Das HiveGrid-Datenmodell — einfach erklärt

Projekt-zentrisches CRM/Pipeline-Modell · Stand 14.06.2026 · für Design & Produkt (nicht-technisch)

Die Grundidee in einem Satz: Es gibt **ein zentrales Objekt — das Projekt**. Alles andere ist entweder eine Vorstufe davor (der Lead), eine Brille darauf (die Pipeline), ein Beteiligter (Kontakte, Mitarbeiter) oder etwas, das daran hängt (Termine, Rechnungen, Checklisten, Aufgaben, Dokumente). „Deal" und „Auftrag" sind **keine eigenen Dinge** mehr, sondern nur Namen von Pipelines — also verschiedene Sichten auf dasselbe Projekt.

Die Bausteine

Lead — die Vorstufe

Ein Sammelbecken für eingehende Anfragen aus allen Quellen (Formular, Website, Social Media, Liste, CSV, manuell, E-Mail, Anruf). Felder dürfen hier noch leer sein. Der wichtigste Punkt ist die **Weiterverarbeitungsfreigabe** (Ja / Nein / Unklar) — sie ist die Grenze zwischen „nur eine Anfrage" und „echtes Projekt".

Projekt — der Kern

Der eigentliche Arbeitsdatensatz. Trägt alle festen Angaben (Name, Quelle, Priorität, Start/Deadline, Auftragsvolumen, Adressen, zuständige Person, Projektleiter …) und beliebige **selbst definierte Felder**. Jedes Projekt läuft in **genau einer Pipeline** und steht dort auf **genau einem Status**.

Pipeline — die Brille

Eine frei benennbare Ansicht auf Projekte („Vertrieb", „Aufträge", „Wartungsverträge" …). Der Admin legt pro Pipeline fest:

- welche **Status-Schritte** (Spalten) es gibt,
- welche **Felder sichtbar oder Pflicht** sind (auch je Status),
- **wer** die Pipeline sehen und bearbeiten darf.

Kontakt — Personen & Organisationen

Ein Oberbegriff: ein Kontakt ist **entweder eine Privatperson oder eine Organisation**. Dazu kommen ein Status (Kontakt / Kunde / Bestandskunde) und mehrere Erreichbarkeiten. **Hauptkontakt** und **Sekundärkontakt** sind keine Eigenschaft des Kontakts selbst, sondern eine **Beziehung** zwischen Kontakten.

Anhänge am Projekt (Sub-Bereiche)

Termine, **Rechnungen**, Checklisten (mit Punkten), die **Abnahme** (ein eigenes, nach Unterschrift gesperrtes Protokoll) und — **ab Version 2** — KWP-/Schnittstellen-Felder. „Mängel" ist dabei einfach eine Checkliste mit dem Namen „Mängel".

Querschnitt — hängt überall dran

Aufgaben, Aktivitäten (der Verlauf), Dokumente und Tags können an jeder Sache hängen — an einem Projekt, einem Kontakt oder einem Lead.

Die Beziehungen — in einfachen Sätzen

Vom Lead zum Projekt

> Ein Lead wird zu **höchstens einem** Projekt.

> Ein Projekt entsteht aus **höchstens einem** Lead — kann aber auch direkt ohne Lead angelegt werden.

> Ein Lead kann mit **einem** Kontakt verknüpft werden; der Kontakt bleibt eigenständig.

Projekt und Pipeline

> Ein Projekt liegt in **genau einer** Pipeline.

> Eine Pipeline enthält **beliebig viele** Projekte.

> Eine Pipeline hat **mehrere** Status-Schritte; ein Projekt steht immer auf **genau einem** Status-Schritt.

> Eine Pipeline bestimmt selbst, **welche** Felder sichtbar oder Pflicht sind und **wer** sie sehen darf.

Kontakte

> Ein Kontakt ist **entweder** eine Privatperson **oder** eine Organisation.

> Ein Kontakt kann **mehrere** E-Mail-Adressen und **mehrere** Telefonnummern haben.

> Eine Organisation kann **mehrere** Hauptkontakte und **mehrere** Sekundärkontakte haben.

> Eine Privatperson kann **selbst** Hauptkontakt sein und zusätzlich **mehrere** Sekundärkontakte haben.

> Ein Kontakt kann an **mehreren** Projekten beteiligt sein, und ein Projekt kann **mehrere** Kontakte haben — jeweils mit einer Rolle (Auftraggeber, Bauherr, Rechnungsempfänger, Ansprechpartner).

Beteiligte und Adressen

> Ein Projekt hat **eine** zuständige Person und **einen** Projektleiter.

> Ein Projekt kann **mehrere** Monteure haben, und ein Monteur kann an **mehreren** Projekten arbeiten.

> Ein Projekt hat **eine** Projektadresse und **optional je eine** separate Rechnungs- und Bauherradresse.

Anhänge am Projekt

> Ein Projekt kann **beliebig viele** Termine, Rechnungen, Checklisten, Aufgaben und Dokumente haben — jedes davon gehört zu **genau einem** Projekt.

> Eine Checkliste besteht aus **mehreren** Punkten; jeder Punkt gehört zu **genau einer** Checkliste.

> Eine Abnahme gehört zu **genau einem** Projekt, bindet **eine** Mängel-Checkliste ein und sammelt **mehrere** Unterschriften (Kunde und Monteur). Nach der Unterschrift ist sie gesperrt.

> Ein Projekt kann **mehrere** KWP-/Schnittstellen-Felder tragen — **erst ab Version 2** (in Version 1 noch nicht enthalten; KWP-Anbindung kommt nach dem Launch).

Eigene Felder & Querschnitt

> Ein Betrieb kann **beliebig viele** eigene Felder definieren; jedes davon lässt sich pro Pipeline ein-/ausblenden und zur Pflicht machen.

> Aufgaben, Aktivitäten, Dokumente und Tags können an **jeder** Sache hängen — Projekt, Kontakt oder Lead.

Was das für den Nutzer bedeutet

> **1. Der Betrieb baut sich sein CRM selbst.** Jeder SHK-Betrieb legt eigene Pipelines an, benennt sie, blendet Felder ein/aus, macht Felder zur Pflicht und legt fest, wer reinschauen darf — ohne Entwickler.

> **2. Keine Brüche, keine Doppelerfassung.** Ein Auftrag ist derselbe Datensatz wie der frühere „Deal" — er wechselt nur die Pipeline. Der Verlauf bleibt lückenlos vom ersten Kontakt bis zur Abnahme.

> **3. Ein klarer Status statt zwei.** „Status" heißt immer: wo steht das Projekt in *dieser* Pipeline. Kein verwirrender zweiter Gesamtstatus.

> **4. Flexible Kontaktwelt.** Privatkunden, Firmen mit mehreren Ansprechpartnern, Sekundärkontakte — alles abbildbar, und ein Kontakt taucht überall im richtigen Zusammenhang auf.

> **5. Rechtssichere Abnahme & Rechnungen am Projekt.** Die Abnahme bleibt unveränderlich mit Unterschrift; Rechnungen lassen sich direkt am Projekt führen.

Die Abläufe — so fühlt es sich an

> ### A · Anfrage wird zum Projekt
>
> Anfrage landet als Lead in der Inbox. Der Sachbearbeiter prüft und setzt die Weiterverarbeitungsfreigabe auf **Ja** → ein Projekt entsteht, die bekannten Daten werden übernommen, der Kunde wird als Kontakt verknüpft. Bei **Nein/Unklar** bleibt es ein Lead.

> ### B · Projekt wandert durch den Vertrieb
>
> Im Kanban der Vertriebs-Pipeline zieht man das Projekt von Spalte zu Spalte (= Status). Bei „gewonnen" taucht **dasselbe** Projekt in der Auftrags-Pipeline auf — mit deren Feldern und Schritten. Keine Übergabe, kein Neuanlegen.

> ### C · Admin richtet eine neue Pipeline ein
>
> Beispiel „Wartungsverträge": Spalten definieren, festlegen welche Felder sichtbar/pflichtig sind und welche Rollen Zugriff haben. Ab sofort können Projekte dort laufen.

D · Monteur auf der Baustelle

Öffnet das Projekt mobil → Abnahme-Checkliste abarbeiten, Mängel mit Foto und Schweregrad erfassen → Kunde und Monteur unterschreiben auf dem Display → Protokoll wird gesperrt und als PDF abgelegt. Funktioniert **offline**.

E · Büro arbeitet am Projekt

Rechnung am Projekt anlegen, Termin einplanen, Aufgabe an einen Kollegen zuweisen, Dokument anhängen — alles hängt am selben Projekt.

F · Kontaktpflege mit Rückschreiben

In der Projektansicht steht die Telefonnummer des Auftraggebers; man ändert sie direkt dort → die Änderung wird **auf den Kontakt zurückgeschrieben** (das Feld „gehört" dem Kontakt, wird nur im Projekt angezeigt). Der Verlauf hält fest, wer wann was geändert hat.

Kurz: ein Datensatz, viele konfigurierbare Brillen. „Vertrieb" und „Auftrag" werden über die Pipeline-Konfiguration unterschieden statt über getrennte Datentöpfe — flexibler für den Betrieb. *Die sorgfältig durchdachte Stelle dabei sind Sichtbarkeit & Rechte pro Pipeline — voll ausgebaut: einzelne Felder lassen sich **pro Rolle/Gruppe** ein- oder ausblenden (z.B. das Auftragsvolumen nur für den Vertrieb).*