

# **Artificial Intelligence (AI) Large Language Models (LLMs) for Systems Engineering (SE) Tasks**

*Presented at: SERC 2023 AI4SE & SE4AI Workshop*

John Crissman  
Research Analyst



# Introduction

---

- LLMs have numerous applications, including:



Content Generation



Text Summarization



Chatbots and Virtual Assistants



Information Retrieval

- AI products and LLMs can accelerate several critical tasks in SE, but contain risks of providing factually incorrect information (i.e., hallucinations).

# LLMs can help in numerous SE tasks

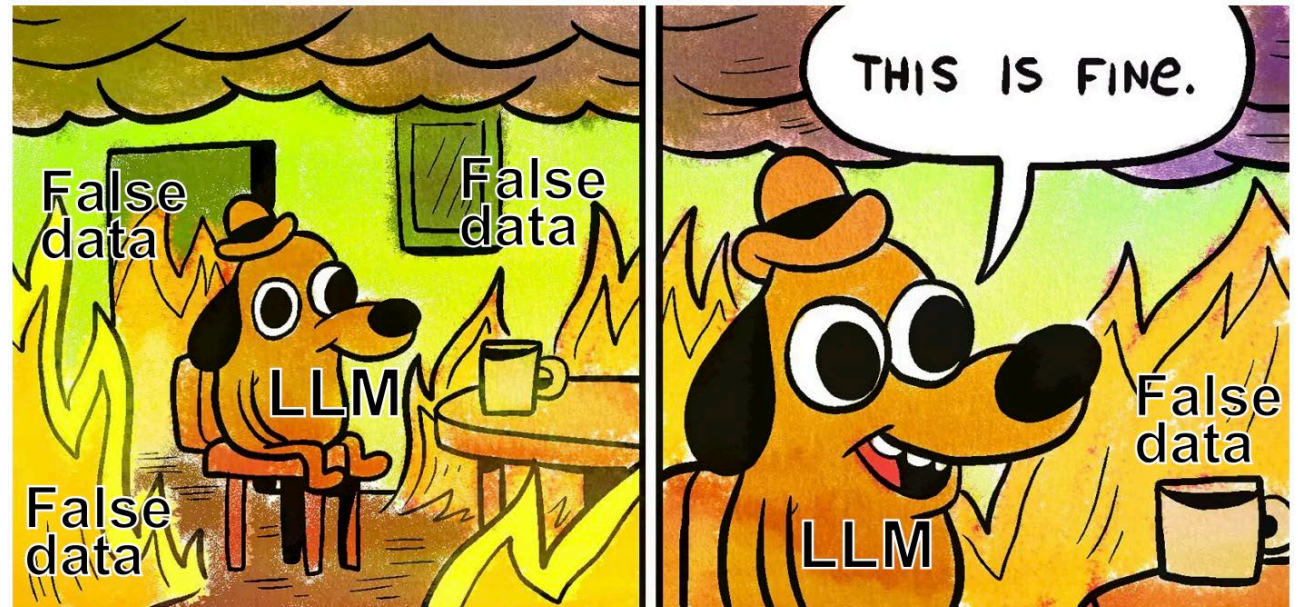
---

- LLMs can help in numerous SE tasks, including:
  - Generating requirements
  - Producing documentation
  - Automating the development of test cases
  - Assisting in identifying and writing system risks
  - Developing Agile user stories



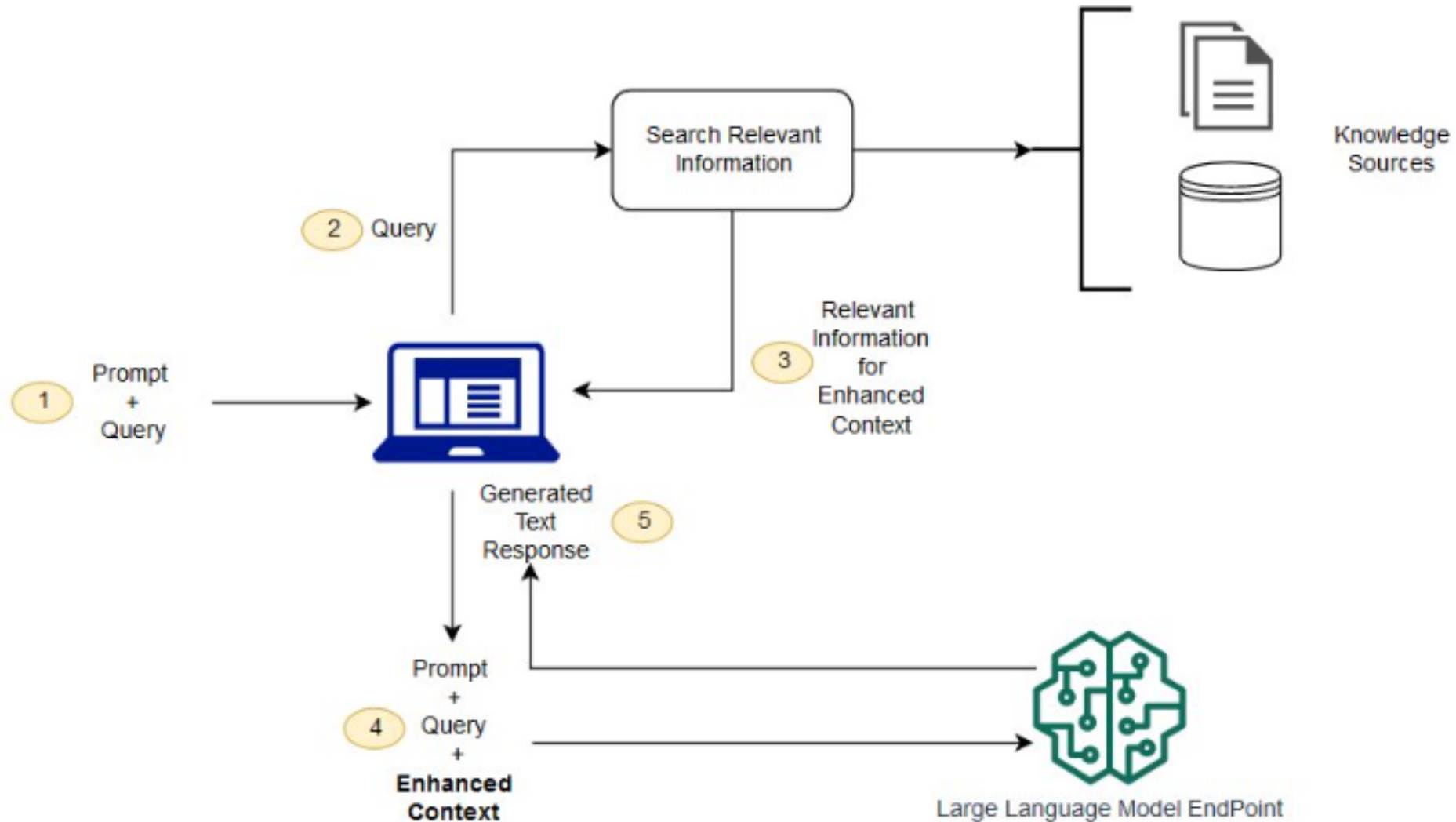
# Hallucinations and Mitigation Strategies

- LLM **hallucinations** are when LLMs generate text that is **factually incorrect**.
- Mitigation Techniques:
  - Fine-tuning LLMs
  - Retrieval-Augmented Generation (RAG)
  - Limiting LLM response
  - Prompt engineering
  - LLM parameter tuning
  - Manually checking accuracy



Source: [Results from the AI testing hackathon \(alignmentjam.com\)](https://alignmentjam.com/)

# Retrieval Augmented Generation (RAG)

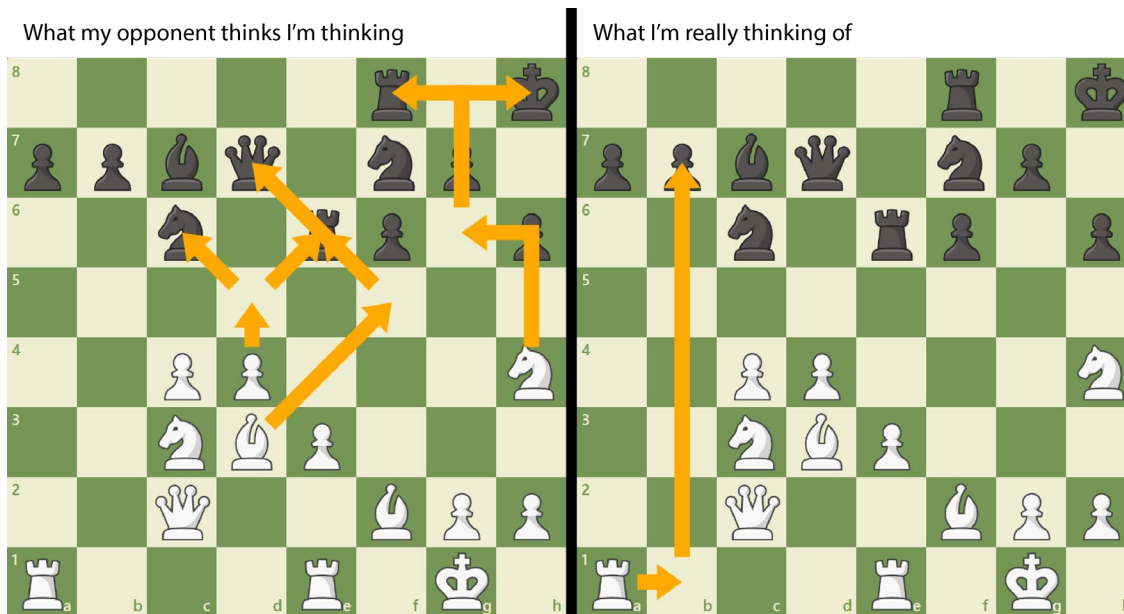


Source: [Retrieval Augmented Generation \(RAG\) - Amazon SageMaker](#)

# Case Studies

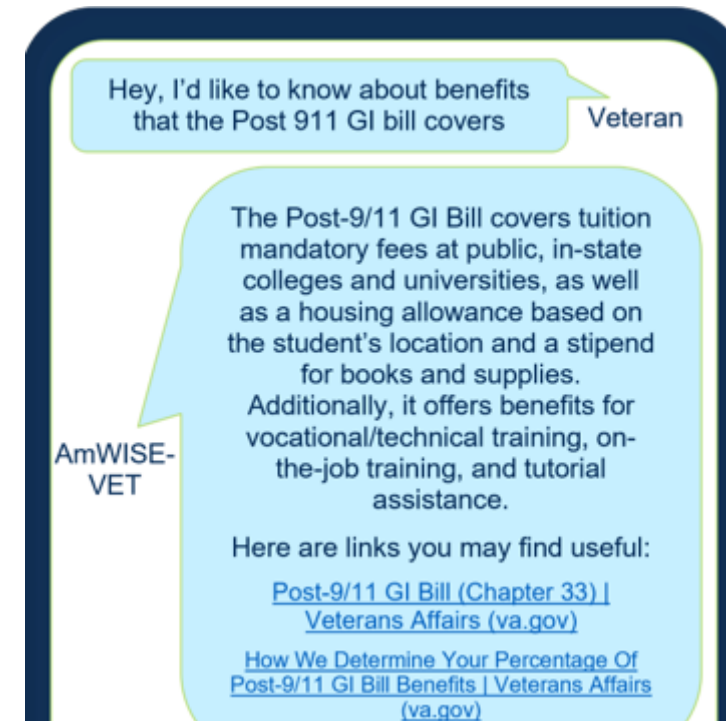
LLMs were used to generate **functional and non-functional requirements** and **test cases** for these two projects

## Chess Engine



Source: <https://www.chess.com>

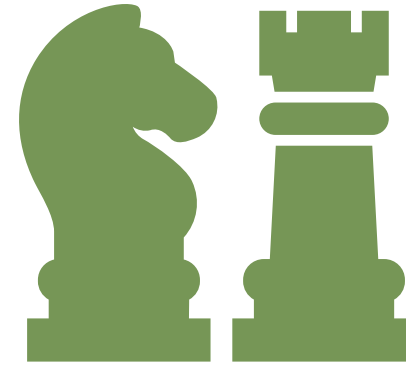
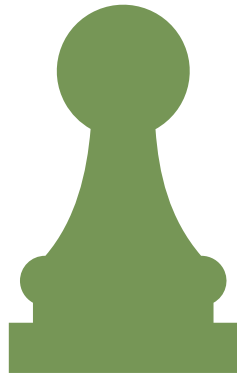
## American's Wellness and Information Support Environment (AmWISE)



# Chess Engine

---

- Many organizations have developed chess engines, and there is **extensive documentation** on chess engines that LLMs have been trained on already.
- Thus, fine-tuning or using RAG was not needed for this use case.



- Future efforts may involve fine-tuning or using RAG to develop requirements and test cases based on a **specific chess engine product** (e.g., Stockfish, Leela Chess Zero, Komodo, Houdini).

# Chess Engine - Results

---

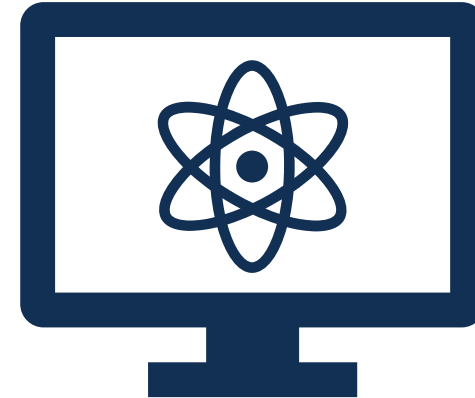
- **Prompt:** What are the functional and non-functional system requirements for an AI-powered chess engine? Generate test cases for each of these requirements.
  - **Position Evaluation:**
    - **Requirement:** The AI engine should evaluate the current position and assign a numerical score.
    - **Test Cases:**
      - » Provide different board positions and verify that appropriate scores are assigned.
      - » Test endgame positions to ensure accurate evaluation.
  - **Move Selection:**
    - **Requirement:** The AI engine must select the best move from a list of possible moves based on position evaluation.
    - **Test Cases:**
      - » Feed the engine with various positions and check if it selects the move with the highest score.
      - » Test complex positions where multiple moves seem good but one is objectively better.



# AmWISE

---

- An **enhanced digital companion**, AmWISE, leverages instruction tuned **LLMs** to answer nuanced questions from the American public and provide the most relevant resource links.



- We used **Retrieval Augmented Generation** (RAG) with background documentation of AmWISE. This **background documentation** is only 7 pages long.

# AmWISE – Results without RAG

---

- **Prompt:** What are the functional and non-functional system requirements for AmWISE?  
Generate test cases for each of these requirements.
  - I'm sorry, but I don't have enough information about AmWISE to provide you with the functional and non-functional system requirements or generate test cases.



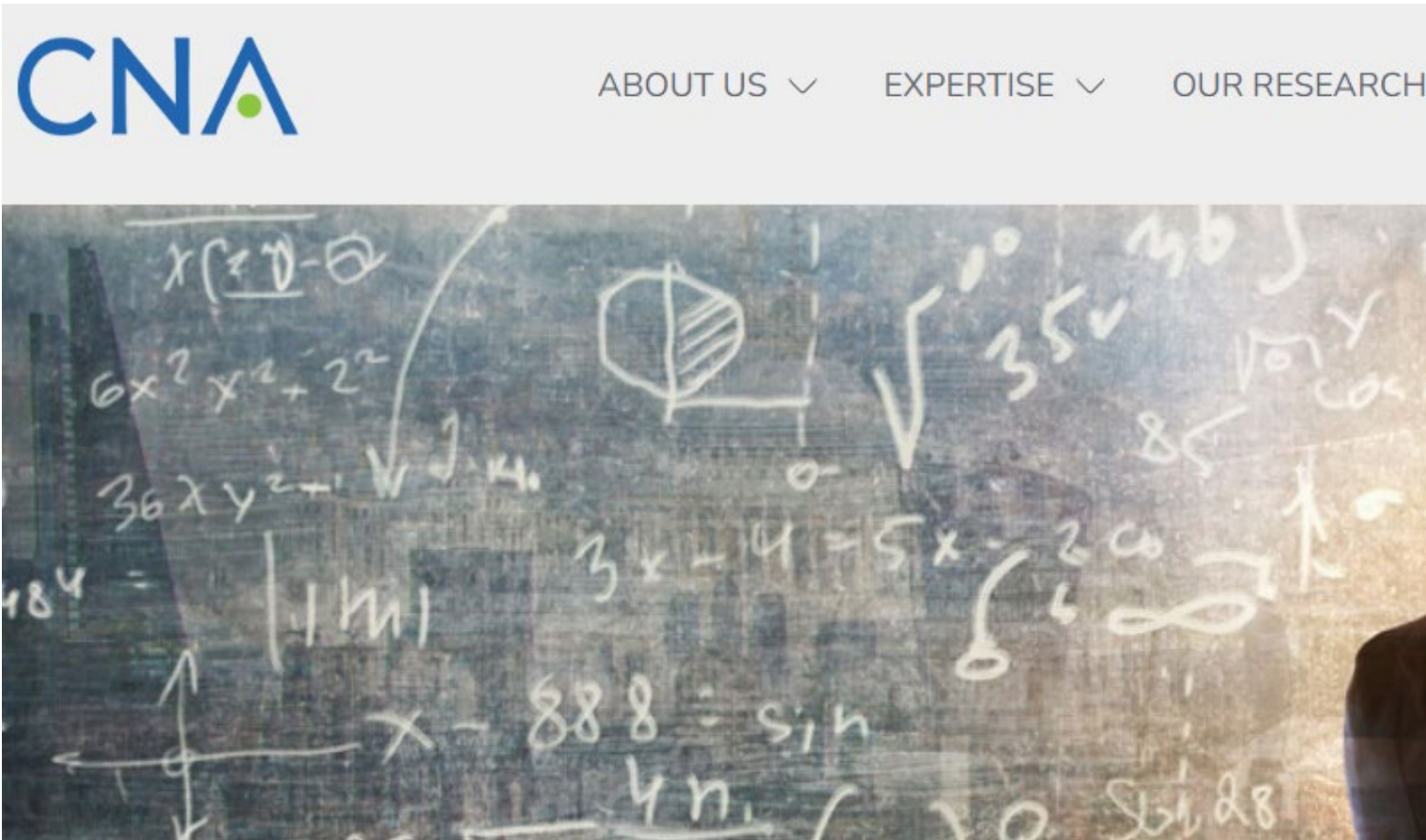
# AmWISE – Results with RAG

---

- **Prompt:** What are the functional and non-functional system requirements for AmWISE?  
Generate test cases for each of these requirements.
  - **Functional Requirements:**
    - AmWISE should accurately and effectively answer nuanced questions from the American public.
    - AmWISE should understand the intent and context of the entire conversation.
    - AmWISE should provide instant delivery of the most relevant answers and links related to user's questions.
  - **Test Cases:**
    - Test the accuracy of AmWISE's responses by providing a set of questions and comparing the generated answers with expected answers.
    - Test the effectiveness of AmWISE by evaluating its ability to understand the intent and context of a conversation.
    - Test the instant delivery of relevant answers by measuring the response time for different types of queries.

# Continue the Conversation!

<https://www.cna.org/centers-and-divisions/ipr/dma>



Center for Data  
Management and  
Analytics

Researcher

John Crissman

[crissmanj@cna.org](mailto:crissmanj@cna.org)

Director

Shaelynn Hales

[haless@cna.org](mailto:haless@cna.org)