Project Brief: Docs Directory AI Summariser

Project Title: Docs Directory AI Summariser

Project Overview:

The **Docs Directory AI Summariser** is a desktop or web-based utility designed to provide **intelligent, high-level insights into document folders** without requiring users to manually open or inspect individual files. By inputting a folder path, users receive a **comprehensive summary of the directory's contents**, including file counts, word and character statistics, and breakdowns by file type.

The system also includes a **template matching module**, allowing users to upload reference templates (e.g. Word documents, PDFs, spreadsheets, presentations, markdown files) that the summariser uses to identify and count matching files within the directory. This makes it ideal for **administrators, researchers, and content managers** who need to audit or assess document repositories quickly and efficiently.

Objectives:

1. Build a tool that accepts a folder path and generates summary statistics for all files inside.
2. Extract and display file-level and aggregate metrics such as word count, character count, and file type distribution.
3. Implement a template matching system that allows users to upload sample files and count how many similar files exist in the directory.
4. Ensure support for common textual formats including DOCX, PDF, TXT, MD, XLSX, PPTX, and others.
5. Display statistics only for templates with at least one match, keeping the output clean and relevant.
6. Provide a clear, structured summary dashboard that highlights key metrics and insights.
7. Allow users to filter or group statistics by file type, template match, or other metadata.

Key Features:

1. **Folder Path Input:** Users can specify a directory to analyse.
2. **File Count & Type Breakdown:** Total number of files, grouped by extension (e.g. .docx, .pdf, .txt).
3. **Textual Analysis:** Word and character counts for each file (unless non-textual), plus total and average across the folder.
4. **Template Matching System:** Upload templates to detect and count similar files in the directory.
5. **Selective Display:** Only show template categories with at least one match.
6. **File Type Summary:** Aggregate statistics grouped by file type.
7. **Clean Summary Output:** Structured dashboard or report view with sortable metrics.
8. **Scalable Design:** Handles large directories with hundreds of files efficiently.

Technical Specifications:

1. **Programming Language:** Python
2. **Frameworks and Libraries:**
   - Flask/Django: Web application framework.
   - os, pathlib: For directory traversal
   - python-docx, PyPDF2, openpyxl, python-pptx, etc: For file parsing
   - difflib or fuzzywuzzy: For template similarity matching
   - Pandas: For tabular data aggregation and filtering
3. **Database:** SQLlite for storing template metadata and cached summaries.
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 fully functional directory summarisation tool that provides instant insights into file collections.
2. Time-saving analytics for users managing large document repositories.
3. A flexible template matching system that adapts to different file types and use cases.
4. A clean, intuitive interface for viewing and exporting summary statistics.

Risks and Mitigations:

1. **Performance Bottlenecks:** Optimise file parsing and use background threads for large directories.
2. **Template Matching Accuracy:** Use configurable thresholds and preview matches before counting.
3. **File Format Limitations:** Ensure graceful handling of unsupported or corrupted files.
4. **Security & Privacy:** Avoid storing file contents; only extract metadata and summary statistics.