

# Virtual Theme Manager — Használat TL;DR

Gyors referencia. Teljes verzió: [USAGE-GUIDE.md](#) | Beépítés: [INTEGRATION-GUIDE.md](#)

## Hozzáférés

```
const vtm = (this.getOwnerComponent() as Component).getVirtualThemeManager();
```

## Témák

Kulcs	Leírás
normal	Alapértelmezett (világos)
normal_branded	Zöld branded
warning	Sárga figyelmeztetés
alarm	Piros riasztás (sötét)
nightshift	Sötét mód
nightshift_dimmed	Extra sötét

## API

```
vtm.switchTheme("alarm")      // váltás → true/false
vtm.switchTheme("normal")     // vissza
vtm.getActiveThemeKey()       // aktuális kulcs
vtm.getThemes()              // összes téma
```

## Leggyakoribb minták

```
// Hiba → piros → vissza
vtm.switchTheme("alarm");
MessageBox.error(err.message, { onClose: () => vtm.switchTheme("normal") });

// Figyelmeztetés 5 mp-re
vtm.switchTheme("warning");
setTimeout(() => vtm.switchTheme("normal"), 5000);

// WebSocket
if (oMessage.command === "THEME_CHANGE") vtm.switchTheme(oMessage.data.themeKey);
```

## Tudnivalók

- Nem perzisztens — újratöltés = `normal`
- `switchTheme()` idempotens
- Egyszerre egy téma aktív
- `getActiveThemeKey()` a virtuális kulcsot adja, nem a base theme-t