HiveGrid 2.0

What This Is

HiveGrid 2.0 ist eine Multi-Tenant-SaaS-Plattform für SHK-Handwerksbetriebe (Sanitär, Heizung, Klima). Sie bündelt CRM, Vertriebs- und Auftragspipeline, Team-Chat, visuelle KI-Automation und die Anbindung an das KWP-ERP in einem EU-gehosteten System.

HiveGrid ist ein Multi-Kunden-SaaS für den gesamten SHK-Markt. Die **Geerkens GmbH** (Kontakt: Noah) ist erster Kunde und Investor — Validierungspartner, nicht alleiniger Zweck. Das Produkt ersetzt dort HubSpot (CRM + Sales-Pipeline), Quiply (Team-Chat), Zapier/n8n (Automation) sowie mehrere Excel-Prozesse. Die Codebase ist ein Git-Downstream der ToolCTRL-Boilerplate (~35–45 % Abdeckung: Fundament high, Fachdomäne greenfield).

> 🔴 ***MODELL-PIVOT (2026-06-13):*** *CRM/Pipeline-Modell ist **Projekt-zentrisch** — ein `Projekt`-Kern, Pipelines = frei konfigurierbare Views; „Deal"/„Auftrag" sind Pipeline-Typen, keine eigenen Entitäten. Single Source: `.planning/ENTITY–MODEL.md`. ENTITY-MODEL/REQUIREMENTS/SCREEN-MAP/PHASE-SOURCES werden gegen den Anker neu gefasst (ER-Modell wird gemeinsam gebaut).*

Core Value

Jeder SHK-Betrieb (Mandant) wickelt die komplette Wertschöpfungskette — **Lead → Projekt (frei konfigurierbare Pipelines: Sales/Auftrag/…) → KWP-Sync (ab v2)** — inklusive visueller KI-Automation in einem einzigen, DSGVO-konformen EU-System ab. Wenn diese Kette nicht durchgängig funktioniert, scheitert das Produkt.

Requirements

Vollständige Liste mit 79 v1-Requirements über 36 Module: siehe `.planning/REQUIREMENTS.md`. Detaillierte Acceptance-Kriterien je Requirement: `.planning/intel/requirements.md` + Modul-Specs unter `_bmad-output/specs/`.

Validated

(Noch keine — Big-Bang-Launch ohne phasiertes Produktiv-Go-Live, AD-19. Validierung erfolgt pro Modul über Noah/Geerkens via PROGRESS-TRACKER-Klick-Pfad.)

Active

USP-Säulen (Differenzierung, volle Tiefe day-1 — AD-16):

- [ ] Automation-Engine (M17) — visueller Editor, alle Trigger/Aktionen, 24h-Undo, Guards
- [ ] KI-Layer (M18) — EU-Provider, RAG, Sprach-zu-Automation, Pay-as-you-go
- [ ] KWP-Integration (M19) — bidirektionaler Sync mit dem SHK-ERP KWP (Kernsystem des Erstkunden)

SHK-Kernpfad (AD-11, Projekt-zentrisch nach AD-20):

- [ ] CRM/Kontakt-Kern (M01), Lead-Inbox/LeadInput (M02), **Projekt & Pipelines** (vormals Sales M03 + Auftrag M04 → ein `Projekt`-Kern + konfigurierbare Pipeline-Views; Abnahme bleibt v1 als Checklisten-Sub-Area, Rechnung = Sub-Entity v1)

Fundament-Kern:

- [ ] Auth/Multi-Tenancy (M27), Apps-Engine/Adapter (M33a), Audit-Log (M21), Notifications (M22), i18n (M30), Feature-Flags (M32), Plattform-Admin (M37)

Breite (AD-13):

- [ ] HR-Hub (M14), Customer-Portal (M15), Email (M09/M10), Telefonie (M12), Team-Chat (M13), Booking/Forms/Angebot (M05/M06/M07), Files/Suche/Calendar/Tasks (M23/M24/M25/M26), Billing (M28), Onboarding (M29), Branding (M31), Dashboard (M20), Public-API (M34), DSGVO-Export (M35), Mobile-PWA (M36 — PWA-only, kein nativer Client), Help-Center/Desk (M38/M39)

Out of Scope

- **Phase-2-Module (z.B. M11 WhatsApp)** — nicht spec'd, späterer Release; von M13/M18 referenziert, aber nicht im Ingest-Set
- **Konkrete Pricing-Stufen** — Billing-Engine wird gebaut, Preise geparkt; Geerkens auf Sonderkondition/Trial (AD-15)
- **Volle Migrations-Pipelines pro Altsystem** — sauberer Start, KWP-Sync füllt nach; HubSpot/Quiply als Read-only-Archiv, nur punktuelle CSV-Importe (AD-14)
- ~~EU-Email-Provider (AD-6) / Captcha (AD-7)~~ — **entschieden 11.06.: Mailjet + mCaptcha** (Boilerplate-verdrahtet)
- **KWP Windows-Sync-Programm (M089, alte M19-Variante)** — entfällt durch offizielle KWP-API; M089–M095 vor Umsetzung neu schneiden

Context

- **SHK-Domäne:** Geerkens = Heizungsanlagen/Wartung/Reparatur. Entitäten: B2C/B2B-Kunden mit Anlagen-Zustand (Custom-Felder), mehrstufige Anlagen-Verkäufe, Wartungsverträge, Reparatur-Aufträge. Monteure arbeiten offline auf Baustellen (AD-17).
- **SHK-Seeds beim Tenant-Anlegen:** 10 Standard-Rollen + Permission-Bundles, SHK-Sales-Pipelines, Custom-Felder für Heizungsanlagen, Projekt-Pipelines, Automation-Templates, Onboarding-Branchen-Zweig — deklarativ, von Noah iterativ validiert (AD-12).
- **Boilerplate-Reuse:** Geteilte Pakete behalten `@toolctrl/*`-Namen (`packages/ui-app`, `ui-subframe`, `ui-mock`, `@toolctrl/search`, `@toolctrl/foundation`). Neue HiveGrid-Querschnittsdienste im `@hivegrid/*`-Namespace (activity, audit, notifications, entitlements, search, i18n).
- **Phase-0-Pflichten (vor jedem Fachmodul):** (1) Boilerplate als Code-Basis einziehen — Repo ist heute Docs-only, (2) Naming-Drift beheben (AD-10), (3) EU-Sterilisation (GitHub/Google-OAuth-Reste raus), (4) M00 + 4 Querschnitts-Neubauten (logActivity, Search-Registry, Subframe-Adapter, Mass-Action-Schutz), (5) fehlende Dependencies ergänzen.
- **Spec-Stil:** Alle 36 Specs sind „requirements-pur" (benötigte UI-Elemente als Liste, keine Layouts — Designer hat freie Hand) und im Status `detail-spec-draft`. Vor Modulbau: Noah-Verifizierung, Index-/Migrations-SQL, SHK-Defaults finalisieren, Subframe-UI-Designs.

Constraints

- **Mandanten-Trennung**: Echtes Postgres-RLS als bauliche Vorgabe (USING/WITH CHECK, `SET LOCAL app.tenant_id`, App-Rolle ohne BYPASSRLS) — AD-1
- **Datenmodell**: Typisierte First-Class-Prisma-Tabellen; Boilerplate-Entity-Builder (Entity/Row/RowValue) wird entfernt; Custom-Felder über `CustomFieldDefinition` + `CustomFieldValue` — AD-2
- **EU-Stack (keine US-Kern-Abhängigkeit)**: Meilisearch (zentral auf `node1`, scoped Key — Infra-ADR-008/013) + pgvector v2 + RRF (AD-5 sequenziert), Per-Tenant-DEK/Envelope-Encryption **app-seitig** (AD-8 revidiert 12.06.: Hetzner-Stack hat kein Cloud-KMS — Master-Key im Coolify-Secret-

Store, Rotations-Runbook; bei P1-Ist-Stand-Check bestätigen), Mailjet (AD-6 ✓), mCaptcha zentral `captcha.saas-ctrl.com` (AD-7 ✓)

- **Infra-Stack (Infra-Repo `~/Documents/GitHub/infra`, ADR-009/010, ersetzt Scaleway seit 30.05.)**: Hetzner Cloud, ein Projekt pro SaaS (`HiveGrid` : hivegrid-db/app/worker/lb + S3-kompatibles Object Storage + Storage Box); Staging konsolidiert auf `staging-1` ; Deploys via zentrale Coolify-Control-Plane; Registry = Gitea (build-once/promote); DNS = Cloudflare (`hivegrid.de` -Zone existiert); zentrale Dienste: Meilisearch/Sentry/Loki (node1), ClamAV, mCaptcha. `fly.toml` entfernt (Infra = Hetzner/Coolify); `docs/deployment-region-provider-pinning.md` stale → P1-Ist-Stand-Check
- **Sessions**: Postgres als Source-of-Truth + Redis-Cache; Cookie hält nur Session-ID (AD-4)
- **Realtime**: WebSocket für Chat (M13), SSE für Read-Updates (M03/M22) — AD-9
- **Tech-Stack**: React Router 7, Node 22.12.x (dev/test) / node:20 (runtime), Prisma/Postgres, Redis; **Subframe-UI ist Pflicht** für jede Route
- **Repo/Naming**: Git-Downstream der @toolctrl-Boilerplate, upstream-Remote erhalten, `@toolctrl/*` -Naming behalten (AD-10)
- **Go-Live**: Big-Bang — erst live mit vollständigem Phase-1-Produkt (AD-19); interne Bau-Reihenfolge nach AD-11

Key Decisions

Vollständige Extraktion mit Scope/Begründung: `.planning/intel/decisions.md` . Quelle: `docs/ARCHITECTURE-DECISIONS.md` .

Decision	Kern	Status
AD-1	Mandanten-Trennung via echtes Postgres-RLS	✓ locked
AD-2	Typisierte Tabellen, Entity-Builder raus — **revidiert 13.06.:** `Projekt` = EINE typisierte Tabelle + CustomFields + Pipeline-Feld-Config-Layer; generischer Entity-Builder bleibt entfernt	✓ revidiert (13.06.)
AD-3	Plattform-Admin getrennt (Email+PW+TOTP, opt. Authentik-SSO)	✓ locked
AD-4	Sessions: Postgres Source-of-Truth + Redis-Cache	✓ locked
AD-5	Such-Engine: Meilisearch (v1 Keyword); pgvector-Hybrid → v2	✓ sequenziert (11.06.)
AD-6	EU-Email-Provider: **Mailjet** (Boilerplate-verdrahtet)	✓ locked (11.06.)
AD-7	Captcha: **mCaptcha self-hosted** (infra/ vorhanden)	✓ locked (11.06.)
AD-8	Verschlüsselung: Per-Tenant-DEK + Envelope, Crypto-Shredding	✓ locked
AD-9	Realtime: WebSocket (Chat) + SSE (Read-Updates)	✓ locked
AD-10	Naming: `@toolctrl/*` behalten, Docs anpassen	✓ locked
AD-11	Bau-Strategie: Hybrid (Fundament-Kern → SHK-Kernpfad → Breite)	✓ locked
AD-12	SHK-Defaults als deklaratives Seed, Noah validiert	✓ locked

AD-13	Go-Live-Scope: breiter Launch (Großteil der 36 Module)	✓ locked
AD-14	Migration: sauberer Start + KWP-Sync füllt nach	✓ locked
AD-15	Billing-Engine bauen, Preise geparkt	✓ locked
AD-16	USP-Säulen (Automation + KI) in voller Tiefe day-1	✓ locked
AD-17	Mobile: **PWA-only (v1 + v2)** — Offline via PWA/Web (P9), Web-Push in v2; **kein Expo/native/App-Store** (native → frühestens v3)	✓ revidiert (16.06.)
AD-18	Customer-Portal → komplett v2	✓ revidiert (11.06.)
AD-19	Go-Live-Modus: Big-Bang	✓ locked
AD-20	**Projekt-zentrisches Modell (NEU 13.06.):** ein `Projekt`-Kern; Pipelines = konfigurierbare Views; „Deal"/„Auftrag" = Pipeline-Typen, keine eigenen Entitäten; Lead → Projekt (Weiterverarbeitungsfreigabe); Abnahme v1 als Checklisten-Sub-Area, Rechnung v1-Sub-Entity, KWP-Felder v1-Halter/v2-Sync	✓ locked (13.06.)
KWP-Recut	Offizielle KWP-API → M089 entfällt, M090–M095 vereinfacht; M093 bleibt; KWP als `DataConnection type='rest-api'` (M33a). M089–M095 vor Umsetzung neu schneiden.	⚠ Recut nötig

---

Erzeugt: 2026-06-07 via `/gsd:ingest-docs` (Stufe 1 — WHAT) aus 38 Docs (37 SPEC + 1 ADR). ROADMAP folgt in Stufe 2.