📋 Web Research Agent — Detailed Plan

🛠️ Step-by-Step Process:

1. User Input
   ➔ User sends a query (e.g., "Impact of AI in Healthcare 2025") via API /agent/ endpoint.

2. Search Phase
   ➔ Agent uses Google Search API or Playwright scraping to search for top 10–15 relevant links.

3. News Fetching
   ➔ Simultaneously, it fetches the latest news articles about the same topic via News API.

4. Scraping Phase
   ➔ Agent visits each link found and scrapes the content (using BeautifulSoup + Playwright).

5. Text Cleaning Phase
   ➔ Agent removes noise, extracts clean main text from each webpage.

6. Content Analysis Phase
   ➔ Agent sends the scraped text to the Content Analyzer, where:

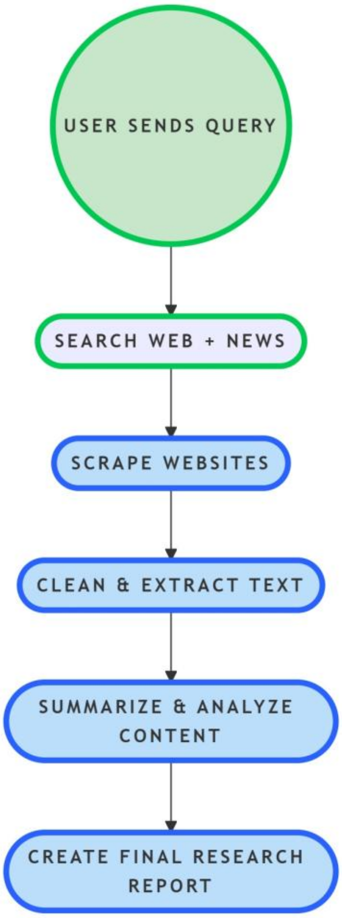
   - It summarizes the text (currently using HuggingFace BART model).
   - It extracts named entities (like organizations, people, locations).
   - It calculates relevance with the original query.

7. Synthesis Phase
   ➔ Agent compiles all summaries, news articles, and important entities into a final research report.

8. Output Phase
   ➔ Returns a clean JSON response with:

   - Final Summary
   - Key Entities
   - Top Web Results
   - News Articles

📈 Simple Flowchart:



🔥 Step-by-Step Decision Making Process

Step	Decision
1	Receive user query via API /agent/.
2	Use query directly as search input.
3	Perform Google Search and News API search.
4	Collect 10–15 URLs and 5–10 news articles.
5	Try to scrape each URL one by one.
6	If a website fails scraping ➔ skip and move to next.
7	Clean and prepare the text.
8	Analyze text: summarization + entity extraction.
9	Aggregate all individual summaries into a full report.
10	Respond with the final compiled JSON result.

🛡️ Error Handling Plan

Problem	How Agent Handles It
Website not reachable / blocked	Log error, skip that website, continue with others.
Scraper crashes on a page	Catch exception, skip that page, continue.
No results from search engine	Return an error message: "No web results found."
API token expired (HuggingFace, NewsAPI)	Log error clearly, return "Service temporarily unavailable."
Conflicting information	Summarize facts neutrally without taking sides.
Huggingface summarization API fails	Catch error, skip summarization, just return extracted text instead.
Redis / Celery not connected (local)	Log and retry multiple times (Celery already retries connection).
JSON error (e.g., invalid API response)	Catch and log, move to next task.

🏗 Architecture Layers (Clean Design)

- Router ➔ app/main.py (FastAPI endpoints)
- Agents ➔ app/agents/ (Manages query flow)
- Tools ➔ app/tools/ (Searcher, Scraper, Analyzer)
- Utils ➔ app/utils/ (Logger, Common helpers)
- Config ➔ app/config.py (Environment configs)
- Worker ➔ worker/celery.py (Async background jobs)