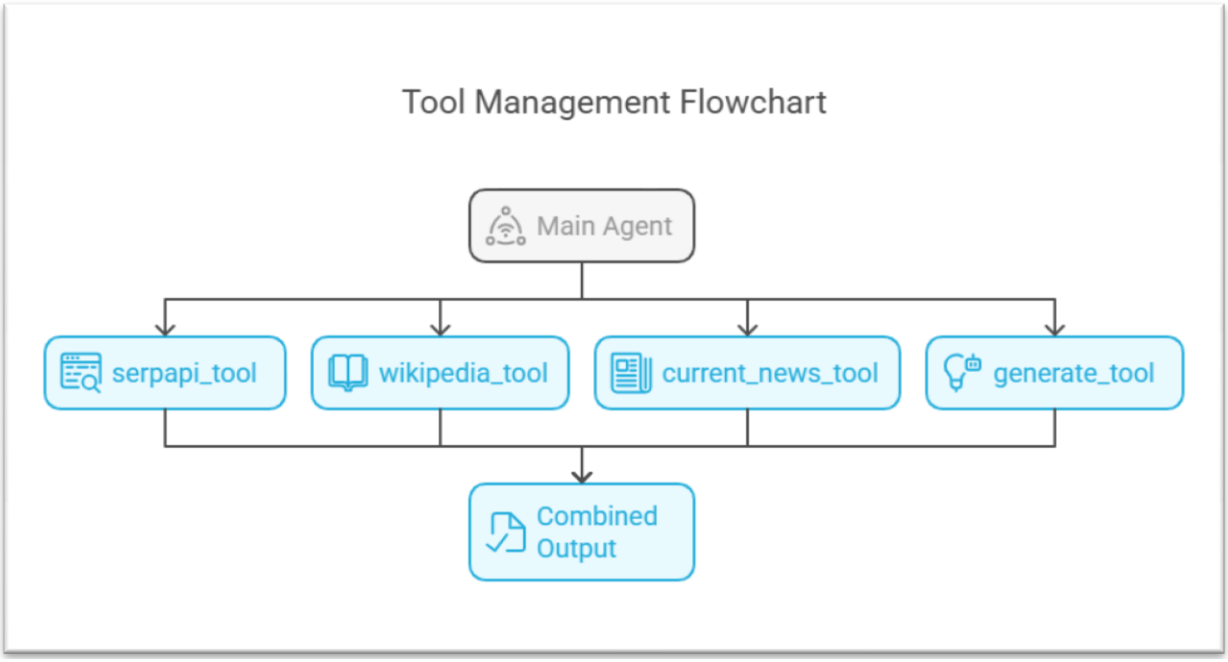
Documentation: AI Research Agent

1. How Agent is Structured.

The AI Research Agent is composed of several components that work together to gather, analyze, and generate research reports. The system is modular, with each tool performing a specific task. The main agent orchestrates these tools to process the user's query and compile results into a comprehensive report. Here's a breakdown of the structure:



- User Interface: The system uses Streamlit to provide a user-friendly web interface for interaction.
- Core Agent: The agent is responsible for managing the flow of information. It receives the user's query and determines which tool(s) to use based on the input.
- External Tools: The agent connects to external tools (e.g., SerpAPI, Wikipedia, Groq, Gemini) to fetch relevant data.
- Combiner: Once the external tools return their results, they are passed to a combining function, which merges the results into a coherent research report.
- Cache System: Frequently asked questions and results are cached in session state to speed up future queries and avoid redundant processing.
- Error Handling: If the agent encounters issues or receives unrecognized queries, it responds with a helpful error message and suggests possible fixes.

2. How I Designed the Prompts/Instructions for the AI

The prompts and instructions for the AI models (Groq, Gemini) are carefully designed to extract the most relevant and coherent responses. The prompts vary based on the type of query:

- General Research Queries: The prompts are open-ended and designed to encourage the AI to synthesize information across multiple domains. For example:
  - "Provide an overview of the current trends in AI research."
- Current News: When asking for recent news, the prompts are specific to ensure the AI can filter the most up-to-date content. Example:
  - "What are the latest developments in AI within the last week?"
- Wikipedia Queries: The prompts are designed to extract information from Wikipedia for well-known topics. Example:
  - "Tell me about quantum computing."
- Error Messages: If the AI cannot generate a valid response, the prompts include instructions to guide the user to refine their query, such as:
  - "Sorry, I couldn't find results for that query. Please try using more specific keywords."

The system also uses Groq's LLaMA model for more advanced reasoning, summarization, and explanation tasks. The model is prompted with specific tasks based on the user's query, such as summarizing long-form articles or explaining complex topics.

3. How my Agent Connects to and Uses External Tools

The AI Research Agent integrates several external tools to expand its knowledge base and capabilities:

- SerpAPI: Used for conducting web searches. The agent queries SerpAPI when a user asks for general web search results. The tool returns URLs that are relevant to the query.
- Wikipedia API: When the user asks for well-known topics (like "quantum computing"), the agent connects to the Wikipedia API to extract a summary of the topic.
- Current News API: For recent or trending topics, the agent fetches news articles or data using a news API to provide the latest insights.
- Gemini API: This is used for more in-depth, reasoning-driven queries. The agent can interact with Google's Gemini API to process natural language and provide advanced AI-driven answers.
- Groq API: Used to generate AI responses for complex tasks such as summarization, explanation, and reasoning. It leverages Groq's LLaMA model to generate high-quality responses.

Each tool is configured via API keys stored in a .env file. The agent identifies which tool to use based on the query type and processes the request accordingly.

4. How my Agent Handles Errors and Unexpected Situations

Error handling is crucial to ensure the agent remains functional even in unpredictable situations. Here’s how errors are managed:

- Invalid Input Handling: If the user submits an invalid query or if the agent cannot identify the appropriate tool for the task, the system provides an error message, such as:
  - "I couldn't understand your query. Please rephrase it or be more specific."
- API Failures: If any external API (like SerpAPI, Gemini, etc.) fails or is unreachable, the agent will log the error and notify the user that there was a technical issue. For example:
  - "There was an issue connecting to the external data source. Please try again later."
- No Results: In cases where external tools return no results, the agent checks for a response status and provides the user with a helpful message:
  - "I couldn't find any relevant data for that query. Would you like to try something else?"
- Timeouts: If a tool takes too long to respond (for example, due to server issues or slow API responses), the agent will log the timeout and alert the user:
  - "The request timed out. Please try again after a moment."
- General Exceptions: Any unexpected issues during the processing of the query (such as connection errors or invalid responses) are caught and logged. The system will ensure that the user is informed about the issue.