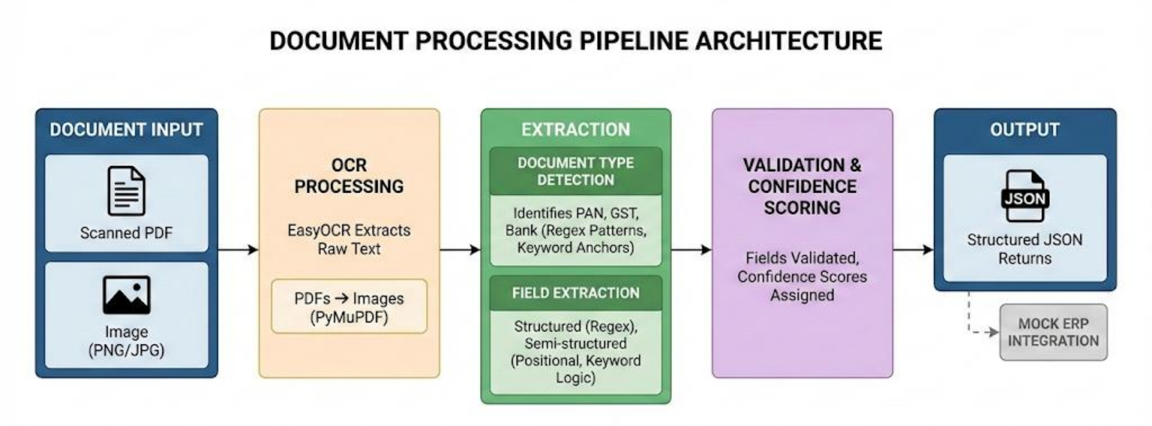
Document OCR Extractor (PAN/GST/Bank)

Problem Statement

Vendor onboarding in ERP systems requires extracting KYC information from documents such as PAN cards, GST certificates, and bank proofs. Manual data entry is slow, error-prone, and difficult to scale. The objective of this project is to build an automated Python-based document extraction system that converts unstructured scanned documents into structured, validated JSON suitable for ERP vendor master creation.

Architecture / Flow


DOCUMENT PROCESSING PIPELINE ARCHITECTURE
DOCUMENT INPUT
Scanned PDF
Image (PNG/JPG)
OCR PROCESSING
EasyOCR Extracts Raw Text
PDFs → Images (PyMuPDF)
EXTRACTION
DOCUMENT TYPE DETECTION
Identifies PAN, GST, Bank (Regex Patterns, Keyword Anchors)
FIELD EXTRACTION
Structured (Regex), Semi-structured (Positional, Keyword Logic)
VALIDATION & CONFIDENCE SCORING
Fields Validated, Confidence Scores Assigned
OUTPUT
JSON
Structured JSON Returns
MOCK ERP INTEGRATION

Tools & Libraries

- Python 3.10 and above
- EasyOCR – OCR engine
- PyMuPDF – PDF processing
- OpenCV – Image preprocessing
- Regex – Pattern-based extraction
- spaCy – Named Entity Recognition
- FastAPI – REST API framework
- Pydantic – Data validation
- pytest – Unit testing

Key Logic Explanation

- Input documents (PDF or images) are pre-processed and processed using EasyOCR to extract text.
- Document type (PAN, GST, or Bank) is automatically detected using keywords and identifier patterns such as PAN, GSTIN, and IFSC formats.
- Structured fields (PAN, GSTIN, IFSC, account number) are extracted using strict regex validation, while names and addresses use anchor-based positional logic.
- OCR noise is handled using character-repair rules and fuzzy keyword matching to improve extraction robustness.
- For multi-page documents, fields are consolidated using highest-confidence selection, and each field is assigned a confidence score for extraction.