AI-Powered Document Processor

End-to-End Implementation Document (React Web Application)

1. Overview

The AI-Powered Document Processor is a React web application that allows users to upload, classify, extract, enrich, and manage documents using AI models (OCR, NLP, classification, and entity extraction). The system supports dynamic handling, meaning new document types can be configured without code changes. Extraction templates, fields, and validation rules are data-driven.

2. Goals and Objectives

- Centralized document processing.
- Dynamic configuration.
- Human-in-the-loop review.
- Traceability and auditability.
- Scalability and extensibility.

3. Core Features

- Document upload & ingestion.
- AI processing pipeline (classification, OCR, extraction).
- Dynamic document handling.
- Review & correction UI.
- Document search & list.
- Document detail view.
- User roles & permissions.

4. High-Level Architecture

Frontend: React, React Query, Material UI, Vite.

Backend: AI services, configuration service, workflow engine, storage.

Data Flow: Upload -> AI processing -> Review -> Approve.

5. Dynamic Handling Design

Document type configuration model, dynamic form rendering, validation rules, extensibility.

6. User Flows

- Upload & process.

- Review & approve.

- Search & retrieval.

7. React Application Structure

Folder structure including api/, components/, pages/, hooks/, context/, routing/, types/.

8. API Contracts

Authentication, document upload, processing results, configuration, document list.

9. Key React Components

DynamicForm, FieldRenderer, DocumentViewer.

10. Sample Code Snippets

Dynamic form rendering and document detail page flow.

11. Non-Functional Requirements

Performance, security, scalability, audit & compliance.

12. Out of Scope

Workflow editors, admin UI, real-time collaboration, offline mode.