**AI Pipeline for BLOOD TEST ANALYZER**

**Summary and Problem Statement**

The Medical Report Analysis app is a comprehensive tool designed to analyze blood test reports from PDF files. Built using the Streamlit framework, this application processes the data extracted from the PDF and delivers a detailed analysis, which includes key medical findings, potential health concerns, recommended follow-up tests, actionable lifestyle advice, and references to trusted medical resources. The goal of this application is to provide users with insights into their health based on their blood test reports and to offer guidance on necessary steps for improving or maintaining their health.

**Workflow:**

1. **User uploads a report in Streamlit (**[main.py](vscode-file://vscode-app/c:/Users/Anvesh%20Pudi/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html)**).**
2. **Extract text from the report.**
3. **Pass the text to Crew (from**[medical\_crew.py](vscode-file://vscode-app/c:/Users/Anvesh%20Pudi/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html)**),** 
   * Which Runs the analysis task ([tasks.py](vscode-file://vscode-app/c:/Users/Anvesh%20Pudi/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html)).
   * Uses tools ([tools.py](vscode-file://vscode-app/c:/Users/Anvesh%20Pudi/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html)) as needed.
   * Finds articles and provides recommendations.
4. **Display the structured results in Streamlit.**

**Functionalities of individual methods used :**

1. [tasks.py](vscode-file://vscode-app/c:/Users/Anvesh%20Pudi/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html)

* **Purpose:** Defines structured, reusable "tasks" for analyzing blood tests, finding articles, and providing recommendations.
* **Role:** Instead of sending a generic prompt to Gemini, you could use these tasks to:
  + Break down the analysis into steps (e.g., first analyze the report, then find articles, then give recommendations).
  + Make your workflow more modular and maintainable.

**2.**[tools.py](vscode-file://vscode-app/c:/Users/Anvesh%20Pudi/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html)

* **Purpose:** Provides utility functions or integrations (e.g., web search, data extraction, calculations) that can be used by agents or tasks.
* **Role:** Enhance the capabilities of your agents/tasks, such as:
  + Searching for articles online.
  + Extracting structured data from reports.
  + Any other helper logic needed for analysis.

**3.**[medical\_crew.py](vscode-file://vscode-app/c:/Users/Anvesh%20Pudi/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html)

* **Purpose:** Orchestrates the workflow by combining agents and tasks into a "crew" (using the [CrewAI](vscode-file://vscode-app/c:/Users/Anvesh%20Pudi/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") framework).
* **Role:** Manages the sequence and dependencies between tasks and agents, ensuring the right agent does the right task at the right time.