

# PayloadCMS

**PayloadCMS** ist ein modernes, Headless CMS, das vollständig auf **Node.js** basiert und speziell für Entwickler:innen konzipiert ist. Es kombiniert ein leistungsfähiges Admin-Panel mit einem *code-first*-Ansatz, wodurch Inhalte, Strukturen und Logiken direkt im Code definiert werden können.

## Überblick

Eigenschaft	Beschreibung
CMS-Typ	Headless CMS
Backend	Node.js + Express
Sprache	TypeScript (auch JavaScript möglich)
API-Schnittstellen	REST und GraphQL
Admin-Oberfläche	Automatisch generiert aus dem Code
Authentifizierung	Integriert (JWT, Sessions, Role-based Access)
Datenbanken	MongoDB (Standard), <b>PostgreSQL / SQLite</b> via Kysely (experimentell)
ORM/Query Builder	Mongoose (MongoDB) / Kysely (SQL-DBs wie SQLite)

## Code-First Schema

Alle Collections (Inhaltstypen) werden im Code als JavaScript/TypeScript-Objekte definiert.

```
import { CollectionConfig } from 'payload/types';

const Posts: CollectionConfig = {
  slug: 'posts',
  fields: [
    { name: 'title', type: 'text', required: true },
    { name: 'content', type: 'richText' },
  ],
};
```

```
],  
};  
  
export default Posts;
```

## Authentifizierung & Rollen

- Integrierte Benutzerverwaltung
- Rollenbasierte Zugriffskontrolle (Access Control Policies)
- Auth-Collection konfigurierbar

## Dateiuploads & Medien

- Unterstützung für File-Uploads (lokal oder via Cloud)
- Optimierung und Vorschau automatisch im Admin-Panel

## Hooks & Middleware

- Asynchrone Hooks vor/nach Aktionen
- Business-Logik z. B. bei beforeChange, afterDelete, etc.

## Mehrsprachigkeit

- Unterstützung für i18n (lokalisierte Inhalte)

## Datenbankunterstützung

Datenbank	Standard?	ORM / Query Layer	Hinweise
MongoDB	☐	Mongoose	Reif & empfohlen
SQLite	☐☐ (ab 1.12+)	Kysely	Gut für lokale Dev

Datenbank	Standard?	ORM / Query Layer	Hinweise
PostgreSQL	☐☐ (ab 1.12+)	Kysely	Für produktive SQL-Setups

---

Revision #1

Created 23 April 2025 08:49:37 by Marius Klein

Updated 23 April 2025 08:53:43 by Marius Klein