Intern Project Requirement: OCR Text Extraction and Visualization Tool

Objective

Build a Python-based OCR web application with the following functionalities:
1. Upload scanned documents (PDF/TIFF)
2. Extract text using Tesseract or PaddleOCR (Option to select OCR engine on UI)
3. Store extracted text and coordinates in PostgreSQL
4. Display results in a structured table on the front-end with bounding box coordinates

Tech Stack

- Backend: Python (FastAPI)
- OCR Engine: Tesseract or PaddleOCR (choose one or support both)
- Database: PostgreSQL
- Frontend: HTML/CSS/JavaScript (basic UI using Bootstrap or React optional)
- File Handling: PDF or TIFF image input

Core Features

1. File Upload

- Accept PDF or TIFF images from user via web UI. Validate file format and size (<10MB).

2. OCR Extraction

- Convert input to image (for PDFs, use pdf2image). Apply OCR (Tesseract or PaddleOCR) to extract:
  - Text
  - Bounding box coordinates (x, y, width, height)

  Output format:
  ```
  [
   {"text": "Invoice No", "x": 150, "y": 420, "width": 100, "height": 20},
   ...
  ]
  ```

3. Database Storage

- Store each OCR result in a PostgreSQL table with:
  - File ID
  - Line Text
  - Bounding Box Coordinates

- Page Number (for multi-page documents)
- Timestamp

4. UI Display

- After OCR, show a table in UI with:
  - Line Number
  - Text
  - X, Y, Width, Height

Database Schema Suggestion

```
CREATE TABLE ocr_results (
   id SERIAL PRIMARY KEY,
   file_name TEXT,
   page_number INT,
   line_number INT,
   line_text TEXT,
   x INT,
   y INT,
   width INT,
   height INT,
   processed_at CURRENT_TIMESTAMP
);
```

Test Cases

Test Case Description	Expected Result
Upload valid PDF	File processed, OCR results shown in table
Upload invalid file (e.g. .docx)	Show error: 'Unsupported file format'
Extract text with Tesseract from 1-page PDF	Text + coordinates stored and displayed
Extract text from multi-page PDF	All pages processed, paginated in UI (optional)
Database stores each text line with coordinates	Check PostgreSQL entries post processing
OCR lines are sorted as per image order	Table displays text lines in logical reading order
Upload large scanned image (up to 10MB)	Processes successfully and displays data
Missing dependencies (Tesseract or PaddleOCR not installed)	App shows setup error or guidance

Deliverables

- Source code on GITLab, GitHub or ZIP
- ReadMe with instructions
- Sample output screenshot

- Test data set (sample PDFs/TIFFs)