

✂️ نظام إدارة محتوى (CMS) للمقالات

📌 مقدمة نظام إدارة محتوى (CMS) للمقالات يعتمد على واجهة API فقط، يتيح للمستخدمين إدارة المقالات والتصنيفات، مع دعم البحث، التفاعل، تحسين الأداء، وتأمين البيانات.

⚙️ تفاصيل المشروع

1. إدارة المستخدمين والصلاحيات

- نظام تسجيل دخول وتسجيل مستخدمين باستخدام JWT Authentication.
- دعم OAuth2 للربط بحسابات خارجية مثل Google.
- نظام أدوار وصلاحيات (Roles & Permissions) يشمل :
 - المدير (Admin): إدارة جميع المستخدمين، المقالات، التصنيفات، والإعدادات.
 - المحرر (Editor): إضافة وتعديل وحذف المقالات.
 - الكاتب (Author): إضافة مقالات جديدة وتعديل مقالاته الخاصة.
 - المستخدم العادي (User): قراءة المقالات والتعليق عليها فقط.
- تأمين الحساب باستخدام :
 - Rate Limiting للحماية من Brute Force Attacks.
 - 2FA Authentication.
 - إشعارات عند تسجيل الدخول من جهاز جديد.

2. إدارة المقالات والتصنيفات

- المقالات تحتوي على :
 - العنوان، المحتوى، التصنيف، الكلمات المفتاحية، الحالة، عدد المشاهدات، إحصائيات القراءة.
 - دعم تحميل وضغط الصور تلقائيًا.
 - البحث المتقدم باستخدام Elasticsearch أو Meilisearch.

- دعم التصنيفات الفرعية ونظام الوسوم (Tags)

3. البحث والتصنيف المتقدم

- دعم البحث الذكي داخل المقالات.
- تصنيف المقالات حسب الشعبية، الأحدث، الأعلى تقييمًا.
- تقديم توصيات بناءً على سجل قراءة المستخدم.

4. التفاعل والتعليقات

- دعم التعليقات المتداخلة (Nested Comments)
- إمكانية الإعجاب وعدم الإعجاب على المقالات والتعليقات.
- الإبلاغ عن التعليقات المخالفة.

5. تحسين الأداء

- استخدام Redis Caching لتسريع تحميل المقالات.
- ضغط وتحسين الصور تلقائيًا.
- تحسين SEO عبر JSON-LD و Meta Tags.

6. توثيق API

- استخدام Swagger أو Postman.
- توفير ملف Postman Collection.

7. تسجيل النشاطات والمراقبة

- تسجيل جميع العمليات التي يقوم بها المستخدمون باستخدام Activity Log.
- لوحة تحكم لمراقبة المقالات الأكثر زيارة وأكثر المستخدمين تفاعلًا.

✦ هيكلية الجداول (Database Schema)

لا يشترط الاعتماد عليها وإنما هيا للتوضيح فقط

- جدول المستخدمين (users)

- id: INT (Primary Key)
- name: VARCHAR(255)
- email: VARCHAR(255) UNIQUE
- password: VARCHAR(255)
- role: ENUM('admin', 'editor', 'author', 'user')
- created_at: TIMESTAMP
- updated_at: TIMESTAMP

- جدول المقالات (articles)

- id: INT (Primary Key)
- title: VARCHAR(255)
- content: TEXT
- category_id: INT (Foreign Key)
- featured_image: VARCHAR(255)
- tags: JSON
- views_count: INT DEFAULT 0
- status: ENUM('Draft', 'Published', 'Archived')
- created_at: TIMESTAMP
- updated_at: TIMESTAMP

- جدول التصنيفات(categories)

- id: INT (Primary Key)
- name: VARCHAR(255)
- parent_id: INT (Foreign Key)
- created_at: TIMESTAMP
- updated_at: TIMESTAMP

- جدول التعليقات(comments)

- id: INT (Primary Key)
- article_id: INT (Foreign Key)
- user_id: INT (Foreign Key)
- content: TEXT
- parent_id: INT (Foreign Key)
- created_at: TIMESTAMP
- updated_at: TIMESTAMP

Endpoints (واجهات API)

1. إدارة المستخدمين

- POST /api/register تسجيل مستخدم جديد
- POST /api/login تسجيل الدخول
- POST /api/logout تسجيل الخروج
- POST /api/reset-password إعادة تعيين كلمة المرور
- POST /api/verify-2fa المصادقة الثنائية

2. إدارة المقالات

- GET /api/articles عرض جميع المقالات
- GET /api/articles/{id} عرض مقالة محددة
- POST /api/articles إنشاء مقالة جديدة
- PUT /api/articles/{id} تعديل مقالة
- DELETE /api/articles/{id} حذف مقالة

3. إدارة التصنيفات

- GET /api/categories عرض التصنيفات
- POST /api/categories إنشاء تصنيف جديد

4. البحث والتصنيف

- GET /api/search?q=keyword البحث عن مقالات
- GET /api/articles/popular المقالات الأكثر مشاهدة
- GET /api/articles/recommended المقالات المقترحة

5. التعليقات والتفاعل

- POST /api/comments/{article_id} إضافة تعليق
- PUT /api/comments/{id} تعديل تعليق
- DELETE /api/comments/{id} حذف تعليق
- POST /api/comments/{id}/like إضافة إعجاب
- POST /api/comments/{id}/report الإبلاغ عن تعليق

6. تحسين الأداء

- GET /api/cache/articles عرض المقالات المخزنة في الكاش
- DELETE /api/cache/articles حذف كاش المقالات

7. تسجيل النشاطات

- GET /api/logs عرض جميع الأنشطة
- GET /api/analytics عرض الإحصائيات العامة