

API-Schnittstelle (Web-to-CRM)

API-Schnittstelle: Web-to-CRM Integration

Die CloudCRM API ist das digitale Bindeglied zwischen Ihrer eigenen Webseite und Ihrer Unternehmensverwaltung. Sie ermöglicht es, externe Datenquellen (wie Kontaktformulare, Landingpages oder Onlineshops) direkt mit Ihrem CRM zu verknüpfen. Eingehende Anfragen werden in Echtzeit verarbeitet, sicher gespeichert und lösen sofortige In-App-Benachrichtigungen an Ihr Team aus.

1. Der API-Endpoint (Die Adresse)

Alle Daten müssen per **HTTP POST** an die folgende zentrale Adresse gesendet werden:

```
https://cloudcrm.dwoberland.de/api/v1/lead.php
```

2. Authentifizierung

Damit das System die eingehenden Daten Ihrem Unternehmen zuordnen kann, muss bei jeder Anfrage Ihr persönlicher **API-Key** mitgesendet werden. Sie haben hierzu zwei Möglichkeiten:

- **Option A (Empfohlen):** Übermittlung sicher im HTTP-Header als `X-API-KEY`.
- **Option B (Alternative):** Als URL-Parameter am Ende des Links:

```
.../lead.php?key=IHR_API_KEY
```

3. Unterstützte Datentypen (type)

Das System erkennt anhand des Feldes `"type"` im gesendeten JSON, wie die Daten im CRM verarbeitet werden sollen:

Typ: Web-Bestellung (`"type": "order"`) Verwenden Sie diesen Typ für Verkäufe oder verbindliche Buchungen auf Ihrer Webseite.

- **Speicherort:** Die Daten werden im Dashboard unter "Web-Bestellungen" (Posteingang) zur weiteren Prüfung abgelegt.
- **Team-Info:** Alle Mitarbeiter Ihres Unternehmens erhalten sofort eine Push-Benachrichtigung im CRM: *"Neue Bestellung! Eine Web-Bestellung über X,XX € ist eingegangen."*

Typ: Kontaktanfrage / Support (`"type": "ticket"`) Verwenden Sie diesen Typ für allgemeine Kontaktanfragen oder Support-Formulare.

- **Speicherort:** Erstellt automatisch ein neues Ticket in Ihrem Ticketsystem.
- **Intelligente Zuordnung:** Erkennt das CRM die übermittelte E-Mail-Adresse eines bereits existierenden Kunden, wird das Ticket automatisch in dessen digitaler Kundenakte hinterlegt.
- **Fallback:** Fehlt in der Datenübertragung ein Betreff (`subject`), wird das Ticket automatisch unter dem Titel "Anfrage Webseite" angelegt.

4. Technisches Format (JSON)

Die API erwartet den Daten-Payload zwingend im JSON-Format (Content-Type: `application/json`). Hier ist ein Beispiel für eine vollständige Übertragung:

JSON

```
{
  "type": "ticket",
  "email": "max.mustermann@beispiel.de",
  "subject": "Interesse an Beratung",
  "message": "Hallo, ich hätte gerne ein Angebot für Ihre Dienstleistungen.",
  "total": 0.00
}
```

Datenfelder im Überblick:

Feld	Pflicht?	Beschreibung
<code>type</code>	Ja	Entweder <code>"order"</code> oder <code>"ticket"</code> .
<code>email</code>	Ja	E-Mail-Adresse des Absenders (wird zur intelligenten Kundenzuordnung genutzt).
<code>message</code>	Ja	Der eigentliche Textinhalt der Anfrage (nur bei Tickets zwingend).
<code>subject</code>	Nein	Der Titel des Tickets oder der Bestellung.
<code>total</code>	Nein	Nur bei <code>order</code> : Der Bruttobetrag der Bestellung als Zahl (z.B. <code>49.90</code>).

5. Rückgabewerte & Fehler

Nach dem Absenden der Daten liefert die API eine Antwort (HTTP Status Code) im JSON-Format zurück:

- **Erfolg (200 OK):** `{"status": "success", "message": "Daten verarbeitet", "type": "..."}`

- **Format-Fehler (400):** Wenn das JSON ungültig ist oder das Pflichtfeld `type` fehlt.
 - **Fehlender Key (401):** Wenn der API-Key nicht mitgesendet wurde.
 - **Ungültiger Key (403):** Wenn der API-Key falsch ist oder im System deaktiviert wurde.
 - **Systemfehler (500):** Ein interner Serverfehler (z. B. Datenbankverbindung).
-

Revision #2

Created 2026-04-06 16:36:08 UTC by Alexander Paetzoldt

Updated 2026-05-09 14:16:47 UTC by Alexander Paetzoldt