```
module: 31
title: Branding / Theme
phase: 1
status: detail-spec-draft
last_updated: 2026-05-08
session_source: brainstorming-session-2026-05-07-2206.md
related_lps: [LP-9]
spec_style: requirements-pur
```

Modul 31 — Branding / Theme

> ***Detail-Spec gemäß LP-9.*** *Phase 1 minimal: nur Light/Dark-Mode pro User. Voll-Branding (Logo, Farben, White-Label) ist Phase 2. Modul ist in Phase 1 bewusst fast leer.*

Themenblöcke

```
A. Datenmodell (Phase 1)
B. Routes & Settings (Phase 1)
C. Theme-Auflösung (System/Light/Dark)
D. Permissions
E. Phase 2 / spätere Features
```

Block A — Datenmodell (Phase 1)

A.1 — User.theme_preference

Bestehendes/zu ergänzendes `User`-Feld:
- `theme_preference` (Enum: `system | light | dark`, Default `system`)

Anwendung:
- `system` → folgt OS-/Browser-Präferenz (`prefers-color-scheme`)
- `light` → erzwingt Light-Theme
- `dark` → erzwingt Dark-Theme

Persistierung:
- Server-side in `User.theme_preference`
- Client-side Cache in Cookie/LocalStorage für SSR-Konsistenz (verhindert Flash-of-Wrong-Theme)

A.2 — Keine weiteren Tabellen in Phase 1

In Phase 1 wird **kein** `TenantBranding`, **kein** Logo-Upload, **keine** Tenant-Farb-Tabelle angelegt. Voll-Datenmodell siehe Phase 2 (Block E).

Block B — Routes & Settings (Phase 1)

B.1 — User-Profile-Settings (`/{tenant}/settings/profile/`)

Theme-Auswahl:
- Toggle/Picker mit drei Optionen: `System` / `Hell` / `Dunkel`
- Sofortige Anwendung ohne Reload
- Persistierung in `User.theme_preference`

Klartext:
- Option `System` : *„Folgt der Einstellung deines Geräts"*
- Option `Hell` : *„Helles Design"*
- Option `Dunkel` : *„Dunkles Design"*

B.2 — Keine Tenant-Admin-Route in Phase 1

`/{tenant}/admin/branding/` existiert in Phase 1 **nicht**. Erst Phase 2 (siehe Block E).

Block C — Theme-Auflösung

C.1 — Auflösungs-Reihenfolge

Beim Rendern einer User-Session:

1. Wenn User eingeloggt + `User.theme_preference = light` → Light
2. Wenn User eingeloggt + `User.theme_preference = dark` → Dark
3. Sonst (`system` oder nicht eingeloggt): Browser-Media-Query `prefers-color-scheme`
4. Fallback: Light

C.2 — Live-Reaktivität bei `system`

- Wenn User `theme_preference = system` und OS-Theme wechselt während der Session, soll Theme live mitwechseln
- Listener auf `matchMedia('(prefers-color-scheme: dark)')` Change-Event

C.3 — SSR-Konsistenz

- Server rendert mit Theme-Klasse auf `<html>` basierend auf Cookie/Header
- Verhindert Flash-of-Wrong-Theme beim Hydrate

C.4 — Anwendung in App und Capacitor

- Web: über CSS-Variablen + Theme-Class auf `<html>` (Implementierungs-Detail Tailwind- `dark:` -Variants)
- Capacitor (Modul 36): native Status-Bar / Splashscreen müssen Theme respektieren (Native-Hooks setzen)

Block D — Permissions

Permission-IDs

Phase 1 hat **keine** branding-spezifischen Permissions. User kann eigenes Theme jederzeit setzen.

Phase 2: `branding.admin` als eigene Permission, gegated über `tenant.admin` für Tenant-weite Branding-Settings.

Klartext-Fehler

In Phase 1 keine relevanten Fehler-Cases.

Phase 2 / spätere Features

E.1 — Tenant-Branding (Standard, alle Pläne)

Datenmodell — neue Tabelle `TenantBranding`:
- `tenant_id` (PK, FK)
- `logo_file_id` (FK auf File, optional)
- `primary_color` (Hex-String, validiert)
- `accent_color_1` (Hex-String, optional)
- `accent_color_2` (Hex-String, optional)
- `tenant_display_name` (überschreibt System-Default in UI, optional)
- `updated_at`, `updated_by_user_id`

Routes:
- `/{tenant}/admin/branding/` — Verwaltungs-Seite

Funktionen:
- Logo-Upload (PNG/SVG, max-Größe definiert)
- Farb-Picker für Primärfarbe + 1-2 Akzentfarben mit Live-Preview
- Tenant-Name-Override für UI-Anzeige
- Branding-Anwendung in: Sidebar/Header-Logo, Email-Header, Customer-Portal, Login-Screen, Capacitor-Splashscreen
- Reset-auf-Default-Funktion

Permission: `branding.admin` (über `tenant.admin`)

E.2 — Outbound-Email-Templates mit Branding

- System-Email-Templates erben Tenant-Logo + Primärfarbe
- Header-/Footer-Bereiche im Template-System branding-bewusst
- Marketing-Emails (Modul 10) übernehmen Branding standardmäßig, optional override

E.3 — Custom-Login-Screen

- Tenant-Logo + ggf. Hintergrund-Bild auf `/login` wenn Tenant-Subdomain/-Slug erkannt
- Phase 2 für alle Pläne

E.4 — Custom-Splashscreen für Capacitor-App

- Tenant-Logo als Splashscreen in iOS/Android-Builds
- Generierung pro Tenant über Build-Pipeline (Implementierungs-Detail) oder dynamisches Theming beim App-Start

E.5 — White-Label-Modus (Enterprise only)

Funktionen:
- Eigene Domain (`crm.geerkens.de` als CNAME zu `app.hivegrid.de/{tenant}`)
- Eigene Fonts (Upload von Webfonts oder Google-Fonts-Auswahl)
- Vollständig anpassbares Theme (alle Farb-Variablen, nicht nur Primary/Accent)
- Eigenes Email-Template-System (Tenant baut Templates selbst inkl. HTML)
- HiveGrid-Branding entfernt (Footer, Email-Footer, Login-Page)
- Custom-Domain-Workflow inkl. Cert-Management (Caddy-Auto-HTTPS)

Permission: `branding.white_label` — nur in Enterprise-Plan via Feature-Gate (Modul 27, Block F)

Datenmodell-Erweiterung `TenantBranding`:
- `custom_domain` (String, validiert)
- `custom_font_file_ids[]`
- `theme_variables` (JSONB — alle CSS-Variablen)
- `email_templates_custom` (Boolean Toggle)
- `hide_hivegrid_branding` (Boolean Toggle)

E.6 — Branding-Preview vor Live-Schalten

- Tenant-Admin sieht Vorschau vor Anwendung
- Apply-Button schaltet live für alle User des Tenants

Architektur-Vorschriften (Querschnitt)

- **LP-9 — Spec-Tiefe:** Phase 1 minimal gehalten, Phase 2 explizit umrissen
- **AV #4 — Alle Strings extern:** Tenant-Name-Override aus DB, nicht hartkodiert (relevant für Phase 2)
- **LP-13 — EU-Stack:** Logo-Storage über Scaleway-EU (Modul 23)

Detail-Spec-Status: DRAFT

Modul ist in Phase 1 bewusst minimal. Phase-2-Features sind grob umrissen für Roadmap-Klarheit; Detail-Spec für Phase 2 erfolgt vor Phase-2-Implementierung.

Offene Detail-Punkte (nicht-blockierend):
- Konkrete CSS-Variable-Namen (Implementierungs-Detail Tailwind-Config)
- SSR-Cookie-Name für Theme-Persistierung
- Capacitor-Native-Hook für Status-Bar-Theming (Implementierungs-Detail)

Abhängige Module:
- Modul 27 (Auth) — `User.theme_preference`-Feld
- Modul 36 (Mobile/Capacitor) — Native-Status-Bar/Splashscreen-Theming
- Phase 2: Modul 23 (Files) — Logo-Upload-Storage
- Phase 2: Modul 27 (Auth, Block F) — Feature-Gate für White-Label-Enterprise