Project Brief: AI Document Extractor & Converter

Project Title: AI Document Extractor & Converter

Project Overview:

The AI Document Extractor & Converter application automates the process of extracting, processing, and converting data from various document formats, such as Word, PDF, and plain text files. This tool provides users with a web-based interface for uploading files, extracting meaningful content, and converting documents into desired formats. Designed to be modular and general-purpose, this application can accommodate diverse use cases by processing structured or semi-structured data and applying flexible configurations for mapping and output.

Objectives:

1. Develop a dynamic application to extract and process data from multiple document formats.
2. Provide a user-friendly interface for file uploads and real-time feedback on processing.
3. Enable seamless conversion of documents into various output formats (e.g., CSV, PDF, DOCX).
4. Allow users to customise extraction and processing through predefined or user-defined configurations.
5. Ensure modularity, making the components reusable and adaptable for other projects.

Key Features:

1. **Document Upload:** Allow users to upload documents in formats such as .docx, .pdf, or .txt via a web interface.
2. **Data Extraction:** Automatically extract structured or semi-structured data using intelligent parsing algorithms, including regex and NLP-based methods.
3. **Data Mapping:** Map extracted fields to custom-defined schemas for further processing or analysis.
4. **File Conversion:** Convert input documents to user-specified formats, such as .csv, .pdf, or .docx.
5. **Error Handling and Logging:** Provide detailed logs for errors and successful operations for traceability.
6. **Web Interface:** Include an intuitive UI for file uploads, progress tracking, and output management.
7. **Configurable Workflow:** Enable users to upload or select predefined configurations for field extraction and mapping.
8. **Multi-format Compatibility:** Ensure seamless handling of diverse document structures and file formats.

Technical Specifications:

1. **Programming Language:** Python
2. **Frameworks and Libraries:**
    - Flask/Django: Web application framework.
    - PyPDF2/Fitz: For PDF handling and text extraction.
    - python-docx: For processing Word documents.
    - Regex and NLP libraries (e.g., SpaCy or NLTK): For data extraction.
3. **Database:** SQLlite for storing user-defined configurations and logs.
4. **Deployment:** Docker containerization and cloud service deployment (e.g., AWS or Azure).
5. **Version Control:** Git for source code management.

Expected Outcomes:

1. A general-purpose AI Document Extractor & Converter application, fully functional for various data extraction and conversion needs.
2. Enhanced efficiency in document processing and reduced manual workload.
3. Modularity for easy adaptation in different projects or workflows.

Risks and Mitigations:

1. **Data Privacy:** Implement encryption and secure access controls to protect sensitive user data.
2. **File Format Challenges:** Test the application thoroughly with different document structures to ensure robust compatibility.
3. **Performance:** Optimise processing algorithms for speed and accuracy, especially with large or complex files.