa>`.

Plik	Komponent	Użycie
<code>table.tsx</code>	<code>Table</code> , <code>TableRow</code> , <code>TableCell</code> , <code>TableHead</code> , <code>TableHeader</code> , <code>TableBody</code>	Tabele T1, T2, T5
<code>switch.tsx</code>	<code>Switch</code>	Dark mode toggle
<code>checkbox.tsx</code>	<code>Checkbox</code>	Filtrowanie wierszy
<code>button.tsx</code>	<code>Button</code>	Akcje UI
<code>tooltip.tsx</code>	<code>Tooltip</code> , <code>TooltipContent</code>	Podpowiedzi
<code>badge.tsx</code>	<code>Badge</code>	Oznaczenia
<code>card.tsx</code>	<code>Card</code>	Karty sekcji
<code>dropdown-menu.tsx</code>	<code>DropdownMenu</code>	Menu kontekstowe

Utilities

`cn()` — `src/lib/utils.ts`

```
import { cn } from "@lib/utils"
```

```
cn("base-class", condition && "conditional-class", "another-class")  
// łączy klasy przez clsx, usuwa konflikty przez tailwind-merge
```

`formatRowIndexId(index)`

Zamienia indeks wiersza na literę Excel: `0→A`, `1→B`, ..., `25→Z`, `26→AA`.

Revision #5

Created 2026-05-28 04:01:03 CEST by matzxp84

Updated 2026-05-28 04:34:07 CEST by matzxp84