Bücher Immobilien — was wir individuell entwickelt haben.

9 EIGENENTWICKLUNGEN · KEIN ZUGEKAUFTES PLUGIN

Kennzahlen auf einen Blick.

Individualentwicklung mit Qualitätssicherung — nicht nur Theme-Konfiguration.

9
Eigenentwicklungen

≈14.700
Zeilen eigener Code

≈420
Automatisierte Tests

3
WP-CLI-Kommandos

01 — WORDPRESS-PLUGIN

Bauvorhaben-Plugin.

Bündelt importierte Wohnungen automatisch zu Bauvorhaben-Seiten — z. B. alle Einheiten der Fraunhoferstraße 4–10 auf einer Projektseite, gruppiert nach Ort.

Fakten `plugins/bauvorhaben/ · ≈ 2.350 Zeilen + Templates · v2.4.0`

Kein manueller Pflegeaufwand — Zuordnung Wohnung → Bauvorhaben live aus der Adresse, importfest.

Automatische Kennzahlen — Preisspanne, Größen, Zimmer aus den Wohnungen aggregiert.

Die Unterüberschrift schreibt sich selbst — ohne Eingabe entsteht „Fraunhoferstraße 4–10" aus den Hausnummern der aktuellen Einheiten.

Elementor-Integration — eigene Dynamic-Tags-Gruppe „Bauvorhaben", Übersichten per Loop Grid ohne Code.

Natives Houzez-Look&Feel — Wohnungsliste nutzt Houzez' Karten-Templates inkl. Slider, Favoriten, Vergleich.

Markup ohne PHP änderbar — Übersicht und Kurzliste rendern aus HTML-Templates mit Platzhaltern; Theme-Override und Filter vorgesehen.

01 — WORDPRESS-PLUGIN · REDAKTIONSPROZESS

Nichts geht ungefragt live.

Ein frisch importiertes Bauvorhaben legt seine Seite selbst an — und stand damit bisher sofort öffentlich, ohne Überschrift, Teaserbild und Text. Jetzt startet es unsichtbar, und die Redaktion erfährt davon.

Fakten

`bauvorhaben.php` — Statusmodell, Benachrichtigung, Dashboard-Widget.

Neu heißt inaktiv — automatisch angelegte Bauvorhaben bleiben unsichtbar, bis ein Mensch sie freigibt.

Benachrichtigung statt Zufall — legt der Import ein neues Bauvorhaben an, geht eine Mail an die Redaktion.

Prüfen vor dem Freigeben — inaktive Seiten antworten öffentlich mit 404, sind für die Redaktion aber unter ihrer echten URL erreichbar.

Dashboard-Widget „Stand der Bauvorhaben" — neu, aktiv, inaktiv auf einen Blick, direkt nach dem Einloggen.

Abgleich auf der Bauvorhaben-Seite — dort stehen letzte Bauvorhaben-Aktualisierung und letzter Immobilien-Import nebeneinander, samt Auslöser und Anzahl.

Redaktions-Komfort — Media-Uploader, Admin-Spalte „Wohnungen (live)", Projektvideo mit eigenem Player.

02 — KARTEN-FRONTEND

Eigene Kartenlösung mit Clustering.

Ersetzt die Standard-Karte der Immobiliensuche durch Mapbox GL im eigenen Kartenstil — mit selbst entwickeltem Clustering für Wohnungen desselben Hauses.

Fakten `custom-half-map.js · ≈ 850 Zeilen · eigener Mapbox-Kartenstil`

Adressbasiertes Clustering — mehrere Einheiten an einer Adresse werden zu einem Zähler-Marker.

Cluster-Popups mit Mehrwert — nach Preis sortierte Liste, gehovertes Listing rückt nach oben.

Marker-Spreading — Gebäude mit identischen Koordinaten werden kreisförmig auseinandergelegt.

Robuste Datenprüfung — ungültige Koordinaten verworfen; Preis-Parser versteht deutsches, US- und Kurzformat.

Sauberes Verhalten — initiales Auto-Zoom, danach bleibt der Nutzer-Ausschnitt erhalten; AJAX-Anbindung an die Houzez-Suche.

03 — NAVIGATION

Dynamisches Städte-Menü.

Das „Immobilien"-Menü pflegt sich selbst: Die Unterpunkte entstehen automatisch aus allen Orten, in denen aktuell Objekte im Angebot sind.

Fakten `child-theme/functions.php` — selbstpflegend über den Import.

Null Redaktionsaufwand — neue Stadt im Import erscheint automatisch; verschwindet das letzte Objekt, verschwindet der Menüpunkt.

Direkt in die Kartensuche verlinkt — jeder Eintrag führt auf die vorgefilterte Suche mit der eigenen Half-Map-Karte.

Flexibel platzierbar — Platzhalter-Menüpunkt mit CSS-Klasse, in jedem Menü und an beliebiger Position.

Alphabetisch sortiert — nur Orte mit tatsächlich verfügbaren Objekten.

04 — CHILD-THEME

Child-Theme „buecher-immobilien".

Sammelt alle kundenspezifischen Anpassungen update-sicher an einem Ort — vom Print-Exposé bis zum Frontend-Werkzeugkasten.

Fakten `themes/buecher-immobilien/ · functions.php + lib.js + Overrides ≈ 3.800 Zeilen`

Print-Exposé — Druckansicht mit Bücher-Logo, QR-Code zum Online-Objekt, Grundrissen, Energieklasse, Galerie.

Energieausweis-PDFs öffentlich — Downloads ohne Login-Zwang, Analyse der gesamten Import-Datenkette.

Migrations-Absicherung — Laufzeit-Rewrite von Alt-Domain-URLs; PDFs funktionieren auf jeder Umgebung.

Frontend-Werkzeugkasten (lib.js, ≈ 2.100 Zeilen) — deutsche Preis-Slider, Objekt-ID in Formularen, Bild-Fallbacks, $→€-Korrekturen, Popups.

Aufgeräumtes Backend — „Theme Options" als Unterpunkt ins Houzez-Menü verschoben.

05 — FRONTEND-DETAIL · MU-PLUGIN

Nachhaltigkeits-Badge „QNG".

Das QNG-Logo „Nachhaltiges Gebäude" liegt als Siegel über den Vorschau- und Kopfbildern — aber nur bei den Einheiten, die tatsächlich zertifiziert sind.

Fakten `buecher-qng-badge.php · ≈ 460 Zeilen · v1.3.1`

Nur wo es stimmt — im Bauvorhaben Königshofener Straße tragen 12 und 14 das Siegel; 16, 18 und 20 gehören zum Projekt, sind aber nicht zertifiziert.

Projektbilder standardmäßig aus — Übersichts- und Kopfbild des ganzen Bauvorhabens sind getrennt schaltbar, damit das Siegel nicht für Unzertifiziertes wirbt.

Ohne Eingriff ins Template — ein Inline-Style, ein Inline-Script. Keine Änderung an Bildern, Slider oder OpenImmo-Import.

Ein Container für alle Ansichten — Liste, Grid, Half-Map und „ähnliche Objekte" nutzen denselben Bildcontainer; eine Regel deckt alles ab.

Position aus der Messung, nicht aus dem Gefühl — auf der Objektseite sitzt das Badge unten links, weil oben Teilen/Drucken/Merken und die Medien-Tabs liegen.

06 — SYSTEMREPARATUR · MU-PLUGIN

OpenImmo Geo-Fix.

Alle Immobilien-Marker lagen plötzlich „im Atlantik". Wir haben die Ursache bis in die Fremdplugin-Bibliothek nachgewiesen und dauerhaft behoben — inklusive Reparatur des Bestands.

Fakten

`fix-oi2wp-geo.php · ≈ 860 Zeilen` — 61 Tests, gegen den echten Objektbestand verifiziert.

Root-Cause-Analyse — Feed liefert Exponentialschreibweise, das Import-Plugin schneidet den Exponenten ab: aus Wiesbaden wird der Golf von Guinea.

Live-Fix — Hook nach jedem Import schreibt Original-Koordinaten zurück; kann Importe nie abbrechen.

Bestands-Reparatur — Backfill aus der Original-XML, per Admin-UI mit Testlauf oder WP-CLI.

Rettet sogar Feed-Fehler — im Feed selbst korrupte Objekte (auch live falsch!) per Heuristik korrigiert.

Datenschutz & idempotent — geblurrte Adressen bleiben unangetastet; bleibt nach Upstream-Fix folgenlos.

07 — BETRIEBS-MONITORING · MU-PLUGIN

Import-Monitor.

Der nächtliche OpenImmo-Import lief bisher unsichtbar. Der Monitor macht ihn im gesamten Backend live sichtbar — niemand muss mehr raten, ob er läuft, hängt oder fertig ist.

Fakten

`oi2wp-import-monitor.php · ≈ 1.100 Zeilen · v1.4.0` — 136 automatisierte Checks.

Status-Ampel in der Admin-Leiste — grün/gelb/rot auf jeder Backend-Seite, mit Zähler „X von Y".

Dashboard-Widget — Details, Tempo (Objekte/min), Restzeit-Schätzung, die letzten 10 Läufe.

Hänger-Erkennung — meldet automatisch, wenn 5 Minuten kein Fortschritt kommt; ein Lauf-Token unterscheidet echte Neustarts von fortgesetzten Läufen.

Status als Datei nach außen — atomar geschriebene JSON, alle 5 Minuten aufgefrischt. Dashboards unterscheiden dadurch „kein Import aktiv" von „Seite tot".

Rein lesend & abgesichert — greift nie in den Import ein; zusätzlich ein WP-CLI-Status-Kommando.

08 — BETRIEBS-STATUS · WEBROOT

Statusseite ohne WordPress.

Eine einzelne Datei im Webroot beantwortet die Betriebsfragen: Laufen die Importe? Ist das Zertifikat gültig? Antwortet der Server sich selbst? Sie bootet WordPress bewusst **nicht** und bleibt dadurch auch dann bedienbar, wenn WordPress klemmt.

Fakten

`status.php · ≈ 5.400 Zeilen` — 85 Szenariotests + 111 Logik-Checks.

Fünf Prüfungen, ein Urteil — Importe, SSL-Zertifikat, Loopback/cURL, Plugins, Monitor & Datenquelle. Der schlechteste Einzelbefund bestimmt das Banner.

Das Urteil steht auch im Seitentitel — „OK / Warnung / Problem" im Browser-Tab; externes Monitoring kann allein den Titel auswerten.

Fristband für das Zertifikat — der einzige Befund mit festem Termin bekommt eine eigene Zeile mit Countdown und Datum, sobald es eng wird.

Zählt, statt zu deuten — „3 Probleme gefunden" plus die betroffenen Bereiche. Bei mehreren Befunden verdeckt keiner die anderen.

Keine Abhängigkeiten, keine Kundendaten — liest die Status-JSON direkt vom Dateisystem, zeigt nur Zähler, Zeiten und Versionen.

09 — AUSFALLSICHERUNG · MU-PLUGIN

Formular-Eingang.

Anfragen liefen bisher nur per E-Mail und ins Houzez-CRM. Ohne Datenbank- oder Shell-Zugang war nicht nachweisbar, ob eine Anfrage überhaupt angekommen ist. Der Eingangskorb legt jede Absendung zusätzlich im Backend ab.

Fakten `buecher-form-inbox.php · ≈ 670 Zeilen · v1.0.2` — 31 Standalone-Checks.

Rein additiv — hängt sich erst nach dem Mailversand ein. E-Mail und CRM bleiben unverändert, es entstehen keine eigenen Tabellen.

Alle Formulare abgedeckt — Kontakt, Anfrage, Agent, Realtor und Besichtigungstermin.

Erfasst, was das CRM verliert — die drei Zusatzwünsche Exposé, Rückruf und Termin landen als eigene Felder statt als angehängter Fließtext.

Diagnose ohne SQL — prüft CRM-Tabellen, Admin-Benutzer und Dashboard-Seite und deckt die Falle auf, dass Leads ohne passenden Benutzer im Houzez-Dashboard unsichtbar bleiben.

Kann nichts kaputtmachen — vollständig gekapselt; eine Absendung kann daran nie scheitern. Dazu ein CSV-Export für Excel.

cicero

ARBEITSWEISE & QUALITÄT

01	**WordPress Coding Standards, phpcs-geprüft.**
02	**Konsequentes Escaping, Sanitizing, Nonces.**
03	**Import-sicher: alle Hooks in try/catch gekapselt.**
04	**≈ 420 Tests; Geo-Fix gegen den echten Objektbestand, Statusseite gegen einen echten Webserver verifiziert.**
05	**Dokumentiert: READMEs je Werkzeug + zentrale Projekt-Doku.**
06	**Offene Punkte festgehalten: Bug-Report Hersteller, Feed-Fehler an Datenlieferant.**
07	**Ausfallsicher gedacht: Statusseite und Formular-Eingang arbeiten weiter, wenn WordPress oder das CRM klemmen.**

cicero

EINORDNUNG

Das war der technische Teil.

Die neun Eigenentwicklungen sind das, was man im Code sieht. Die Website ist daneben in gleichem Maß eine konzeptionelle, gestalterische und redaktionelle Leistung — von der ersten Struktur bis zum laufenden Betrieb.

Konzeption — Seitenstruktur, Navigation und Nutzerführung, Aufbau der Immobiliensuche, Abstimmung mit Vertrieb und Datenlieferant.

Design — Gestaltung aller Seitentypen von Startseite bis Objektdetail, Bildsprache, Responsive-Verhalten bis zum Print-Exposé.

Branddesign — Farb- und Typosystem, Anwendung der Marke Bücher Immobilien über alle digitalen Berührungspunkte hinweg.

Inhalte — Texte und Bildauswahl, redaktionelle Betreuung und laufende Pflege der Seiten im Tagesgeschäft.

Deployment — Einrichtung der Umgebungen, Umzug auf die Live-Domain, laufende Updates.
Monitoring und Support sind als weitere Services optional buchbar.

Danke.

Fragen gerne jetzt.