```
module: 5
title: Form Builder
phase: 1
status: detail-spec-draft
last_updated: 2026-05-08
session_source: brainstorming-session-2026-05-07-2206.md
related_lps: [LP-3, LP-9, LP-10]
spec_style: requirements-pur
size: XL
```

Modul 5 — Form Builder

> ***Detail-Spec gemäß LP-9.*** *XL-Modul. Voll Google-Forms-Niveau zu Phase 1 (User-Entscheidung). Klebstoff zwischen externer Welt und HiveGrid. Basis: alte Codebase + Brainstorming-Ergänzungen.*

Themenblöcke

```
A. Datenmodell
B. Form-Editor (Drag-&-Drop)
C. Feldtypen
D. Validation-Regeln
E. Sharing & Embed
F. Form-Templates (SHK)
G. Submissions-Verwaltung
H. Form-Actions (bei Submit)
I. Integration mit anderen Modulen
J. Permissions
K. Phase 2 (Verzweigungslogik, Charts, etc.)
```

Block A — Datenmodell

```
Form                  – Form-Definition pro Tenant
FormField             – Feld innerhalb eines Forms
FormSubmission        – eine Antwort/Einreichung
FormSubmissionAnswer  – pro Feld pro Submission der Antwort-Wert
FormFile              – Datei-Upload aus Submission
FormLink              – Sharing-Link mit Token
FormsTag / FormTag    – Tag-System
FormAction            – was passiert bei Submit
FormActionLog         – Ausführungs-Log pro Action
FormLog               – generelles Audit-Log pro Form
```

Form

- `id`, `tenant_id`, `created_at`, `updated_at`, `created_by_user_id`
- `name`, `description` (Markdown)
- `slug` (URL-Pfad-Bestandteil, unique pro Tenant)
- `subtype` (`standard` | `acceptance_form` | `onboarding` | `survey`)
- `status` (`draft` | `published` | `archived`)

- `published_at`, `archived_at`
- `is_public` (über öffentliche URL einsehbar?)
- `requires_login` (User-Login Pflicht — Customer-Portal-User reicht)
- `submission_limit` (optional — max. Anzahl Submissions)
- `submission_close_at` (optional — Auto-Schließung)
- `success_message` (Markdown)
- `redirect_url` (optional)

FormField

- `id`, `form_id`, `position`
- `field_key` (snake_case, unique pro Form)
- `label`, `description` (Hilfetext)
- `field_type` (Enum — siehe Block C)
- `field_config` (JSON — typ-spezifisch)
- `required` (Boolean)
- `validation_rules` (JSON — siehe Block D)
- `default_value` (JSON)
- `placeholder`
- `conditional_visibility` (JSON — Phase 2)

FormSubmission

- `id`, `tenant_id`, `form_id`, `submitted_at`
- `submitter_user_id` (nullable)
- `submitter_email`, `submitter_phone`
- `submitter_ip`, `submitter_user_agent` (Audit)
- `submission_data` (JSON — alle Antworten)
- `linked_entity_type`, `linked_entity_id` (z.B. → Deal-ID)
- `is_complete` (Phase 2 für Multi-Step Save-and-Continue)
- `archived_at`

FormFile

- `id`, `submission_id`, `field_key`, `file_id` (FK auf M23 File)

FormLink

- `id`, `form_id`, `token`
- `created_by_user_id`, `created_at`
- `expires_at` (optional)
- `usage_count`, `usage_limit` (optional)
- `purpose` (Freitext)

FormsTag / FormTag

- Standard-Tag-System (analog M1)

FormAction

- `id`, `form_id`, `position`
- `action_type` (Enum — siehe Block H)
- `action_config` (JSON)
- `enabled`

- `runs_on` (`submission | submission_with_filter`)
- `filter_rules` (JSON)

FormActionLog

- `id` , `submission_id` , `action_id`
- `executed_at` , `status` (`success | failed | skipped`)
- `result` (JSON), `error_text`

FormLog

- `id` , `form_id` , `event_type` , `event_data` , `created_at` , `created_by_user_id`
- Audit für Form-Änderungen

Block B — Form-Editor

Editor (`/{tenant}/forms/:id/edit`)

Zu zeigende Daten:
- Form-Stammdaten
- Liste aller Felder in Reihenfolge
- Live-Vorschau
- Validation-Hinweise

Benötigte UI-Elemente:
- Sektion für Stammdaten
- Drag-&-Drop-Liste der Felder
- Add-Field-Funktion mit Feldtyp-Picker
- Pro Feld: Edit / Duplizieren / Löschen / Sortieren
- Live-Vorschau-Pane
- Save + Publish-Funktion
- Sharing-Funktion (Link-Generator)
- Embed-Code-Generator
- Versionierung (Phase 2)

Operationen:
- Stammdaten editieren
- Feld hinzufügen / löschen / duplizieren / verschieben
- Feld-Konfiguration editieren
- Validation pro Feld setzen
- Form publishen / unpublishen
- Form duplizieren (als Vorlage)
- Form archivieren

Klartext-Validation:
- Feld ohne Label: *„Feld 3 hat keinen Label — bitte ausfüllen."*
- Slug-Konflikt: *„Slug ‚kontakt' wird bereits von einem anderen Form genutzt."*
- Required ohne Validation: *„Feld ‚Email' ist required, aber keine Validation gesetzt — Email-Format-Validation hinzufügen?"*

Block C — Feldtypen (Phase 1)

Feldtyp	Anwendung
`text`	Single-Line Text
`textarea`	Multi-Line Text
`number`	Integer / Decimal
`date` / `datetime` / `time`	Datums-/Zeit-Eingaben
`select`	Single-Select Dropdown
`multiselect`	Multi-Select Chips
`boolean`	Checkbox / Toggle
`email`	Auto-Format-Validation
`url`	Auto-Format-Validation
`phone`	E.164-Normalisierung
`file`	Beliebige Datei-Upload
`photo`	Foto-Upload mit Mobile-Kamera-Direktzugriff
`signature`	Touch-Signatur-Komponente
`address`	Auto-Complete (Maps-Adapter M25)
`section_header`	Anzeige-Element
`description_block`	Markdown-Anzeige
`hidden`	Vorausgefüllt aus URL-Params

Phase 2: `repeater`, `multi_step`, `conditional_visibility`.

Block D — Validation-Regeln

Pro Feld:
- `required`
- `min_length`, `max_length`
- `min_value`, `max_value`
- `regex_pattern`
- `accepted_file_types[]`, `max_file_size_mb`
- `custom_error_message`

Phase 2: Cross-Field-Validation.

Block E — Sharing & Embed

Public URL

`app.hivegrid.de/f/{tenant_slug}/{form_slug}` — wenn `is_public=true`.

Personalisierte Links

- `app.hivegrid.de/f/{tenant_slug}/{form_slug}/?token={token}` mit FormLink
- Tracking, optional `usage_limit`, `expires_at`

iFrame-Embed

- Embed-Code-Generator
- Auto-Resize-JS

JS-Snippet (Inline)

- Direkt in externe Website embeddet
- CORS-konform via `app.hivegrid.de/embed.js`

QR-Code

- Generator für Print-Material (PNG/SVG)

Block F — Form-Templates (SHK-Branchen-Template)

Beim Tenant-Anlegen geseedet:

Template	Subtype	Zweck
„Kontakt-Anfrage"	standard	Lead-Kontaktformular für Website
„Sales-Aufnahme Vor-Ort"	standard	Vertriebler nimmt vor Ort technische Daten auf
„Foto-Checkliste"	standard	Monteur dokumentiert Bestandsanlage
„Acceptance-Protokoll Anlage"	acceptance_form	siehe M4 SHK-Default
„Wartungs-Buchung"	standard	Endkunde bucht Wartungstermin
„Krankmeldung"	standard	Mitarbeiter-Form (M14)
„Urlaubsantrag"	standard	Mitarbeiter-Form (M14)
„Customer-Feedback-Survey"	survey	Nach Acceptance an Kunde

Templates deklarativ (LP-3 + AV #3).

Block G — Submissions-Verwaltung

Liste (`/{tenant}/forms/:id/submissions`)

Zu zeigende Daten:
- Submitter (Email/Phone)
- Submission-Datum
- Verlinkte Entität
- Status der Action-Ausführungen
- Ausgewählte Feld-Werte als Spalten

Benötigte UI-Elemente:
- Tabellarische Anzeige

- Filter nach Datum / Entität / Submitter
- Spalten-Konfiguration
- Bulk-Auswahl
- Export als CSV
- Klick öffnet Detail

Detail (`/{tenant}/forms/:id/submissions/:submission_id`)

Zu zeigende Daten:
- Alle Antworten in Form-Reihenfolge
- Submitter-Metadata (IP, User-Agent, Datum)
- Ausgeführte Form-Actions mit Status + Klartext
- Verknüpfte Entitäten
- File-Uploads als Vorschau

Operationen:
- Action erneut ausführen (Re-Run)
- Manuell Entitäten verknüpfen
- Submission archivieren / löschen (DSGVO)
- PDF-Export der Submission

Block H — Form-Actions (bei Submit)

Pro Form mehrere Actions in Reihenfolge:

Action	Zweck
`send_email_to_submitter`	Bestätigung an Submitter
`send_email_to_team`	Email an interne Empfänger
`create_lead`	Lead in Lead-Inbox (M2)
`create_contact`	Kontakt in CRM (M1)
`create_deal`	Deal in Sales-Pipeline (M3) — mit Pipeline-Auswahl
`create_project`	Project in Auftragspipeline (M4) — mit Pipeline-Auswahl
`create_task`	Task (M26)
`link_to_existing_entity`	Auto-Match über Email/Phone
`webhook_send`	Externer Webhook
`trigger_automation`	Automation-Engine triggern (M17)
`kwp_push`	Daten an KWP (M19) — z.B. Acceptance-Daten

Field-Mapping: pro Action konfigurierbar (Form-Feld → Entitäts-Feld).

Conditional Actions: mit `filter_rules` nur bei bestimmten Inhalten.

Block I — Integration mit anderen Modulen

Modul	Integration
M1 CRM	Submission → Contact-Anlage; Auto-Match Email/Phone
M2 Lead-Inbox	Submission → Lead-Anlage Standard-Workflow
M3 Sales-Pipeline	`create_deal`-Action mit Pipeline-Targeting
M4 Projekte	Sales-Aufnahme + Foto-Checklisten + Acceptance als ProjectForm
M9 Email	Bestätigungs-Mails über Actions
M11 WhatsApp (P2)	Bestätigung per WhatsApp
M17 Automation	Trigger `form.submitted`; Actions können Automation triggern
M19 KWP	Acceptance-Forms pushen Daten zu KWP
M22 Notifications	Push/Email an Tenant-User bei neuer Submission
M23 Files	File-/Photo-Uploads landen im Tenant-File-Storage
M24 Suche	Forms + Submissions im Suchindex
M25 Calendar	Address-Feld nutzt Maps-Adapter
M14 HR-Hub	Krankmeldung/Urlaubsantrag-Forms

Block J — Permissions

Permission	Erlaubt
`forms.read_own`	Eigene Forms sehen
`forms.read_all`	Alle Tenant-Forms sehen
`forms.write`	Forms anlegen + bearbeiten
`forms.publish`	Forms publishen / unpublishen
`forms.delete`	Hard-Delete
`forms.view_submissions`	Submissions einsehen
`forms.export_submissions`	CSV-Export
`forms.admin`	Voll-Verwaltung + Templates + Bulk-Operations

Bundle-Mapping

Bundle	Atomare Permissions
`forms.basic`	`read_own` + `view_submissions`
`forms.builder`	obigen + `read_all` + `write`
`forms.publisher`	obigen + `publish` + `export_submissions`
`forms.admin`	alle

Default-Rollen-Zuweisung

Rolle	Permission
GF	`forms.admin`
Vertriebsleiter / PL	`forms.publisher`
Marketing	`forms.publisher`
Vertriebler / Büro	`forms.builder`
Monteur	`forms.basic`

Block K — Phase 2

- Verzweigungslogik (`conditional_visibility`)
- Multi-Step-Forms mit Save-and-Continue
- Form-Charts / Statistik-View
- Cross-Field-Validation
- A/B-Testing
- Repeater-Felder
- Form-Templates-Marketplace (M33c)

Architektur-Vorschriften (Querschnitt)

- **LP-3, AV #3:** SHK-Form-Templates Branchen-Template-Seeding
- **LP-10, AV #1:** Klartext-Validation-Errors
- **AV #5:** Permissions-aware-Default
- **AV #6:** Audit-Trail (FormLog)

Detail-Spec-Status: DRAFT

Modul 5 komplett spec'd basierend auf alter Codebase.

Offene Detail-Punkte:
- Drag-&-Drop-Library-Wahl (Boilerplate)
- Anti-Spam für öffentliche Forms (Honeypot, hCaptcha — analog M27)
- Versionierungs-Strategie (Phase 2)