Long-Context LLMs:
Suppose a user inputs the research topic: "Long-Context Large Language Models (LLMs)."

The application must:

1. **Search Agent**: Fetch related papers on long-context capabilities in LLMs from platforms such as Arxiv, focusing on papers published in the last five years (e.g., 2019-2024). Store these papers in a time-series database.
2. **Database Agent**: Store the collected papers and allow querying of papers based on specific timeframes, such as retrieving all papers related to long-context handling from 2019 onwards.
3. **Frontend Application**: Display all the fetched papers in a user-friendly interface using Streamlit, allowing users to browse the papers easily.
4. **Q&A Agent**: Provide a question-answering feature where users can interact with the research papers. Example questions include:
    - "Summarize the advancements in long-context capabilities for LLMs over the last five years."
    - "What improvements are proposed in Paper X to extend the context length of LLMs?"
    - "How does Paper Y handle the memory limitations of long-context models?"
    - "Which paper presents the most effective approach for long-context handling in LLMs, and why?"
    - (Reference: Paper G - Section 4.5 for details on long-context optimization).
5. **Future Works Agent**: Suggest potential improvements or research opportunities based on the reviewed papers. Generate a review paper summarizing the state-of-the-art techniques and propose new directions for future work, such as enhancing the efficiency of memory usage and reducing computational costs in long-context LLMs.

Chat Interaction Example:
User: "Show me all the papers related to long-context capabilities in LLMs in the last 5 years."
Bot: "Here are the papers on long-context capabilities in LLMs from 2019 to 2024. [List of papers displayed]"
User: "Summarize the advancements made in these papers."
Bot: "The advancements in long-context LLMs over the past five years include the development of sparse attention mechanisms, improvements in memory-augmented architectures, and new techniques to enhance efficiency in long-sequence processing. Notable contributions include Paper A, which introduces a novel memory mechanism, and Paper B, which focuses on sparse attention to extend context length. Reference: Paper A - Section 3.4 | Paper B - Section 2.6"
User: "What improvements are proposed in Paper B to extend the context length of LLMs?"
Bot: "Paper B proposes the use of sparse attention to efficiently allocate computational resources, allowing the model to focus on important segments of input while reducing

overall complexity. This approach significantly extends the effective context length that can be processed. Reference: Paper B - Section 2.6"
User: "What future research can be done based on these papers?"
Bot: "Based on the reviewed papers, future research could focus on optimizing memory efficiency in long-context models, developing hybrid attention mechanisms that balance sparsity and precision, and enhancing the scalability of long-context processing. Reference: Paper A - Section 5 | Paper C - Section 4.7"
User: "Which paper presents the most effective approach for long-context handling in LLMs?"
Bot: "According to the papers reviewed, Paper D presents the most effective approach for long-context handling. This approach incorporates a hybrid of recurrent and transformer-based techniques to efficiently manage extended context lengths, achieving significant improvements in both processing speed and accuracy. Reference: Paper D - Section 4.3 | Paper E - Section 3.8"