

Laravel + AI + PDF Reports

AI Contract & Document Analyzer

خطة تنفيذ مشروع منصة ذكية لتحليل العقود والملفات باستخدام Laravel والذكاء الاصطناعي، مع رفع ملفات PDF و DOCX، تلخيص ذكي، Risk Score، محادثة مع العقد، وتقرير PDF احترافي.

نوع المشروع

AI Contract Analyzer

مدة التنفيذ

14 يوم

المخرجات

Web App + REST API + PDF
Report

الإطار

Laravel

فكرة المشروع

AI Contract & Document Analyzer هو نظام ذكي مبني باستخدام Laravel وتقنيات الذكاء الاصطناعي، يهدف إلى تحليل العقود والملفات تلقائيًا.

بدلاً من أن يقرأ المستخدم عقدًا مكونًا من عشرات الصفحات، يمكنه رفع ملف بصيغة PDF أو DOCX، ليقوم النظام بتحليله باستخدام الذكاء الاصطناعي واستخراج أهم المعلومات.

- تلخيص العقد.
- استخراج أهم البنود.
- تقييم درجة الخطورة Risk Score.
- شرح محتوى العقد بلغة بسيطة.
- الإجابة على أسئلة المستخدم حول العقد.
- إنشاء تقرير PDF احترافي يحتوي على نتائج التحليل.

الأدوات والتقنيات المستخدمة

Database MySQL / MariaDB	Backend Laravel, PHP, Composer
Frontend Blade, Tailwind CSS	Authentication Laravel Breeze
Background Processing Laravel Queues & Jobs	File Management Laravel Storage
File Processing PDF Text Extraction, DOCX Text Extraction	AI Integration OpenAI API أو أي مزود AI متوافق
API Testing Postman	PDF Generation Browsershot أو Dompdf
Development Environment XAMPP أو VS Code, Laragon	Version Control Git & GitHub

المميزات الرئيسية

Dashboard

- عرض عدد الملفات.
- عدد التحليلات.
- آخر الملفات المرفوعة.
- عدد الملفات عالية الخطورة.

Authentication

- Login
- Register
- Logout

AI Analysis

- AI Summary
- AI Risk Score
- استخراج أهم البنود.
- تحليل العقد.

Document Management

- رفع ملفات PDF و .DOCX.
- حفظ الملفات داخل Storage.
- إدارة الملفات الخاصة بكل مستخدم.

Search

- البحث داخل التحليلات السابقة.

Chat with Contract

- سؤال الذكاء الاصطناعي عن العقد.
- حفظ المحادثات.

Reports

- تصدير تقرير PDF احترافي.

Notifications

- إشعار عند انتهاء تحليل الملف.

Admin Dashboard

- إدارة المستخدمين.
- إدارة الملفات.
- عرض جميع التحليلات.

REST API

- API للمستخدمين والملفات والتحليلات.

قاعدة البيانات والعلاقات

الجدول

- document_reports
- notifications
- ai_chats
- analyses
- documents
- users

العلاقات

Relationships	Model
hasMany Documents, hasMany Notifications	User
belongsTo User, hasOne Analysis, hasMany AiChats, hasMany DocumentReports	Document

Foundation

اليوم الأول - إعداد المشروع

- إنشاء مشروع Laravel جديد.
- ربط المشروع بقاعدة بيانات MySQL.
- إعداد ملف .env.
- تشغيل المشروع محليًا.
- إنشاء مستودع GitHub.
- كتابة README أولي.
- تصميم مبدئي للصفحات.

النتيجة: مشروع Laravel يعمل، قاعدة البيانات متصلة، GitHub جاهز، وصفحة Home أولية.

Login/Register

اليوم الثاني - Authentication

- تثبيت Laravel Breeze.
- إنشاء Login و Register و Logout.
- تحويل المستخدم إلى Dashboard بعد تسجيل الدخول.
- إضافة Role: User / Admin.

النتيجة: نظام تسجيل دخول متكامل وصفحات Login و Register و Dashboard.

Migrations

اليوم الثالث - Database Design

- إنشاء جميع Migrations.
- إنشاء Models.
- إنشاء العلاقات بين الجداول.

النتيجة: قاعدة بيانات منظمة وجاهزة.

Blade UI

اليوم الرابع - Dashboard & Layout

- إنشاء Layout موحد.
- إضافة Sidebar و Navbar.
- Dashboard يحتوي على عدد الملفات، عدد التحليلات، آخر ملف، وعدد الملفات عالية الخطورة.

النتيجة: واجهة احترافية ومنظمة.

Upload

اليوم الخامس - رفع الملفات

- إنشاء صفحة رفع الملفات.
- دعم PDF و DOCX.
- التحقق من نوع وحجم الملف.
- حفظ الملفات داخل Storage.
- حفظ بيانات الملف داخل جدول Documents: user_id, original_name, file_path, file_type, status
- الحالات: Pending, Processing, Completed, Failed.

النتيجة: رفع الملفات يعمل بالكامل.

Background Jobs

Queue Jobs - اليوم السادس

- إنشاء AnalyzeDocumentJob.
- إرسال Job بعد رفع الملف.
- تحديث حالة الملف تلقائيًا.
- تشغيل Queue Worker.
- استخدام Mock Analysis مؤقتًا إذا لم يكن AI جاهزًا.

النتيجة: المشروع يدعم Laravel Queue.

Text Extraction

اليوم السابع - استخراج النص

- استخراج النص من PDF.
- استخراج النص من DOCX.
- حفظ النص.
- معالجة الأخطاء.

النتيجة: تحويل الملفات إلى نص قابل للتحليل.

AI Analysis

اليوم الثامن - AI Summary

- إرسال نص العقد إلى AI.
- استخراج نوع العقد، الأطراف، مدة العقد، تاريخ البداية، تاريخ النهاية.
- استخراج أهم الالتزامات، الشرط الجزائي، والتجديد التلقائي.
- حفظ النتائج في جدول Analyses.

النتيجة: ظهور AI Summary Card.

Risk

اليوم التاسع - AI Risk Score

- حساب درجة الخطورة من 0 إلى 100.
- تصنيفها إلى Low Risk و Medium Risk و High Risk.
- حفظ سبب الخطورة.

Risk Score: 78% | Status: High Risk | Reason: Contains a strict penalty clause and automatic renewal.

النتيجة: عرض Risk Score بشكل احترافي.

Results

اليوم العاشر - صفحة النتائج + History + Search

- صفحة تفاصيل التحليل تعرض اسم الملف، تاريخ التحليل، Risk Score، AI Summary، وأهم البنود.
- إضافة History.
- إضافة Search حسب اسم الملف، نوع العقد، درجة الخطورة، والتاريخ.

النتيجة: عرض جميع التحليلات السابقة مع إمكانية البحث.

AI Chat

اليوم الحادي عشر - Chat with Contract

- إنشاء صفحة المحادثة.
- إرسال السؤال إلى AI مع نص العقد.
- حفظ الأسئلة والإجابات.
- عرض المحادثة.
- أمثلة: هل يوجد شرط جزائي؟ متى ينتهي العقد؟ هل يوجد تجديد تلقائي؟ ما أكثر بند خطير؟

النتيجة: إمكانية التحدث مع العقد باستخدام AI.

Reports

اليوم الثاني عشر - Export PDF Report + Notifications

- إضافة زر Download PDF Report.
- التقرير يحتوي على اسم الملف، تاريخ التحليل، Risk Score، AI Summary، أسباب الخطورة، أهم البنود، وأسئلة وإجابات المستخدم.
- حفظ التقرير داخل Document Reports.
- إرسال Notification عند انتهاء التحليل.

النتيجة: تقرير PDF احترافي مع إشعارات.

API/Admin

اليوم الثالث عشر - REST API + Admin Dashboard

- إنشاء REST API باستخدام Laravel Sanctum.
- Endpoints: Login, Register, Documents, Upload Document, Show Analysis, Chat with Document, Notifications.
- Admin Dashboard لعرض المستخدمين، الملفات، التحليلات، وفلتر الملفات عالية الخطورة.
- Authorization: المستخدم يرى ملفاته فقط، والأدمن يرى جميع الملفات.

النتيجة: REST API و Admin Dashboard جاهزان.

Delivery

اليوم الرابع عشر - Testing & Final Preparation

- اختبار دورة العمل كاملة: Register, Login, Upload Document, Queue, AI Analysis, Chat, PDF Report, History.
- إضافة Seed Data.
- تحسين التصميم.
- كتابة README النهائي.
- تجهيز العرض Presentation وتجهيز Demo.

النتيجة: مشروع جاهز للعرض والتقييم.

سيناريو العرض Demo

1. تسجيل الدخول.
2. رفع عقد بصيغة PDF أو DOCX.
3. ظهور حالة Processing.
4. انتهاء التحليل.
5. عرض AI Summary.
6. عرض AI Risk Score.
7. سؤال AI عن العقد.
8. تحميل PDF Report.
9. الدخول إلى Admin Dashboard واستعراض جميع الملفات والتحليلات.

Concept Alignment Addendum

This addendum explains how the project keeps the original Laravel + AI learning concepts while applying them to a different domain: contract and document analysis.

Mapping Table

Original Concept	AI Contract & Document Analyzer Implementation
Blog Post	Document / Contract
Category / Tag	Document Type / Risk Level / Status
Post CRUD	Upload, show, delete, and manage documents and analyses
AI Draft Generator	AI Summary + Risk Analysis
Blog Search	Search in analysis history
Notifications	Analysis completed notification
API Resources	Documents API + Analyses API
Testing	Upload tests + Auth tests + Analysis tests

Discussion Explanation

The project does not leave the learning concept. It changes the application domain only. Instead of managing blog posts, the system manages documents and contracts. The same Laravel concepts still appear through authentication, dashboard pages, resource management, validation, storage, relationships, AI integration, search, notifications, API endpoints, admin views, and testing.

Short Defense Statement

I did not leave the course concepts. I changed the domain only. The same Laravel concepts are implemented through documents, analyses, AI summaries, risk scoring, API endpoints, notifications, and tests.

Concept Coverage Checklist

Auth	Login, register, logout, and reset flow
Dashboard	Document counts, analyses, risk status, and generated reports
Database	Users, documents, analyses, chats, reports, and notifications
AI	Summary, clause extraction, risk score, and chat answers
Search	History and analysis lookup

API	Document and analysis resources
Jobs	Background analysis workflow
Testing	Auth, upload, analysis, report, and API tests