ASDW CORE • PRODUKTDOKUMENTATION

Produktbroschüre

ASDW Core

Das modulare Fundament für geschäftliche Webanwendungen, Mehrmandanten-Plattformen und kontrollierte KI-Workflows

ÜBER DIESE BROSCHÜRE

Diese eigenständige Produktbroschüre zeigt anhand realer Oberflächenansichten, schematischer Architektur- und Ablaufdiagramme, vier durchgängiger Beispielszenarien sowie konkreter Konfigurationsbeispiele, wie die zwanzig Module von ASDW Core im Alltag zusammenwirken.

20 Module
Durchgängige Architektur mit einheitlicher Rechte-, Design- & Hilfe-Basis

4 Szenarien
Praxisnahe Gesamtabläufe von der Visitenkarte bis zum MCP-KI-Gateway

8 Grafiken
Echte Light-Mode-Oberflächen & schematische Vektor-Diagramme

PRODUKT	DOKUMENTTYP	SYSTEMSTAND	AUSGABE
ASDW Core	**Produktbroschüre**	**Core v1.2.93**	**24.11.2026**

Inhaltsverzeichnis

1. Einstieg und Einordnung 3

Ausgangssituation und typische Aufgaben 3
Was ist ASDW Core? 3
Wer arbeitet mit ASDW Core – und wer profitiert davon? 3
Welche Rolle spielt ASDW Core im Anwendungsumfeld? 3

2. Aufbau und Zusammenspiel der Module 4

Die Grundidee: Plattform und Fachfunktion sauber trennen 4
Warum dieses Zusammenspiel im Alltag zählt 4

3. Die Funktionsbereiche im Überblick 5

3.1 Arbeitsplatz, Navigation, Design und Mehrsprachigkeit 5
3.2 Anmeldung, Rollen, Schutz und Nachvollziehbarkeit 6
3.3 E-Mail-Ökosystem und Kontakterfassung 7
3.4 Datenpflege, Betrieb, Releases und Flottensteuerung 8
3.5 Plattform-Erweiterungen: SaaS, Web-Push, KI-Schnittstelle und Video-Studio 8

4. Anwendungsszenarien aus der Praxis 9

Beispielszenario 1: Vom Messekontakt zur ausgewerteten E-Mail-Kampagne 9
Beispielszenario 2: Eine mandantenfähige Fachplattform einrichten und verwalten 10
Beispielszenario 3: Mehrere abgeleitete Kundenprojekte zentral aktualisieren und absichern 12
Beispielszenario 4: KI-Assistenten kontrolliert einbinden und Anwender aktiv begleiten 13

5. Konkrete Beispiele aus Bedienung und Konfiguration 16

Beispiel A: Zweischichtige Ansprache in einer mehrsprachigen E-Mail-Vorlage 16
Beispiel B: Kontrollierte KI-Freigabe im MCP-Cockpit 16
Beispiel C: Mehrstufige Design-, Sprach- und Rollensteuerung im Team-Portal 17

6. Voraussetzungen, optionale Dienste und Grenzen 18

Technische Grundvoraussetzungen 18
Optionale Dienste und Konfigurationsschalter 18
Bewusste Grenzen und nicht belegte Bereiche 19

7. Zusammenfassende Orientierung 19

1. Einstieg und Einordnung

Ausgangssituation und typische Aufgaben

Organisationen, Softwareanbieter und Projektteams stehen bei der Einführung oder Weiterentwicklung eigener Webanwendungen regelmäßig vor denselben Grundaufgaben: Benutzer müssen sicher angemeldet, Zugriffsrechte nachvollziehbar verwaltet, Oberflächen mehrsprachig und einheitlich gestaltet, E-Mails zuverlässig versendet und Änderungen lückenlos protokolliert werden. Kommen später mehrere Kundenprojekte, getrennte Mandanten oder KI-gestützte Arbeitsabläufe hinzu, wächst der Pflegeaufwand rasch an, sobald jede Anwendung ihre eigene Technik mitbringt.

Was ist ASDW Core?

ASDW Core ist ein modulares Web-Anwendungs-Framework auf Basis von PHP 8 und MySQL beziehungsweise MariaDB. Es stellt ein fertiges, erweiterbares Systemfundament bereit, das sechzehn zentrale **Core-Module** (Systemmodule) mit vier sofort nutzbaren **fachlichen Modulen** (Standardmodulen) verbindet.

Statt wiederkehrende Basisfunktionen in jedem Projekt neu zu entwickeln, bündelt ASDW Core diese Aufgaben in einer einheitlichen Plattform. Gleichzeitig lässt sich die Anwendung nicht nur klassisch im Browser nutzen, sondern auch als **Progressive Web App (PWA)** – also als Webanwendung, die sich mit eigenem Symbol direkt auf Desktop-Rechnern und Smartphones installieren lässt.

Wer arbeitet mit ASDW Core – und wer profitiert davon?

- **Entscheider und Plattformbetreiber** erhalten eine klar strukturierte, modulare Basis, die sich von der Einzelinstanz bis zur mehrmandantenfähigen SaaS-Plattform oder zu einer Flotte abgeleiteter Kundenprojekte einsetzen lässt – ohne unnötige Abhängigkeiten von externen Diensten, solange diese nicht ausdrücklich aktiviert werden.
- **Administratoren und Projektverantwortliche** steuern Erscheinungsbild, Menüs, Benutzer, Gruppenrechte, Sicherheitsregeln, Datenbankprofile, Sicherungen, Auslieferungen und KI-Freigaben zentral an einer Stelle.
- **Fachanwender ohne Programmierkenntnisse** arbeiten in einer durchgängig gestalteten, auch mobil bedienbaren Oberfläche mit eingeblendeter Online-Hilfe, klaren Tabellenfiltern und nachvollziehbaren Arbeitsabläufen – etwa beim Gestalten von E-Mails, beim Steuern von Kampagnen oder beim Digitalisieren von Visitenkarten.

Welche Rolle spielt ASDW Core im Anwendungsumfeld?

ASDW Core kann je nach Bedarf vier verschiedene Rollen einnehmen:

1. **Eigenständige Geschäftsanwendung (Standalone):** Direkt nutzbares Portal für eine einzelne Organisation.
2. **Mehrmandantenfähige SaaS-Plattform:** Zentral verwaltete Plattform, bei der jeder Mandant über eine eigene Subdomain und eine eigene, getrennte Datenbank arbeitet (**Silo-Modell**).
3. **Master-Plattform für abgeleitete Satellitenprojekte:** Zentrale Weiterentwicklung des Core-Fundaments mit kontrollierter Verteilung von Aktualisierungen auf eigenständige Kunden- und Fachprojekte.

4. **Ergänzung bestehender Scriptcase-Umgebungen (Dual-Mode):** Ausgewählte Module lassen sich sowohl eigenständig als auch eingebettet in Scriptcase über eine gesicherte IFrame-Sitzungsbrücke betreiben.

---

2. Aufbau und Zusammenspiel der Module

Die Grundidee: Plattform und Fachfunktion sauber trennen

Das Herzstück von ASDW Core ist die konsequente Trennung zwischen **Core-Modulen** und **fachlichen Modulen**:

- Die **16 Core-Module** liefern alle querschnittlichen Dienste: Portal-Shell, Navigation, Design-System, Mehrsprachigkeit, Authentifizierung, Gruppenrechte, Sicherheits-Firewall, Protokollierung, Datenbank- und Release-Werkzeuge, Flottensteuerung, SaaS-Mandantenbetrieb, Web-Push, die KI-Schnittstelle nach dem **Model Context Protocol (MCP)** sowie das integrierte **Video-Studio** zur Erstellung von Dokumentationen, Kurzvideos und Online-Hilfe-Clips aus den vorhandenen Modulen.
- Die **4 fachlichen Module** bilden konkrete Arbeitsprozesse im Tagesgeschäft ab: visuelle E-Mail-Gestaltung (**email_editor**), E-Mail-Versand und Rückläufer-Auswertung (**email_delivery**), KI-gestützte Visitenkartenerfassung (**scan_business_card**) sowie direkte Tabellenpflege (**dbeditor**).

Warum dieses Zusammenspiel im Alltag zählt

Sobald ein fachliches Modul oder ein später ergänztes Kundenmodul geöffnet wird, greifen automatisch sämtliche Mechanismen des Systemkerns:

- Die **Sicherheits-Firewall** prüft Anmeldestatus und Modulberechtigungen.
- Die **Design- und Theme-Verwaltung** liefert konsistente Farben, Schriften und Abstände – passend zur Auswahl des Systems, des Moduls oder des angemeldeten Benutzers.
- Die **Sprachverwaltung** stellt alle Beschriftungen und Hilfetexte in der gewählten Sprache bereit.
- Die **Online-Hilfe** öffnet über das Fragezeichen-Symbol neben jeder Überschrift genau die Anleitung (und gegebenenfalls das passende Hilfe-Video) zum aktuellen Modul.
- Das **Audit-Protokoll** hält relevante Änderungen nachvollziehbar fest, und das **MCP-Gateway** ermöglicht bei angebundenen Modulen kontrollierte KI-Unterstützung innerhalb exakt derselben Benutzerrechte.

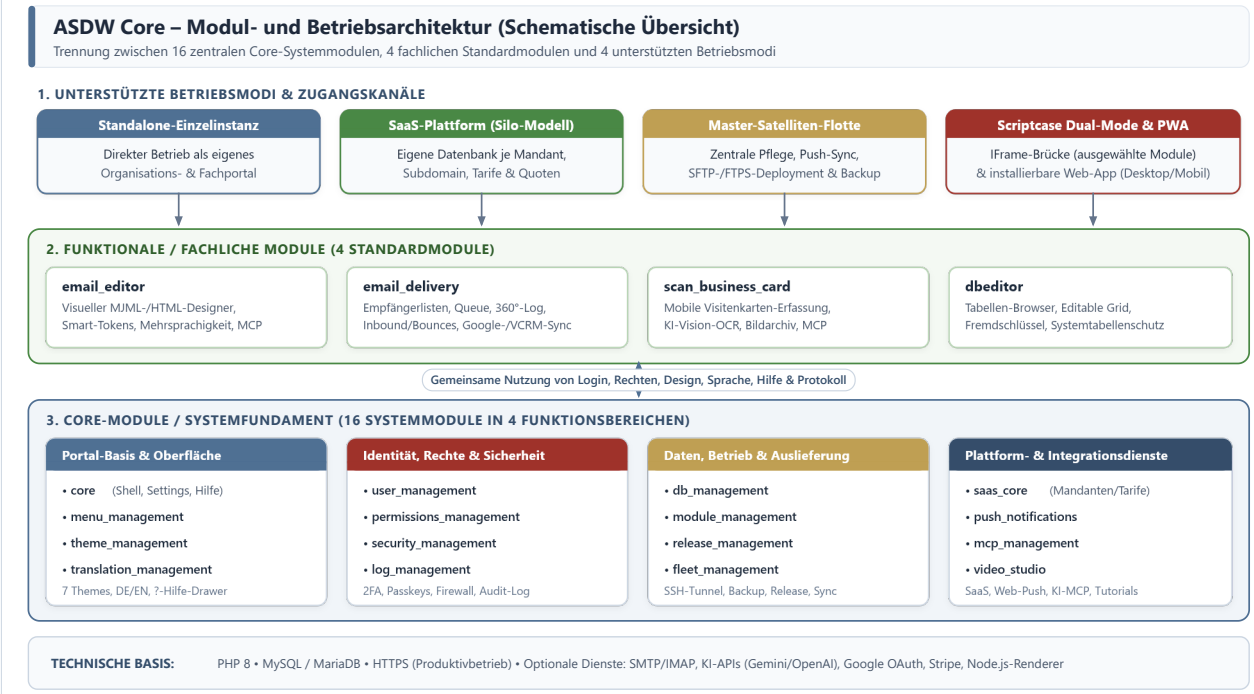


Abbildung 1 – Schematische Darstellung: Zusammenspiel der 16 Core-Module, der 4 fachlichen Module und der vier Betriebsmodi von ASDW Core.

---

3. Die Funktionsbereiche im Überblick

Um die Funktionsvielfalt greifbar zu machen, werden alle 20 bestätigten Module im Folgenden entlang von fünf zusammenhängenden Themenbereichen vorgestellt.

3.1 Arbeitsplatz, Navigation, Design und Mehrsprachigkeit

*Module: **core**, **menu_management**, **theme_management**, **translation_management***

Nach der Anmeldung begrüßt das Portal Anwender im zentralen Arbeitsbereich (**core**). Mehrere Ansichten lassen sich parallel in internen Tabs öffnen, sodass paralleles Arbeiten ohne gegenseitige Sitzungsblockaden möglich ist. Das Dashboard zeigt – abgestimmt auf die jeweilige Rolle – relevante Kennzahlen, Modul-Kacheln und auf Wunsch eine Schnellstartleiste. Über die sechsstufigen Systemeinstellungen pflegen Administratoren Anwendungsnamen, Logos, App-Icons, Fußzeilen, Startseiten-Verhalten, Sicherheitsvorgaben, System-E-Mail-Vorlagen, Satelliten-Vorgaben und optionale Support-Anbindungen zentral.

Die **Menüverwaltung** (**menu_management**) steuert Hauptmenü, seitliche Navigationsleisten und rollenspezifische Startseiten dynamisch aus der Datenbank – inklusive Sortierung per Ziehen und Ablegen sowie automatischer Ausblendung nicht berechtigter Einträge. Für das visuelle Erscheinungsbild sorgt die **Theme-Verwaltung** (**theme_management**): Sieben vorkonfigurierte Standard-Themes (*Corporate Clean*, *Neo-Grotesk Tech*, *ASDW Midnight Glow*, *Nordic Frost*, *Catppuccin*, *Emerald Business*, *Dracula*) lassen sich per geteilter Echtzeit-Vorschau anpassen, exportieren und nach einer klaren Hierarchie (Benutzer vor Modul vor Systemstandard) anwenden.

Gleichzeitig verwaltet **translation_management** alle Oberflächentexte und Hilfe-Artikel zentral in mehreren Sprachen (standardmäßig Deutsch und Englisch, um weitere Sprachen erweiterbar) und liefert sie über einen schnellen Text-Cache aus.

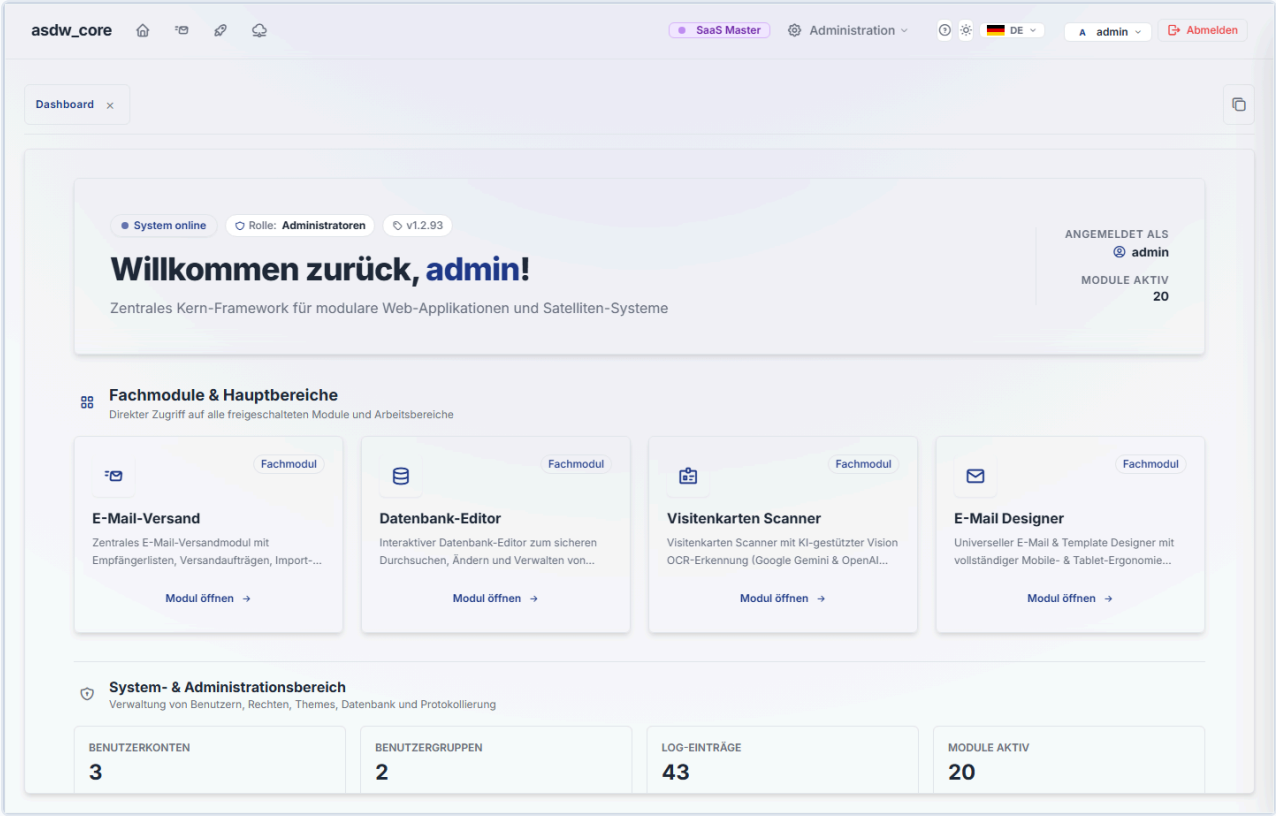


Abbildung 2 – Echte Bedienoberfläche: Zentrales Arbeitsportal und Dashboard von ASDW Core im hellen Design (Light Mode) mit Modul-Kacheln, Sprachumschaltung, Online-Hilfe-Symbol und Systemkennzahlen.

3.2 Anmeldung, Rollen, Schutz und Nachvollziehbarkeit

*Module: **user_management**, **permissions_management**, **security_management**, **log_management***

Sichere Zugangskontrolle und alltagstaugliche Bedienung schließen sich in ASDW Core nicht aus. Die **Benutzerverwaltung** (**user_management**) kombiniert klassische Passwörter (mit konfigurierbaren Komplexitätsregeln und erzwungenem Erst-Passwortwechsel) mit modernen Verfahren: **Zwei-Faktor-Authentifizierung (2FA)** über zeitlich begrenzte 6-stellige E-Mail-Codes, passwortlose Anmeldung via **Passkeys** (WebAuthn/FIDO2, etwa über Fingerabdruck oder Geräte-PIN), optionale **Google-Anmeldung** sowie eine kryptografisch abgesicherte **30-Tage-Daueranmeldung** für lokal installierte Web-Apps – inklusive Selbstverwaltung aktiver Sitzungen und sofortiger Widerrufsmöglichkeit.

Über **permissions_management** werden Benutzer einer oder mehreren Gruppen zugeordnet; die jeweiligen Modulrechte wirken kumulativ. Bei jedem Seiten- oder Schnittstellenaufruf prüft die **Sicherheits-Firewall** (**security_management**) Anmeldestatus, Passwortwechsel-Pflicht und Modulberechtigungen. Ergänzend greifen eine Fehlversuchssperre gegen wiederholte falsche Anmeldungen sowie ein konfigurierbarer Inaktivitäts-Sperrbildschirm, der geöffnete Eingabemasken schützt, ohne ungespeicherte Eingaben zu verwerfen. Alle sicherheits- und datenrelevanten Vorgänge – einschließlich strukturierter Vorher-Nachher-Vergleiche – landen gemeinsam mit automatisch abgefangenen Laufzeitfehlern im übersichtlichen **Audit- und Fehlerprotokoll** (**log_management**).

3.3 E-Mail-Ökosystem und Kontakterfassung

*Module: **email_editor**, **email_delivery**, **scan_business_card***

Drei der vier fachlichen Module greifen im Bereich Kunden- und Kontaktkommunikation eng ineinander:

- Mit dem **E-Mail-Editor** (**email_editor**) gestalten Fachanwender responsive E-Mail-Vorlagen per Drag-and-Drop auf MJML-Basis – auch mobil auf Tablet oder Smartphone. Mehrsprachige Varianten pro Vorlage, optionale 1-Klick-KI-Übersetzung mit Platzhalterschutz, intelligente Platzhalter (etwa für geschlechtsspezifische Anreden), Preheader-Steuerung mit Posteingang-Simulator, eine Bild- und Dokumentenbibliothek sowie die *Time-Machine*-Versionshistorie unterstützen eine konsistente Kommunikation.
- Das **E-Mail-Versandmodul** (**email_delivery**) übernimmt diese Vorlagen für Kampagnen und Einzelmails. Es verwaltet Empfängerlisten mit Import- und Duplikatprüfung, optionalem **Google-Kontakte-Abgleich** (verschlüsselter OAuth-Tresor, Gruppenfilter, Trockenlauf-Vorschau ohne Übernahme privater Notizen) sowie optionalem **VCRM-Import**. Eine asynchrone Versand-Warteschlange mit einstellbarer Versanddrosselung (Rate-Limiter), Blocklisten mit Abmeldelink (*Opt-out*), eine 360°-Versandhistorie mit 1-Klick-Nachversand und Adresskorrektur sowie eine dreistufige Rückläufer- und Antwort-Auswertung (technische Zustellmeldungen, Regel-Heuristiken und optional KI-Klassifizierung) schließen den Kreislauf.
- Der **Visitenkarten-Scanner** (**scan_business_card**) beschleunigt die Kontakterfassung am Smartphone oder Desktop: Hochgeladene oder fotografierte Visitenkarten werden über optionale KI-Bilderfassung (Google Gemini oder OpenAI) in strukturierte Kontaktfelder übertragen, gemeinsam mit dem Originalbild archiviert, im mobilen Bearbeitungsdialog geprüft und auf Wunsch (im Einzelinstanz-Betrieb) an ein angebundenes VCRM-Profil übergeben.

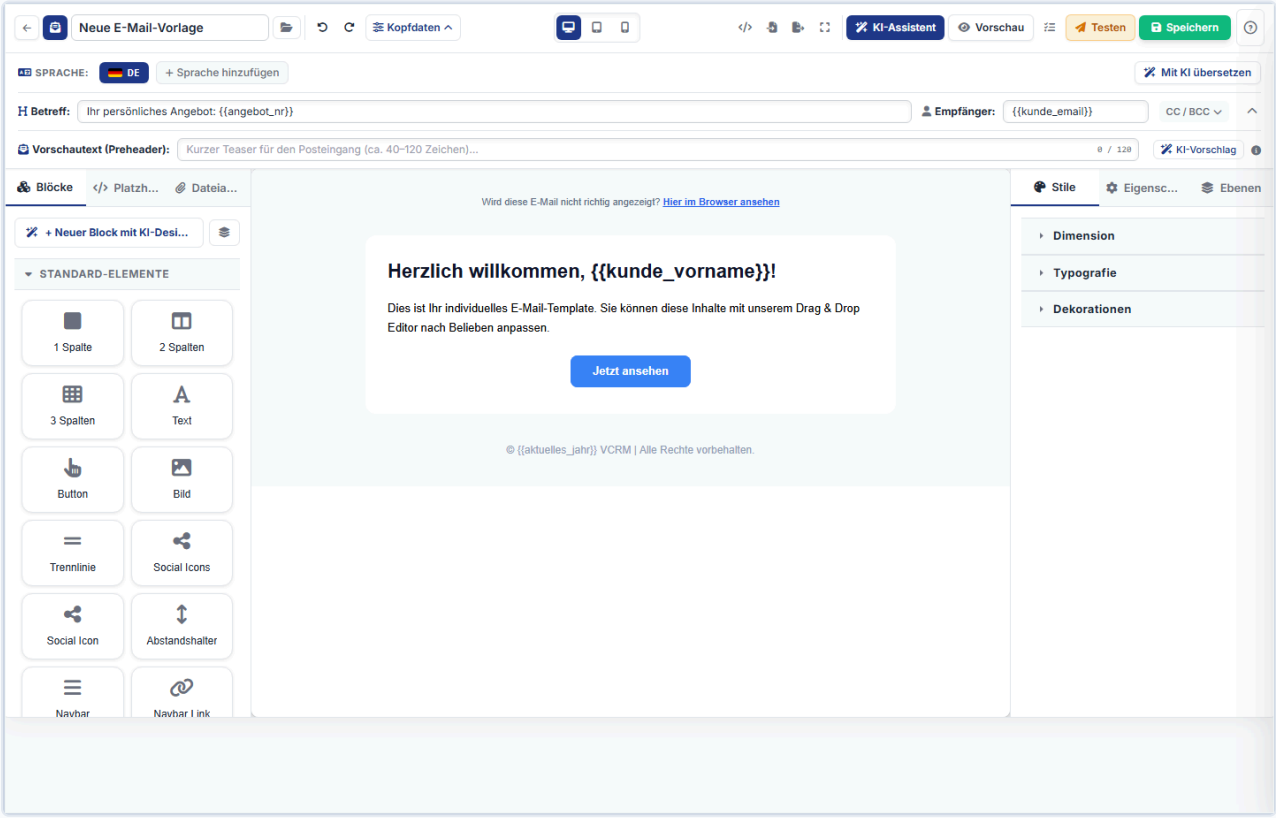


Abbildung 3 – Echte Bedienoberfläche: Visuelle Vorlagengestaltung im E-Mail-Editor mit Drag-and-Drop-Bausteinen, Sprachumschaltung, Preheader-Feld, KI-Unterstützung und Live-Arbeitsfläche.

3.4 Datenpflege, Betrieb, Releases und Flottensteuerung

Module: ***db_management***, ***dbeditor***, ***module_management***, ***release_management***, ***fleet_management***

Für den zuverlässigen laufenden Betrieb stellt ASDW Core eine durchgängige Werkzeugkette bereit. Über die **Datenbankverwaltung** (**db_management**) pflegen Administratoren mehrere Datenbankprofile, optionale SSH-Tunnel für entfernte Datenbankserver, Strukturprüfungen über den Schema-Normalizer sowie lokale 1-Klick-Voll-Sicherungen (ZIP-Archiv aus Datenbank, Uploads und lokaler Konfiguration). Für die gezielte inhaltliche Pflege einzelner Datensätze bietet das fachliche Modul **dbeditor** eine komfortable Tabellenoberfläche mit ein- und ausblendbaren Spalten, lesbarer Auflösung von Fremdschlüsseln und schreibgeschützten Systemtabellen, die nur bewusst entsperrt werden können.

Auf Ebene der gesamten Plattform überwacht die **Modulverwaltung** (**module_management**) alle aktiven Bausteine, Versionsstände und Qualitätsprüfungen (z. B. Prüfung auf saubere Übersetzungsschlüssel und Design-Tokens). Die **Release-Verwaltung** (**release_management**) erzeugt daraus bereinigte Auslieferungspakete ohne Entwicklungs- oder Geheimdateien und entfernt inaktive Implementierungsvarianten. Wer mehrere abgeleitete Kundenprojekte betreut, steuert diese zentral in der **Flottenverwaltung** (**fleet_management**): Projekte lassen sich erkennen, klonen, per Push-Sync aus dem Master aktualisieren (mit Schutzsperre für unabhängige Projekte), über SFTP/FTPS/FTP mit lückenloser Delta-Historie ausrollen, als Smart-ZIP exportieren oder direkt zu GitHub synchronisieren.

3.5 Plattform-Erweiterungen: SaaS, Web-Push, KI-Schnittstelle und Video-Studio

Module: ***saas_core***, ***push_notifications***, ***mcp_management***, ***video_studio***

Vier spezialisierte Core-Module erweitern ASDW Core um moderne Plattformfunktionen:

- **saas_core** verwandelt das System in eine Mehrmandanten-Plattform mit 3-Tab-Cockpit (Mandanten, Tarife/Quoten, Gutscheine), automatischer Subdomain-Zuordnung, isolierten Mandanten-Datenbanken (Silo-Modell), Self-Service-Registrierung (auch über Google) sowie Stripe-Anbindung und lokalem Zahlungs-Simulationsmodus.
- **push_notifications** liefert ein zentrales Cockpit für verschlüsselte Web-Push-Mitteilungen an Desktop- und Mobilgeräte – inklusive Echtzeit-Kennzahlen, Geräte-Pausierung/Reset, Dual-OS-Vorschau-Simulator (Android/Windows und Apple iOS), Zustellungshistorie und voller Empfängerkontrolle.
- **mcp_management** öffnet angebundene Module kontrolliert für externe KI-Assistenten über das Model Context Protocol: mit benutzergebundenen SHA-256-Schlüsseln, optionaler IP-Freigabeliste, Modul-Matrix, Prepare-Only-Entwurfsschutz, gesondert freizuschaltenden Mengenlimits, fertigen Client-Konfigurationen und vollständigem Sicherheitsprotokoll.
- **video_studio** schließt die Lücke zwischen Software und Anwender-Begleitung: Direkt aus den vorhandenen Modulen entstehen KI-gestützte Drehbücher in vier Verständlichkeitsstufen, 16:9-PowerPoint-Decks samt Regiebuch (Weg 1, rein über PHP), automatisierte Motion-Videos im 16:9- und 9:16-Kurzvideo-Format für Shorts und Reels (Weg 2) sowie per Makro-Recorder aufgezeichnete Live-Screencasts (Weg 3) – auf Wunsch mit einem Klick direkt im `?`-Online-Hilfe-Drawer veröffentlicht.

---

4. Anwendungsszenarien aus der Praxis

Die folgenden vier **Beispielszenarien** veranschaulichen anhand durchgängiger, vollständig mit den vorhandenen Modulen abbildbarer Abläufe, wie die Bausteine von ASDW Core zusammenwirken. Alle genannten Rollen, Organisationen und Daten sind fiktive, nicht sensible Beispiele; es handelt sich um keine Kundenreferenzen und keine Berichte aus laufenden Produktiveinsätzen.

Beispielszenario 1: Vom Messekontakt zur ausgewerteten E-Mail-Kampagne

Ausgangssituation: Nach einer Fachtagung liegen rund 40 gedruckte Visitenkarten neuer Interessenten vor. Das Team möchte die Kontakte zeitnah erfassen, allen Ansprechpartnern eine zweisprachige, persönlich formulierte Nachfass-E-Mail senden und unzustellbare Adressen sowie Antworten geordnet nachbearbeiten.

Beteiligte Rollen: Vertriebsmitarbeiter (Kontakterfassung), Marketing-Redakteur (Vorlagen- und Kampagnenpflege), Teamleitung (Prüfung und Nachverfolgung).

Ziel beziehungsweise Aufgabe: Strukturierte Digitalisierung der Kontakte, Gestaltung einer einheitlichen E-Mail-Vorlage auf Deutsch und Englisch, kontrollierter Versand unter Berücksichtigung von Sperrlisten sowie vollständige Auswertung von Zustellstatus und Rückmeldungen.

Ablauf in nachvollziehbaren Schritten:

1. **Visitenkarten digitalisieren:** Der Vertriebsmitarbeiter fotografiert die Karten mobil im **Visitenkarten-Scanner** (`scan_business_card`). Die aktivierte KI-Texterkennung übernimmt Name, Unternehmen, E-Mail-Adresse und Telefonnummer; die Felder werden im mobilen Prüf-Dialog mit dem gespeicherten Kartenfoto abgeglichen und gespeichert.
2. **Zweisprachige Vorlage gestalten:** Der Marketing-Redakteur erstellt im **E-Mail-Editor** (`email_editor`) eine responsive Vorlage mit Bausteinen, fügt persönliche Platzhalter (Anrede, Nachname, Firma) ein, ergänzt die englische Sprachvariante (optional mit KI-Übersetzung unter Erhalt der Platzhalter), prüft die Darstellung im Posteingang-Simulator und sendet eine interne Test-E-Mail.
3. **Empfängerliste aufbauen und bereinigen:** Im **E-Mail-Versandmodul** (`email_delivery`) werden die Kontakte in eine neue Empfängerliste übernommen. Die Import-Pipeline erkennt doppelte Einträge automatisch; vor dem Versand gleicht das System alle Adressen gegen bestehende Blocklisten und Abmeldungen (*Opt-out*) ab.
4. **Kampagne gesteuert versenden:** Über den Kampagnen-Assistenten wählt der Redakteur Vorlage, Absender-Postfach und Empfängerliste aus und startet den Versand über die asynchrone Warteschlange mit begrenzter Versandrate.
5. **Ergebnisse und Rückläufer auswerten:** In der 360°-Versandhistorie prüft das Team den Versandstatus. Unzustellbare Adressen (*Bounces*), Abwesenheitsnotizen und direkte Rückfragen erscheinen klassifiziert im Inbound-Explorer; Tippfehler in E-Mail-Adressen lassen sich direkt korrigieren und per 1-Klick-Nachversand erneut zustellen. Relevante Aktionen werden automatisch im **Audit-Protokoll** (`log_management`) vermerkt.

Eingesetzte Module und Funktionen: `scan_business_card` (Foto-Upload, KI-OCR, Prüf-Dialog), `email_editor` (visueller MJML-Designer, Mehrsprachigkeit, Smart-Tokens, Posteingang-Simulator, Time-Machine-Versionierung), `email_delivery` (Empfängerlisten, Deduplizierung, Blocklisten, Kampagnen-Assistent, Warteschlange mit Rate-

Limiter, 360°-Log, Nachversand, Inbound-Explorer), `log_management` (Audit-Protokoll), `core` (Online-Hilfe).

Ergebnis: Alle Messekontakte sind einheitlich digitalisiert, persönlich angeschrieben und mit dokumentiertem Zustell- und Rückläuferstatus nachvollziehbar erfasst.

Daraus nachvollziehbar entstehender Nutzen: Da Kontakterfassung, Vorlagengestaltung, Versand und Rückläufer-Auswertung in einer gemeinsamen Umgebung stattfinden, entfallen manuelle Medienbrüche beim Übertragen von Vorlagen oder Protokollen. Gleichzeitig schützen Blocklisten-Abgleich, Versionierung und 360°-Protokoll vor Doppelversand und unbemerkten Zustellfehlern.

Voraussetzungen oder Grenzen: Erfordert konfigurierte SMTP- und IMAP-Zugangsdaten, laufende Hintergrund-Worker für Versand und Postfachabruf sowie – für automatische OCR-Erkennung, KI-Übersetzung und Stufe-3-Rückläufer-Analyse – einen hinterlegten Gemini- oder OpenAI-API-Schlüssel.

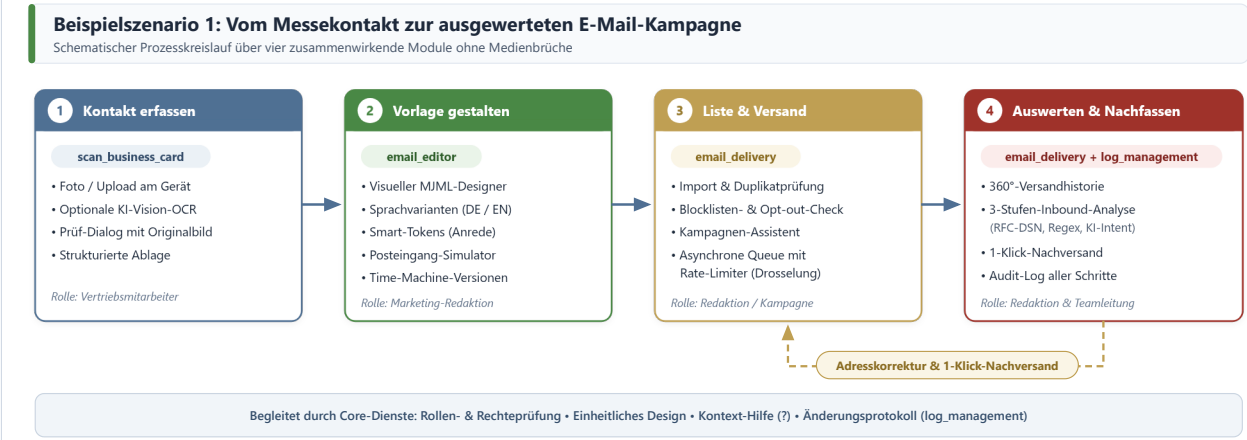


Abbildung 4 – Schematische Darstellung (Beispielszenario 1): Durchgängiger Prozesskreislauf von der Visitenkartenerfassung über Vorlagengestaltung und Versand bis zur Rückläufer-Auswertung.

Beispielszenario 2: Eine mandantenfähige Fachplattform einrichten und verwalten

Ausgangssituation: Ein Softwareanbieter möchte eine auf ASDW Core basierende Arbeitsplattform mehreren eigenständigen Firmenkunden als Web-Service anbieten. Jeder Kunde soll über eine eigene Subdomain zugreifen, sein eigenes Design wählen können und streng von den Daten anderer Kunden getrennt bleiben.

Beteiligte Rollen: Plattformbetreiber (Master-Administration), Mandanten-Administrator (Kundenorganisation), Fachanwender beim Mandanten.

Ziel beziehungsweise Aufgabe: Aufbau eines strukturierten Mandantenbetriebs mit klar definierten Tarifen, automatisierter Bereitstellung neuer Kundeninstanzen, einheitlichen Sicherheitsregeln und rollengerechter Oberfläche.

Ablauf in nachvollziehbaren Schritten:

1. **Tarife und Gutscheine definieren:** Im 3-Tab-Cockpit von **saas_core** legt der Plattformbetreiber passende Tarifstufen (z. B. *Starter* und *Team*) mit Benutzerlimits und Funktionsfreigaben an und hinterlegt bei Bedarf Aktions-Gutscheine.

2. **Neuen Mandanten bereitstellen:** Ein neuer Kunde registriert sich über die Self-Service-Maske (wahlweise klassisch oder per Google-Anmeldung) oder wird vom Betreiber angelegt. Das System provisioniert im Silo-Modell automatisch eine eigene, isolierte Datenbank und weist die Subdomain zu.
3. **Zahlungsabwicklung prüfen oder aktivieren:** Für erste Probeläufe nutzt der Betreiber den integrierten Offline-Simulationsmodus; für den Live-Betrieb wird der Stripe-Treiber mit Webhook-Signaturprüfung aktiviert.
4. **Markenauftritt und Sicherheit abstimmen:** In den zentralen Systemeinstellungen (**core**) werden Logo, Anwendungsname, Passwortrichtlinien, 2FA-Standard, Passkey-Freigabe, Inaktivitäts-Sperre und mehrsprachige System-E-Mails eingerichtet. Über **theme_management** und **translation_management** erhalten der Mandant und seine Nutzer das gewünschte Erscheinungsbild sowie Deutsch und Englisch als Arbeitssprachen.
5. **Benutzer, Gruppen und Menüs steuern:** Der Mandanten-Administrator legt über **user_management** und **permissions_management** Benutzerkonten und Fachgruppen an und ordnet über **menu_management** passende Startseiten und Menüstrukturen zu. Die **Sicherheits-Firewall** (`security_management`) setzt diese Grenzen bei jedem Aufruf durch.

Eingesetzte Module und Funktionen: `saas_core` (Mandanten-Cockpit, Silo-Provisionierung, Tarife/Quoten, Gutscheine, Stripe-/Mock-Billing, Onboarding), `core` (Systemeinstellungen, Branding, System-E-Mails, Governance), `user_management` (Konten, 2FA, Passkeys, Google-Login), `permissions_management` (Gruppenrechte), `security_management` (Endpunkt-Firewall, Sperrbildschirm), `menu_management` (Navigation & Startseiten), `theme_management` (Design-Vererbung), `translation_management` (Mehrsprachigkeit).

Ergebnis: Mehrere Kundenorganisationen arbeiten auf einer gemeinsamen Plattformbasis, jeweils mit eigener Subdomain, physisch getrennter Datenbank, eigenem Rollenmodell und kontrollierten Tarifleistungen.

Daraus nachvollziehbar entstehender Nutzen: Das Silo-Modell trennt Kundendaten bereits auf Datenbankebene sauber voneinander. Automatisierte Provisionierung, Tarif-Quoten und Self-Service-Anmeldung reduzieren manuelle Einrichtungsschritte, während zentrale Sicherheits- und Governance-Regeln für ein einheitliches Sicherheitsniveau sorgen.

Voraussetzungen oder Grenzen: Setzt aktivierten SaaS-Betrieb in der Systemkonfiguration, konfigurierte Subdomain-Auflösung, HTTPS sowie Datenbankrechte zur Erstellung neuer Mandanten-Datenbanken voraus. Live-Abrechnung benötigt ein Stripe-Konto; ein gemeinsamer Datenbankmodus für mehrere Mandanten in denselben Tabellen ist im aktuellen Stand nicht implementiert.

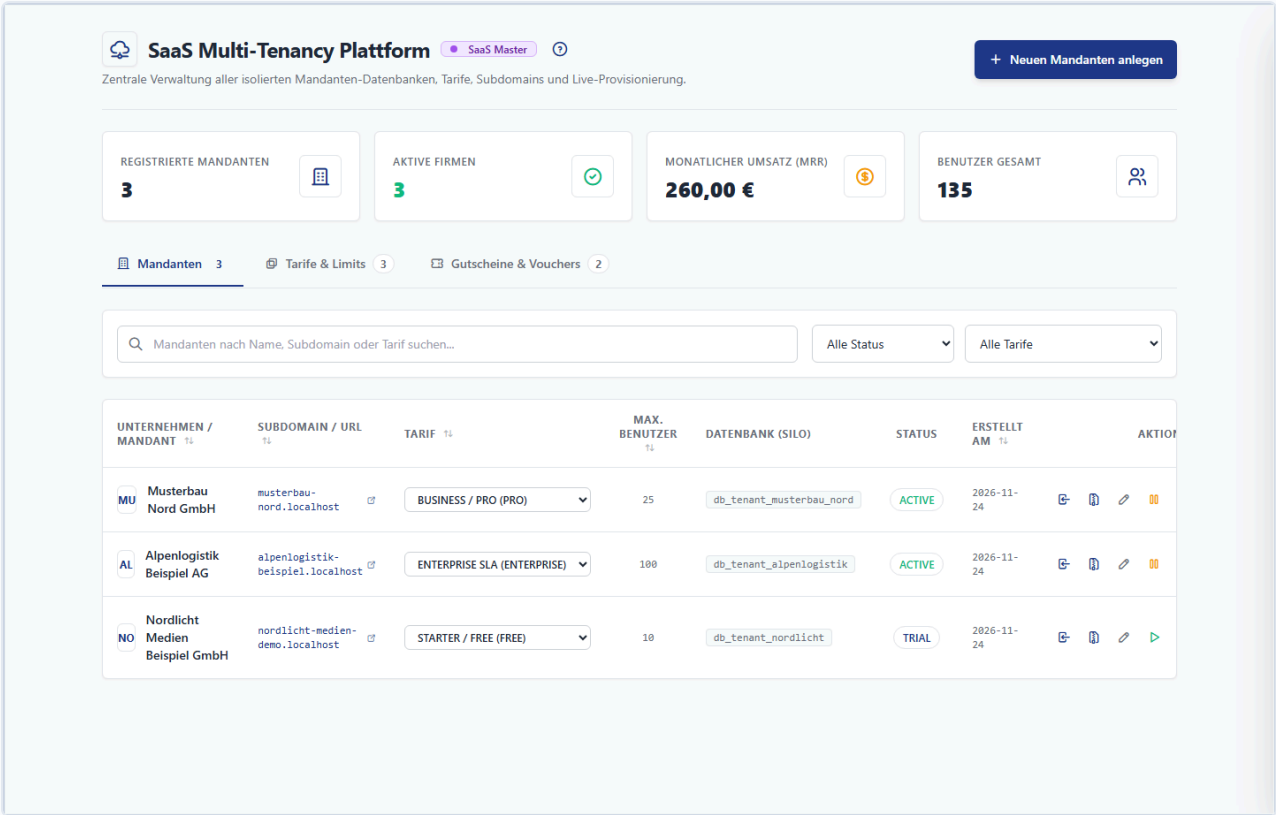


Abbildung 5 – Echte Bedienoberfläche: Verwaltung isolierter Mandanten (Silo-Datenbanken), Tarife und Gutscheine im SaaS-Cockpit von ASDW Core (fiktive Beispieldaten).

Beispielszenario 3: Mehrere abgeleitete Kundenprojekte zentral aktualisieren und absichern

Ausgangssituation: Ein Entwicklungsteam pflegt neben dem ASDW-Core-Master mehrere abgeleitete Kunden-Satellitenprojekte sowie eine unabhängige Website. Nach einer Weiterentwicklung am Systemkern sollen die Satelliten kontrolliert aktualisiert, vorab gesichert und anschließend auf Zielserver übertragen werden – ohne die unabhängige Website zu beschädigen.

Beteiligte Rollen: Systemadministrator, Projektverantwortlicher / Release-Manager.

Ziel beziehungsweise Aufgabe: Qualitätsprüfung und Bereinigung des Auslieferungsstands, vollständige lokale Datensicherung, geschützter Update-Abgleich auf Core-Satelliten, nachvollziehbares Server-Deployment sowie gezielte Datenkontrolle.

Ablauf in nachvollziehbaren Schritten:

1. **Projektbestand erfassen:** In der **Flottenverwaltung** (`fleet_management`) erkennt das System lokale Projektordner und unterscheidet klar zwischen abgeleiteten Core-Satelliten und unabhängigen Projekten.
2. **Qualität prüfen und Release bauen:** Über die **Modulverwaltung** (`module_management`) kontrolliert der Verantwortliche Versionsstände und Qualitäts-Heuristiken (Design-Tokens und Übersetzungsabdeckung). Anschließend erstellt die **Release-Verwaltung** (`release_management`) ein bereinigtes Auslieferungspaket ohne Entwicklungs-, Geheim- oder temporäre Dateien und ohne ungenutzte Modulvarianten.

3. **Voll-Sicherung anlegen:** Vor dem Update erzeugt der Administrator per Klick über **db_management** bzw. **fleet_management** ein lokales Voll-Backup (ZIP-Archiv aus SQL-Dump, Uploads und lokaler Konfiguration) für Master und Ziel-Satelliten.
4. **Push-Sync und Server-Deployment:** Die Kern-Aktualisierung wird gezielt per Push-Sync an die abgeleiteten Satelliten verteilt; für das unabhängige Projekt verhindert die eingebaute Schutzsperre ein versehentliches Überschreiben. Danach erfolgt die Übertragung auf den Zielserver per SFTP oder FTPS; Dateideltas, Laufzeit und Kommentar werden revisionssicher in der Deployment-Historie gespeichert.
5. **Datenbankprofile und Einzeldaten kontrollieren:** Bei Bedarf prüft der Administrator in **db_management** (inklusive optionalem SSH-Tunnel und Schema-Normalizer) die Tabellenstruktur und kontrolliert einzelne Datensätze gezielt im **Datenbank-Editor** (`dbeditor`), ohne Systemtabellen versehentlich zu verändern. Alle relevanten Schritte bleiben in **log_management** dokumentiert.

Eingesetzte Module und Funktionen: `fleet_management` (Projekterkennung, Push-Sync mit Schutzsperre, SFTP/FTPS/FTP-Deployment, Delta-Historie, Smart-ZIP, GitHub-Push, Voll-Backup), `module_management` (Metadaten-Sync, Qualitäts-Heuristik), `release_management` (bereinigter `dist`-Build, Build-Manifest), `db_management` (Profile, SSH-Tunnel, Schema-Normalizer, Voll-Backup), `dbeditor` (Tabellen-Browser, Fremdschlüssel-Auflösung, Systemtabellen-Schutz), `log_management` (Audit-Protokoll).

Ergebnis: Alle abgeleiteten Satellitenprojekte laufen auf einem konsistenten, dokumentierten Stand; Sicherungsarchive und Deployment-Protokolle liegen vollständig vor, und unabhängige Projekte blieben unberührt.

Daraus nachvollziehbar entstehender Nutzen: Wiederholbare Release-Builds, Schutzsperren für Nicht-Satelliten und eine protokollierte Deployment-Historie verringern das Risiko fehlerhafter manueller Datei-Uploads und machen jeden Auslieferungsstand nachvollziehbar.

Voraussetzungen oder Grenzen: Setzt lokalen Dateizugriff der Master-Instanz auf die verwalteten Projektordner sowie gültige Zugangsdaten für SFTP/FTPS/FTP, SSH-Tunnel oder GitHub voraus. Die lokale Voll-Sicherung dient als schneller Sicherungspunkt und ersetzt kein externes Rechenzentrums-Backup.

Beispielszenario 4: KI-Assistenten kontrolliert einbinden und Anwender aktiv begleiten

Ausgangssituation: Ein Team möchte seinen externen KI-Assistenten beim Entwerfen von E-Mail-Vorlagen und Kampagnen unterstützen lassen – jedoch strikt ohne das Risiko, dass eine KI eigenmächtig Massen-E-Mails versendet oder Daten löscht. Gleichzeitig sollen Mitarbeiter über Neuerungen informiert werden und kurze, verständliche Erklärvideos direkt in der Oberfläche vorfinden.

Beteiligte Rollen: Administrator (KI-Freigabe & Push-Steuerung), Marketing-Anwender (Arbeit mit KI-Assistent & Prüfung), Schulungs-/Dokumentationsverantwortlicher (Video-Studio), Fachanwender (Empfänger von Push-Mitteilungen und Online-Hilfe).

Ziel beziehungsweise Aufgabe: Einrichtung eines streng begrenzten KI-Zugangs über MCP, sichere Entwurfsarbeit mit menschlicher Endfreigabe, gezielte Information der Anwender per Web-Push und Bereitstellung verständlicher Hilfe-Clips direkt im Modul.

Ablauf in nachvollziehbaren Schritten:

1. **KI-Schlüssel mit klaren Leitplanken erstellen:** Im MCP-Cockpit (**mcp_management**) legt der Administrator einen API-Schlüssel an, der fest an das Benutzerkonto des Marketing-Anwenders gebunden ist. In der Modul-

Matrix werden Lese- und Entwurfsrechte für **email_editor** und **email_delivery** freigegeben; die Sonderfreigabe zum direkten Kampagnenstart bleibt gesperrt (oder strikt auf ein niedriges Empfänger-Limit begrenzt). Über den Konfigurationsgenerator kopiert der Anwender die fertige JSON-Einstellung in seinen MCP-Client.

2. **Entwurf durch KI, Freigabe durch Menschen:** Der KI-Assistent liest den Markenkontext (**core**), ruft vorhandene Platzhalter ab und legt im **email_editor** eine neue mehrsprachige Vorlage sowie in **email_delivery** einen Kampagnen-Entwurf an (*Prepare-Only*). Ein Testversand ist nur an freigegebene Einzeladressen möglich; jeder KI-Aufruf erscheint transparent im MCP-Sicherheitsprotokoll. Erst nach menschlicher Prüfung wird die Kampagne regulär freigegeben.
3. **Kurztutorial im Video-Studio produzieren:** Der Dokumentationsverantwortliche erzeugt im **Video-Studio** (`video_studio`) ein Drehbuch in der gewünschten Zielgruppenstufe (z. B. *Praxis-Einsteiger*), exportiert wahlweise ein 16:9-PowerPoint-Deck (Weg 1), rendert ein Kurzvideo im 9:16-Format für interne Kanäle (Weg 2) oder zeichnet den Ablauf als Live-Screencast auf (Weg 3) und veröffentlicht das Hilfe-Video per 1-Klick-Publishing direkt im `?`-Hilfe-Drawer des Moduls (**core**).
4. **Anwender per Web-Push informieren:** Über das **Web-Push-Cockpit** (`push_notifications`) prüft der Administrator das Erscheinungsbild der Mitteilung im Dual-OS-Simulator (Android/Windows und iOS) und sendet einen kurzen Hinweis samt Direktlink an alle Benutzer, die Push-Mitteilungen in ihrem Profil aktiviert haben. Die Zustellung lässt sich dreistufig im Protokoll nachverfolgen.

Eingesetzte Module und Funktionen: `mcp_management` (SHA-256-Schlüssel, Benutzerbindung, Modul-Matrix, Prepare-Only-Schutz, Mengenlimits, JSON-Generator, Sicherheitsprotokoll), `core` (Marken-Ressource, Online-Hilfe-Drawer mit Videoplayer), `email_editor` & `email_delivery` (MCP-Adapter), `video_studio` (4-Stufen-KI-Drehbücher, Weg 1–3, 1-Klick-Web-Help-Publishing), `push_notifications` (Rich-Push-Studio, Dual-OS-Simulator, Zustellhistorie, Benutzer-Self-Service).

Ergebnis: KI-Unterstützung arbeitet innerhalb klarer technischer Grenzen mit voller Nachvollziehbarkeit, und Anwender erhalten sowohl aktuelle Benachrichtigungen als auch direkt eingebettete Erklärvideos.

Daraus nachvollziehbar entstehender Nutzen: Die Bindung an reale Benutzerrechte, der Entwurfsschutz und das Sicherheitsprotokoll machen KI-Nutzung für Organisationen beherrschbar. Gleichzeitig senken direkt im Modul abrufbare Hilfe-Videos und gezielte Web-Push-Hinweise die Einstiegshürde bei neuen Funktionen.

Voraussetzungen oder Grenzen: MCP-HTTP-Gateway und Web-Push erfordern im Produktivbetrieb HTTPS; Web-Push muss per Konfigurationsschalter aktiviert und von jedem Empfänger aktiv zugelassen werden (unter iOS nach Installation auf dem Home-Bildschirm). Video-Studio Weg 2 und Weg 3 benötigen eine lokale Node.js-Render-Umgebung sowie für KI-Drehbücher/Sprachausgabe entsprechende KI-API-Schlüssel.

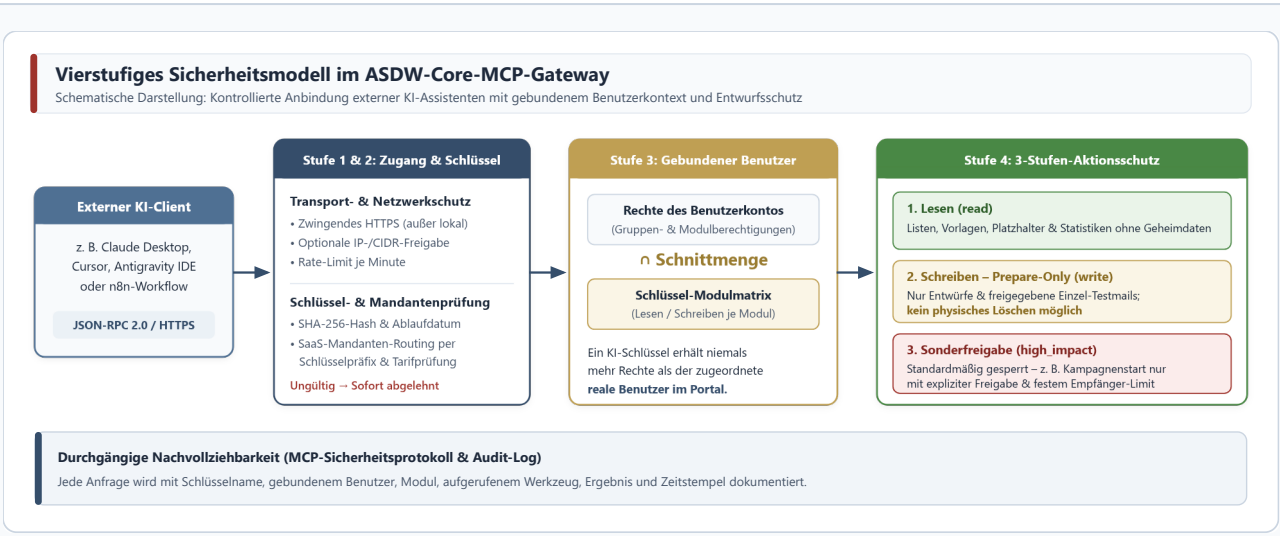


Abbildung 6 – Schematische Darstellung: Vierstufiges Sicherheitsmodell für kontrollierte KI-Zugriffe über das ASDW-Core-MCP-Gateway.

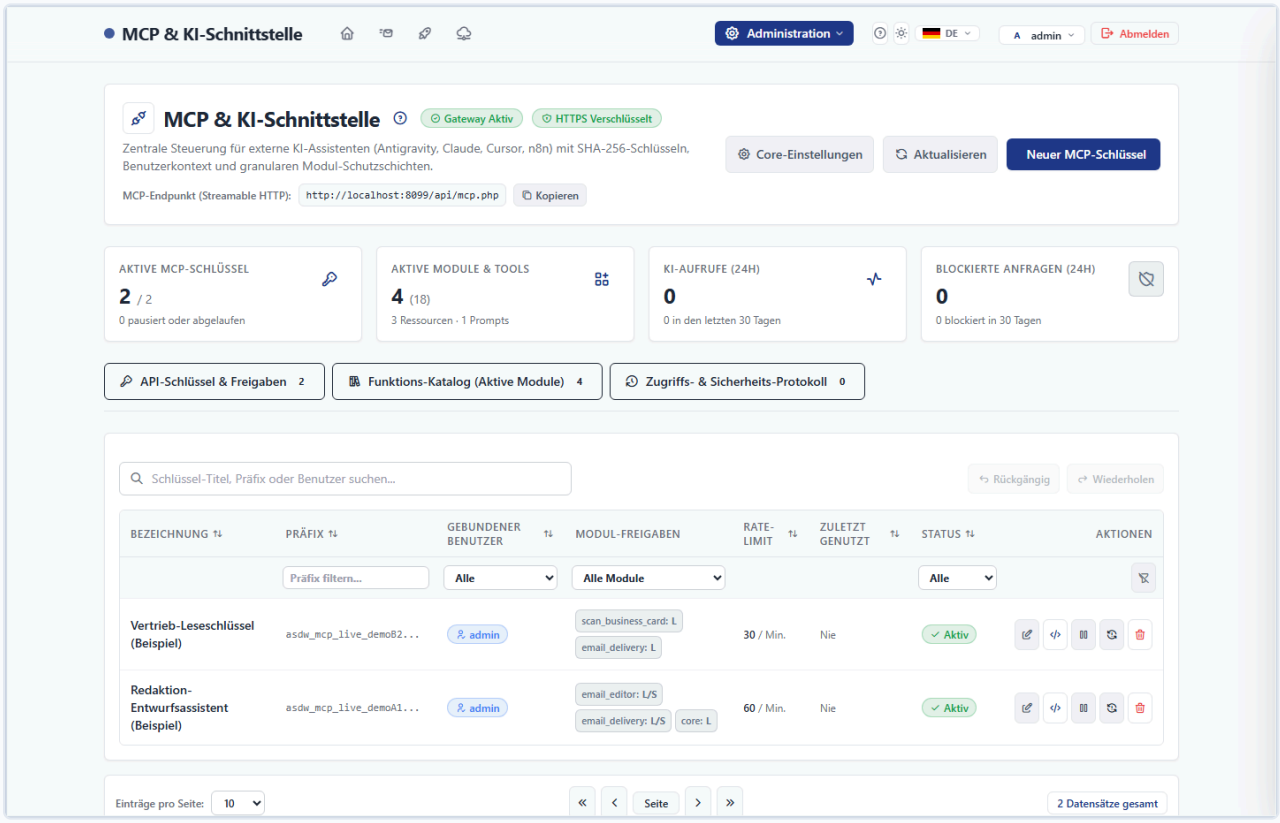


Abbildung 7 – Echte Bedienoberfläche: Verwaltung benutzergebundener KI-Zugriffsschlüssel, Modul-Freigaben, Funktionskatalog und Sicherheitsprotokoll im MCP-Cockpit (fiktive Beispieldaten, ohne produktive Schlüssel).

5. Konkrete Beispiele aus Bedienung und Konfiguration

Die folgenden drei Beispiele veranschaulichen typische Einstellungen und Bedienabläufe. Sämtliche Namen, E-Mail-Adressen und Werte sind **fiktive Beispieldaten**. Sie wurden aus dem untersuchten Systemumfang abgeleitet und stellen keine protokollierten Kundenprojekte dar.

Beispiel A: Zweischichtige Ansprache in einer mehrsprachigen E-Mail-Vorlage

Kategorie: Abgeleitet aus der vorhandenen Bedienoberfläche des Moduls `email_editor` (fiktive Beispieldaten).

In einer Kundeninformation soll jeder Empfänger korrekt angesprochen werden – auf Deutsch oder Englisch, je nach hinterlegter Sprache. Im visuellen Designer von **email_editor** legt der Redakteur eine Vorlage *Quartalsinformation* an und ergänzt neben der deutschen auch eine englische Sprachvariante:

- **Fiktiver Empfänger 1:** Anrede `Frau`, Nachname `Beispiel`, Sprache `de` → Die Smart-Token-Logik erzeugt automatisch: *„Sehr geehrte Frau Beispiel,“*
- **Fiktiver Empfänger 2:** Anrede unbekannt, Sprache `de` → Automatischer Rückfall auf eine neutrale Begrüßung: *„Guten Tag,“*
- **Fiktiver Empfänger 3:** Anrede `Mr.`, Nachname `Sample`, Sprache `en` → Das Versandmodul wählt die englische Vorlagenvariante samt englischer Betreffzeile und Preheader.

Über den **Posteingang-Simulator** prüft der Redakteur bereits vor dem Versand, wie Betreff und Vorschautext (Preheader) auf Desktop- und Smartphone-Bildschirmen abgeschnitten werden. Muss eine frühere Formulierung wiederhergestellt werden, genügt ein Klick in der *Time-Machine*-Versionshistorie.

Beispiel B: Kontrollierte KI-Freigabe im MCP-Cockpit

Kategorie: Abgeleitet aus der vorhandenen Bedienoberfläche des Moduls `mcp_management` (fiktive Beispieldaten).

Eine Organisation möchte einem Redaktions-Assistenten erlauben, Kampagnen vorzubereiten, aber nicht selbstständig zu versenden. Im MCP-Cockpit wird dazu ein Schlüssel nach folgendem Muster konfiguriert:

Einstellung (fiktives Beispiel)	Gewählter Wert	Wirkung im System
Schlüsselbezeichnung	`Redaktion-Entwurfsassistent`	Eindeutige Kennzeichnung im Zugriffsprotokoll
Gebundener Benutzer	`m.muster` *(Gruppe: Redaktion)*	Der KI-Client besitzt höchstens die Rechte dieses Kontos
Modul `email_editor`	Lesen: `Ja`, Schreiben: `Ja`	Vorlagen abrufen, Entwürfe anlegen/ändern, Einzel-Testmail an freigegebene Adresse
Modul `email_delivery`	Lesen: `Ja`, Schreiben: `Ja`	Listen/Statistiken lesen, Kampagnen als Entwurf anlegen (*Prepare-Only*)
Sonderfreigabe „Kampagnenstart“	`Gesperrt` *(alternativ z. B. Limit `25`)*	Eigenmächtiger Massenversand durch die KI ist technisch blockiert

Einstellung (fiktives Beispiel)	Gewählter Wert	Wirkung im System
Optionale IP-Freigabe & Limit	Firmen-IP / `60 Anfragen pro Minute`	Schutz vor Missbrauch und übermäßiger Abfragelast

Jeder einzelne Aufruf wird im Sicherheitsprotokoll mit dem gekoppelten Benutzerkontext festgehalten, sodass jederzeit nachvollziehbar bleibt, welche Eingabe über den Schlüssel erfolgte.

Beispiel C: Mehrstufige Design-, Sprach- und Rollensteuerung im Team-Portal

Kategorie: Schematische Veranschaulichung des Zusammenspiels von `core`, `theme_management`, `translation_management`, `permissions_management` und `menu_management` (fiktive Beispieldaten).

In einem gemeinsamen Portal arbeiten zwei Abteilungen mit unterschiedlichen Anforderungen:

1. **Systemstandard (`core` / `theme_management`):** Global ist das helle Theme *Corporate Clean* mit Deutsch als Standardsprache aktiv.
2. **Modulanpassung (`module_management`):** Für die Administration wird ein klar abgesetztes dunkles Theme gewählt, damit Administratoren sofort erkennen, wenn sie sich in sensiblen Verwaltungsansichten befinden.
3. **Benutzerebene (`user_management` / `translation_management`):** Eine englischsprachige Kollegin stellt in ihrem persönlichen Profil die Sprache auf *English* und wählt ihr bevorzugtes helles Theme *Nordic Frost* – ihre persönliche Auswahl überschreibt die Modul- und Systemvorgabe automatisch, ohne andere Benutzer zu beeinflussen.
4. **Sichtbarkeit (`permissions_management` / `menu_management`):** Die Fachgruppe *Kommunikation* sieht im Hauptmenü ausschließlich Dashboard, E-Mail-Editor, E-Mail-Versand und Visitenkarten-Scanner; die Systemeinstellungen, Flottenverwaltung und Datenbankwerkzeuge bleiben für diese Gruppe ausgeblendet und auf Serverebene gesperrt.

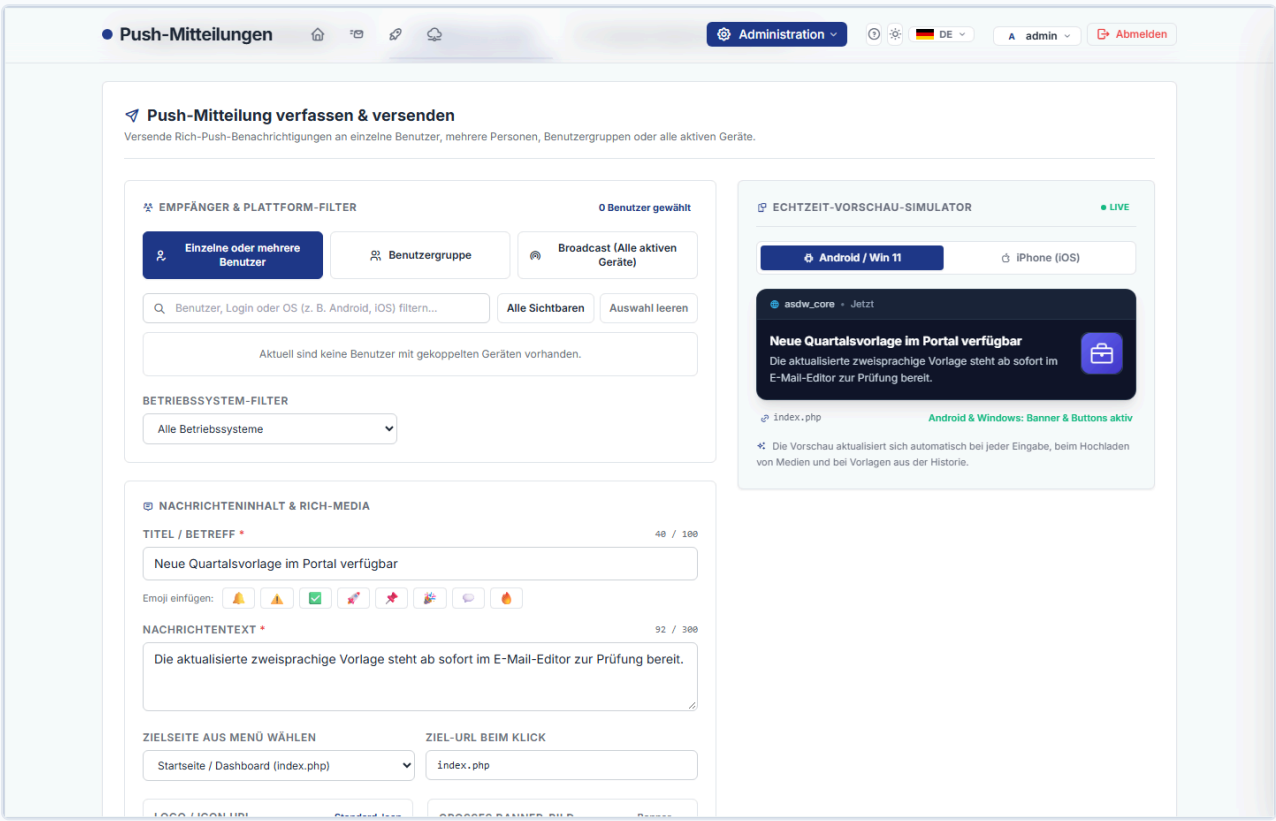


Abbildung 8 – Echte Bedienoberfläche: Zielgruppensteuerung, Nachrichteneditor und Dual-OS-Echtzeit-Vorschau-Simulator im Web-Push-Cockpit (fiktive Beispielmeldung).

---

6. Voraussetzungen, optionale Dienste und Grenzen

Eine transparente Bewertung erfordert neben dem Funktionsumfang auch den klaren Blick auf technische Voraussetzungen und Grenzen des untersuchten Projektstands.

Technische Grundvoraussetzungen

- **Webserver & Datenbank:** PHP 8 (inklusive Datenbank-, Grafik-, Verschlüsselungs-, Archiv- und Netzwerkerweiterungen) sowie MySQL oder MariaDB.
- **HTTPS im Produktivbetrieb:** Zwingend notwendig für PWA-Installation und Service Worker, Web-Push-Mitteilungen, Passkeys (WebAuthn/FIDO2), Scriptcase-IFrame-Cookies sowie das MCP-Gateway (unverschlüsselte HTTP-Zugriffe außerhalb lokaler Testrechner werden abgelehnt).

Optionale Dienste und Konfigurationsschalter

Viele weiterführende Funktionen sind bewusst über Konfigurationsschalter steuerbar und bleiben inaktiv, bis sie ausdrücklich eingeschaltet werden:

Funktionsbereich	Voraussetzung / externer Dienst	Standardverhalten / Hinweis
E-Mail-Versand & 2FA-Codes	SMTP-Versandserver (für Rückläufer zusätzlich IMAP-Postfach + Hintergrund-Worker)	Erforderlich für produktiven Versand und 2FA-Codes
KI-Funktionen (Visitenkarten-OCR, KI-Übersetzung, Preheader, Rückläufer-Stufe 3, Video-Drehbücher)	API-Schlüssel von Google Gemini und/oder OpenAI	Optional; Grundfunktionen der Module bleiben ohne KI-Schlüssel nutzbar
Google-Login & Google-Kontakte-Sync	Google Cloud OAuth 2.0 Zugangsdaten (für Kontakte zusätzlich 256-Bit-Verschlüsselungsschlüssel)	Google-Kontakte-Sync standardmäßig deaktiviert; gezielt freischaltbar
SaaS-Zahlungsabwicklung	Stripe-Konto (Live/Test) oder lokaler Simulationsmodus	Mehrmandantenbetrieb per Konfigurationsschalter steuerbar; Silo-Datenbankmodell
Web-Push-Mitteilungen	HTTPS, VAPID-Schlüssel (automatisch verwaltbar), Empfänger-Zustimmung	Standardmäßig deaktiviert; auf iOS über Home-Screen-App nutzbar
VCRM-Anbindungen	Zweites Datenbankprofil (`vcrm`) oder VCRM-REST-API	Standardmäßig deaktiviert; Scanner-Sync im SaaS-Betrieb gesperrt
Video-Studio Weg 2 & Weg 3	Lokale Node.js-Umgebung mit Remotion und Playwright	Weg 1 (16:9-PowerPoint-Export) läuft rein über PHP ohne Zusatzsoftware

Bewusste Grenzen und nicht belegte Bereiche

- **Kein Shared-Database-SaaS:** Im SaaS-Betrieb ist das Silo-Modell (eine physische Datenbank je Mandant) umgesetzt; ein gemeinsamer Datenbankmodus mehrerer Mandanten in denselben Tabellen existiert im aktuellen Code nicht.
- **Abgrenzung einzelner Spezialfunktionen:** Scriptcase-Dual-Mode-Implementierungen liegen für ausgewählte Module (`user_management` , `email_editor` , `email_delivery` , `scan_business_card` , `video_studio`) vor, nicht für alle 20 Module. Das Modul `dbeditor` besitzt aktuell keinen MCP-Adapter. Eine IP-/CIDR-Freigabeliste existiert gezielt im MCP-Gateway, nicht als allgemeine App-weite Firewall-Whitelist.
- **Keine Aussagen zu kommerziellen oder nicht geprüften Kennzahlen:** Diese Broschüre trifft bewusst keine Aussagen zu Lizenzmodellen, Preisen, Hosting-Konditionen, Support-Reaktionszeiten, formellen Datenschutz-Zertifizierungen, maximaler Last-Skalierbarkeit oder bezifferten Zeit- und Kostenersparnissen.

---

7. Zusammenfassende Orientierung

Nach dem untersuchten Projektstand eignet sich **ASDW Core** sachlich vor allem für folgende Einsatzfelder und Zielgruppen:

Zielgruppe / Bedarf	Passende Stärken von ASDW Core
Organisationen mit Bedarf an einem erweiterbaren Web-Portal	Einheitliche Basis aus 16 Core-Modulen für Benutzer, Rechte, Design, Mehrsprachigkeit, Sicherheit, Online-Hilfe und Protokollierung – sofort ergänzt um 4 Fachmodule
Teams mit regelmäßigem E-Mail- und Kontakt-Workflow	Geschlossene Kette aus Visitenkarten-Scanner, visuellem MJML-E-Mail-Designer, gesteuertem Kampagnenversand, 360°-Versandlog und Rückläufer-Auswertung
Betreiber mehrmandantenfähiger Web-Dienste	Integrierte SaaS-Plattform im Silo-Modell mit Subdomain-Erkennung, automatisierter Mandanten-Provisionierung, Tarifen, Quoten, Gutscheinen und Stripe-/Mock-Billing
Agenturen & Entwicklungsteams mit mehreren Kundenprojekten	Master-Satelliten-Flottenverwaltung mit Push-Sync-Schutzsperre, bereinigten Release-Builds, protokolliertem SFTP/FTPS-Deployment, Voll-Backup und optionaler Scriptcase-Brücke
Vorreiter im kontrollierten Einsatz von KI-Assistenten	Zentrales MCP-Gateway mit benutzergebundenen Schlüsseln, Entwurfsschutz (*Prepare-Only*), Mengengrenzen und lückenlosem Sicherheitsprotokoll
Verantwortliche für Anwender-Befähigung & Dokumentation	Integriertes Video-Studio für Drehbücher, PowerPoint-Decks, Shorts/Reels und Live-Screencasts mit direkter Veröffentlichung im `?`-Online-Hilfe-Fenster

ASDW Core bietet damit ein klar gegliedertes, modulares Fundament, das sich schrittweise an wachsende fachliche und organisatorische Anforderungen anpassen lässt – von der kompakten Einzelanwendung bis zur mandanten- und flottenfähigen Plattform.