CELCOMDIGI KNOWLEDGE BASE ASSISTANT:

By: Boon Yong Yeow

Table of Contents:

Problem Statement

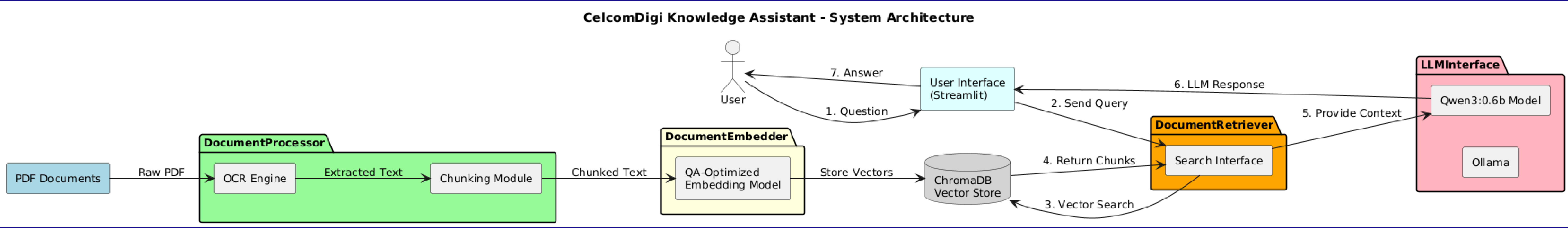
Problem Statement

- Customer service representatives struggle to quickly find specific information buried in multiple CelcomDigi documents
- Longer wait times for customers and inconsistent answers between representatives

Objective

- An AI assistant that instantly finds and presents relevant information from CelcomDigi documents
- A proof-of-concept that demonstrates how AI can make internal knowledge more accessible and useful

System Architecture



This architecture diagram illustrates the complete flow of information through our RAG (Retrieval-Augmented Generation) system:

Document Processing Pipeline:

a. **P**DF documents are processed using OCR to extract text

b. Text is chunked into question-answer pairs

c. Chunks are embedded using a QA-optimized model and stored in ChromaDB

Query Pipeline:

1. User asks a question through the Streamlit interface
2. Query is sent to the DocumentRetriever
3. Vector search finds the most relevant document chunks
4. Relevant chunks are retrieved from the vector store
5. Context is provided to the LLM (Qwen3:0.6b via Ollama)
6. LLM generates a response based on the context
7. Answer is presented to the user with source attribution

Data Processing and Ingestion

OCR Text Extraction

- Implemented Tesseract OCR for image-based PDFs, successfully extracting text despite formatting challenges

Question-Answer Chunking

- Developed custom regex patterns to identify Q&A boundaries in OCR output, preserving the natural FAQ structure

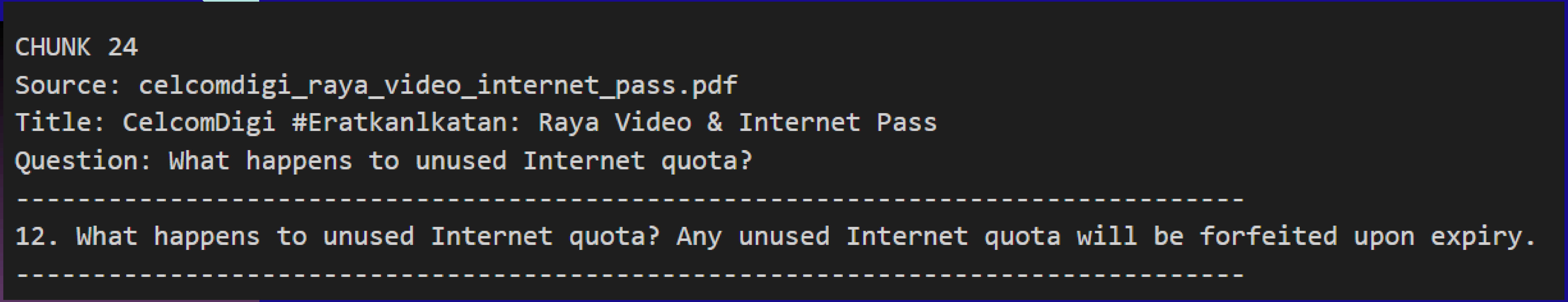
OCR-Specific Text Cleaning

- Applied advanced cleaning with telecom-specific handling, fixing OCR artifacts and standardizing text format

Hierarchical Metadata Extraction

- Captured document source, title, question headings, and content to enhance retrieval performance

Data Processing and Ingestion (processor.py) - Continue



```
CHUNK 24
Source: celcomdigi_raya_video_internet_pass.pdf
Title: CelcomDigi #Eratkanlkatan: Raya Video & Internet Pass
Question: What happens to unused Internet quota?
--------------------------------------------------------------------------------
12. What happens to unused Internet quota? Any unused Internet quota will be forfeited upon expiry.
--------------------------------------------------------------------------------
```

This image shows a perfectly processed document chunk from our system, demonstrating several key features of our chunking approach:

Metadata:

- “Source” : The source document which the content came from
- “Title”: First line of document
- “Question”: Header extracted by regex
- Last line is the actual chunk that will proceed to be embedded by a QA - Specific Embedding Model (sentence-transformers/multi-qa-MiniLM-L6-cos-v1) instead of general purpose models that embed based of semantic meaning, ignoring question answer relationships.

Retrieval Pipeline

QA-Specific Embedding Model

- Choice: (multi-qa-MiniLM-L6-cos-v1)
- specifically tuned for question-answer matching
- Outperforms general-purpose embeddings by better understanding question intent

Retrieval Mechanism

- Choice: top-k with consine similarity
- Reason: Popular, Simple, Fast, (*Note: Can look up some other methods online)*

Vector Database

- Choice: ChromaDB
- Reason: Popular, Referenced a lot online

LLM Implementation

Model Selection and Integration:

Choice: **Qwen0.6b**

Rationale:

- **Transitioned from Mistral-7b to Qwen3:0.6b**
- Qwen is the one of the current most popular open source models with reasoning capabilities (Although Overkill IMO)
- Chose to downsize it due to testing efficiency reasons but personally think it is sufficient for such a small RAG use case.
- More effort was put into making sure chunks are partitioned logically.

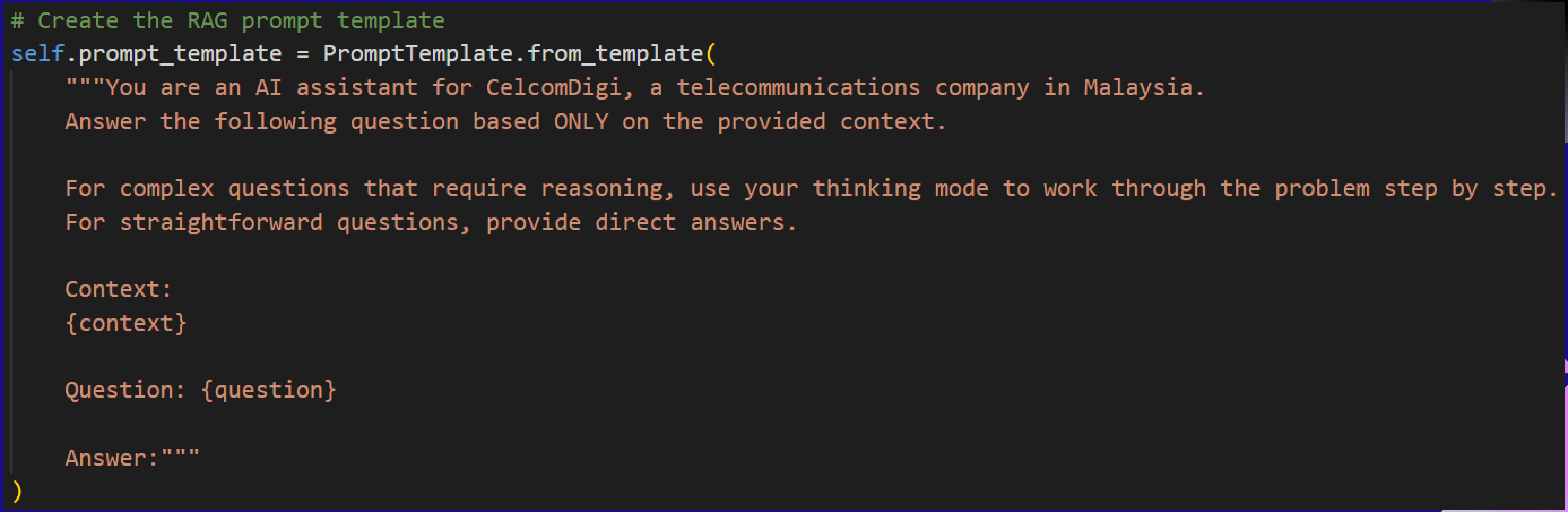
Rationale (Cont.) :

- **Run Locally,**
- **Decreases delay + more privacy.**
- **Decreases Cost.**

LLM Implementation (Cont.)

Prompt Optimization:

- Engineered prompt template specifically for knowledge-based Q&A

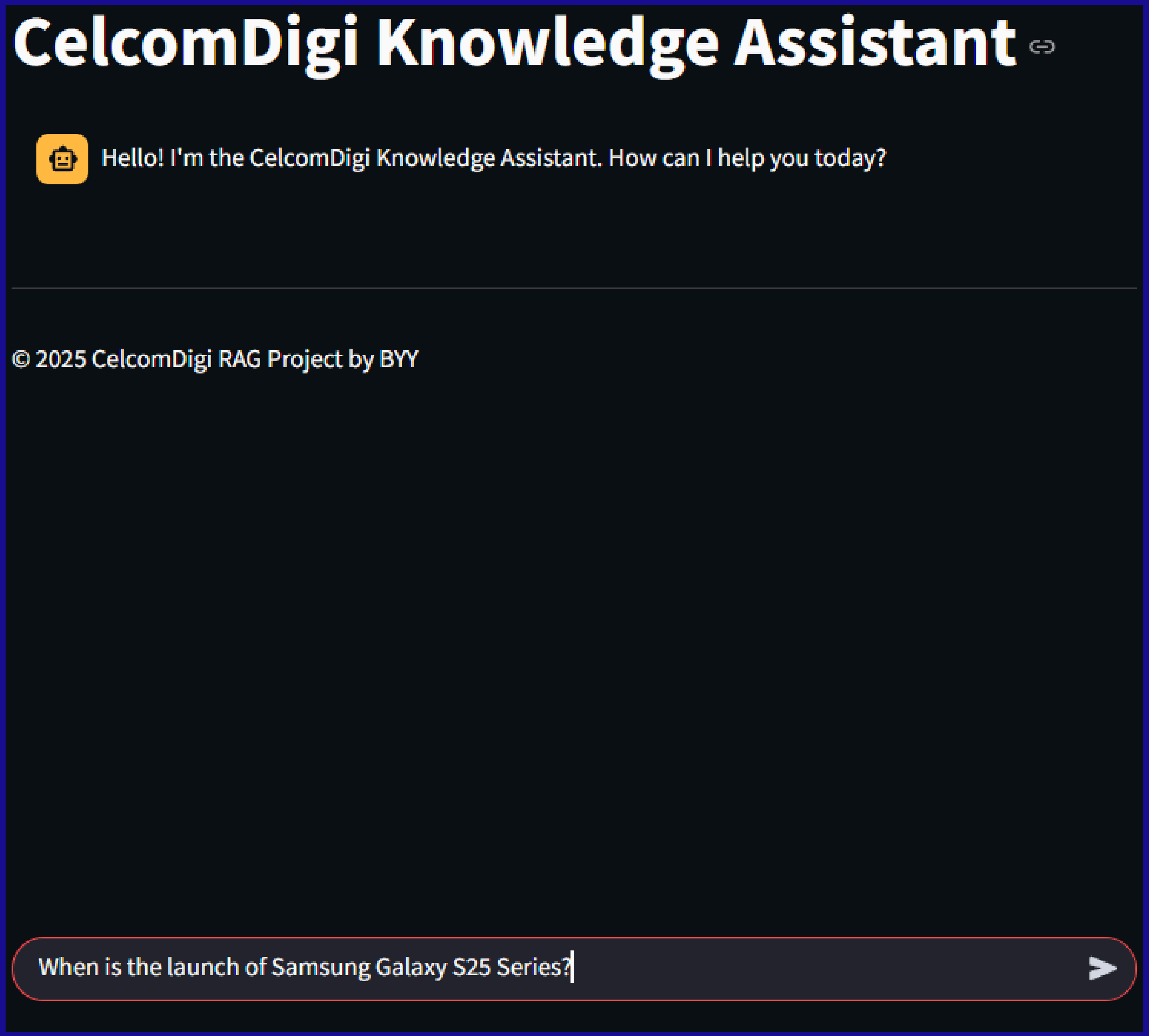


```python
# Create the RAG prompt template
self.prompt_template = PromptTemplate.from_template(
    """You are an AI assistant for CelcomDigi, a telecommunications company in Malaysia.
    Answer the following question based ONLY on the provided context.

    For complex questions that require reasoning, use your thinking mode to work through the problem step by step.
    For straightforward questions, provide direct answers.

    Context:
    {context}

    Question: {question}

    Answer:"""
)
```

Streamlit GUI

User Flow

(1) User enters a question about CelcomDigi



CelcomDigi Knowledge Assistant

Hello! I'm the CelcomDigi Knowledge Assistant. How can I help you today?

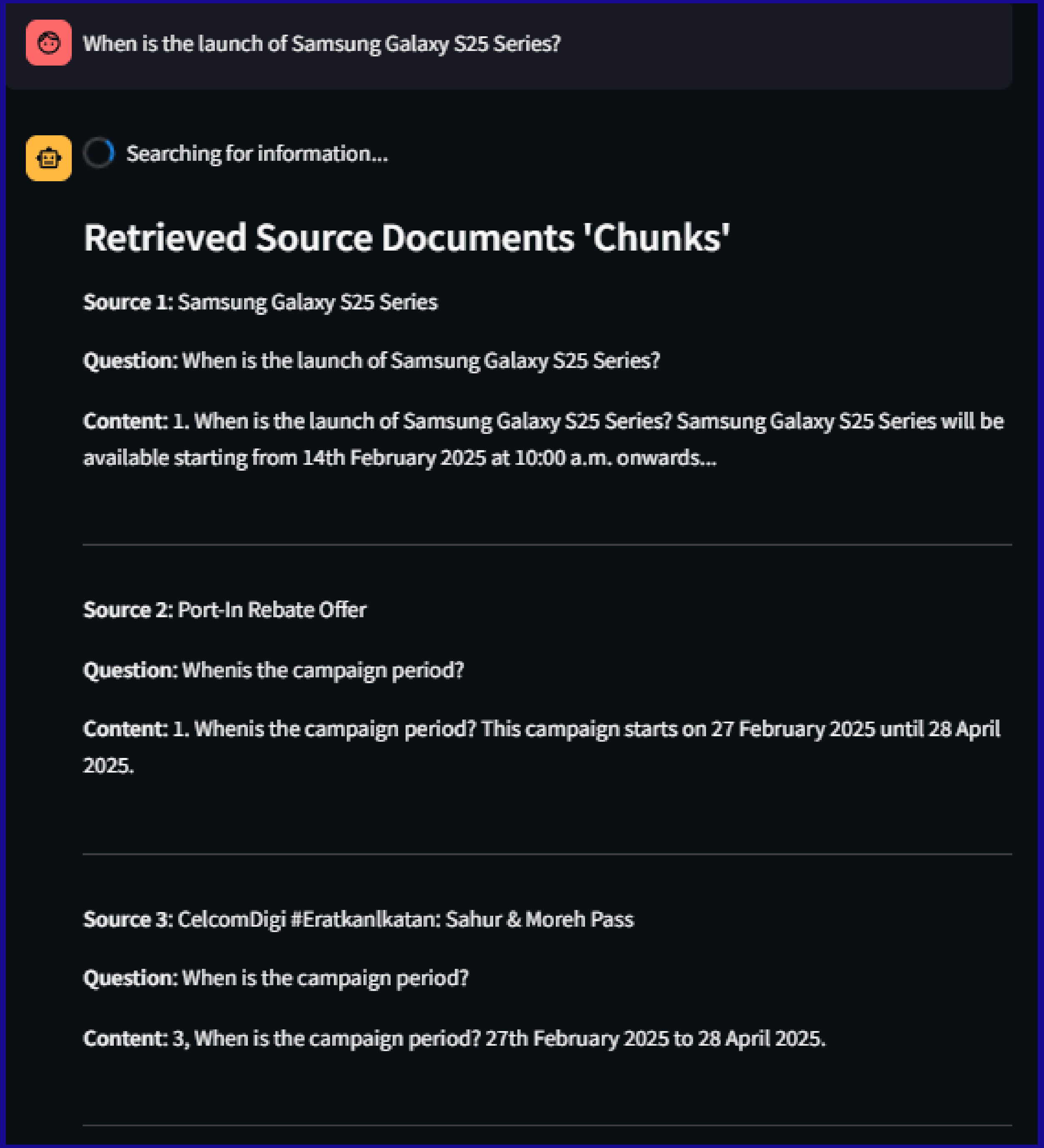
© 2025 CelcomDigi RAG Project by BYY

When is the launch of Samsung Galaxy S25 Series?

>>>

User Flow

(2) Source documents appear first



When is the launch of Samsung Galaxy S25 Series?

Searching for information...

Retrieved Source Documents 'Chunks'

Source 1: Samsung Galaxy S25 Series

Question: When is the launch of Samsung Galaxy S25 Series?

Content: 1. When is the launch of Samsung Galaxy S25 Series? Samsung Galaxy S25 Series will be available starting from 14th February 2025 at 10:00 a.m. onwards...

Source 2: Port-In Rebate Offer

Question: Whenis the campaign period?

Content: 1. Whenis the campaign period? This campaign starts on 27 February 2025 until 28 April 2025.

Source 3: CelcomDigi #Eratkanlkatan: Sahur & Moreh Pass

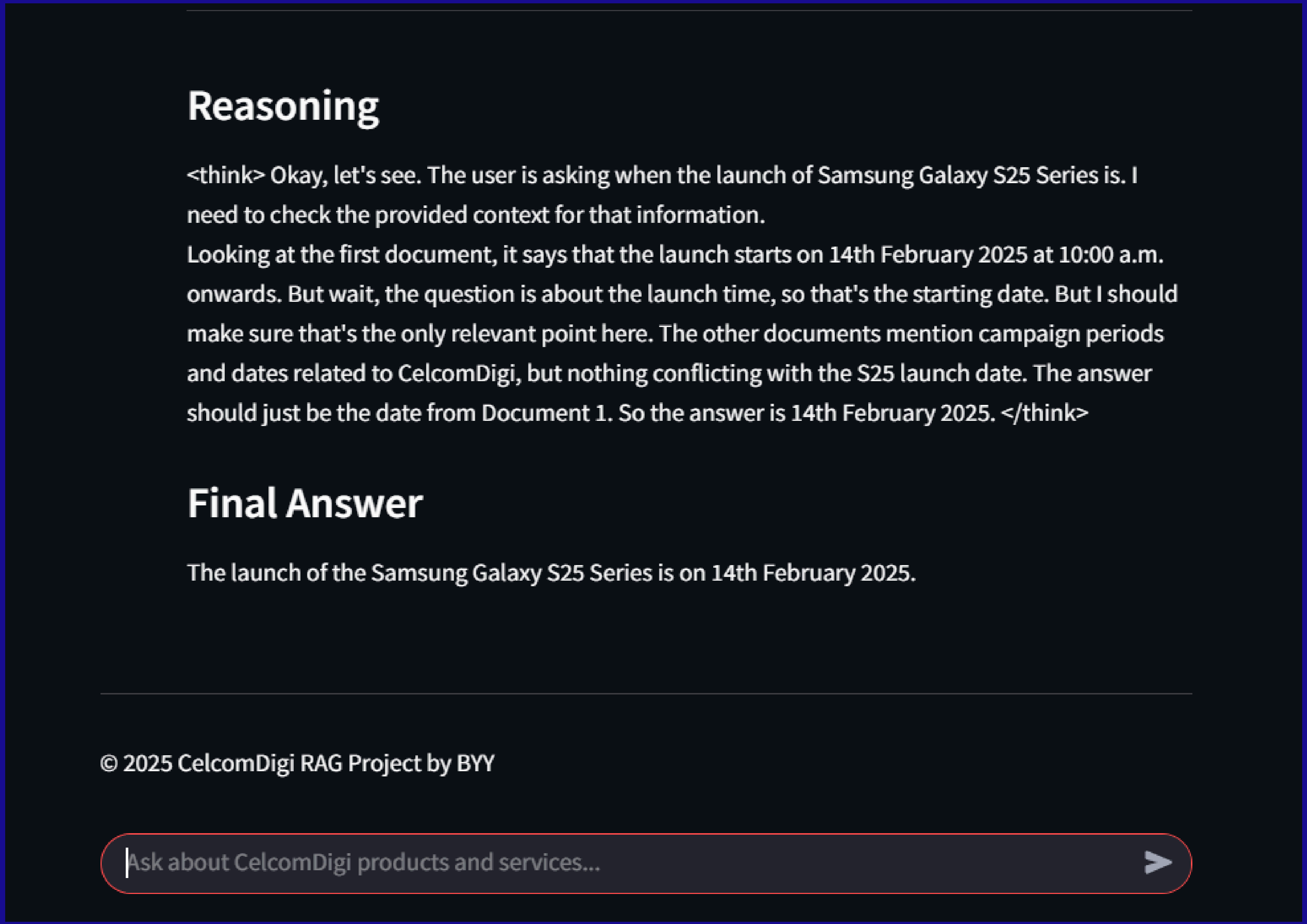
Question: When is the campaign period?

Content: 3, When is the campaign period? 27th February 2025 to 28 April 2025.

Streamlit GUI (Cont.)

User Flow

(3) Thinking process and Final Result is displayed



Reasoning

<think> Okay, let's see. The user is asking when the launch of Samsung Galaxy S25 Series is. I need to check the provided context for that information.
Looking at the first document, it says that the launch starts on 14th February 2025 at 10:00 a.m. onwards. But wait, the question is about the launch time, so that's the starting date. But I should make sure that's the only relevant point here. The other documents mention campaign periods and dates related to CelcomDigi, but nothing conflicting with the S25 launch date. The answer should just be the date from Document 1. So the answer is 14th February 2025. </think>

Final Answer

The launch of the Samsung Galaxy S25 Series is on 14th February 2025.

© 2025 CelcomDigi RAG Project by BYY

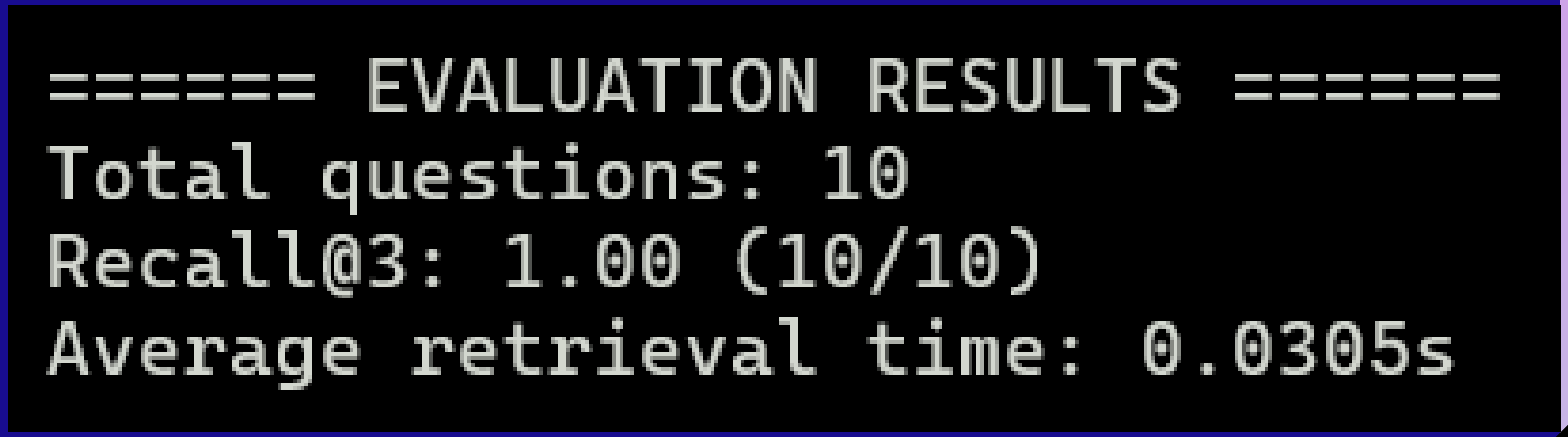
Evaluation

Evaluation:

- Choice: Recall@K
- Reason: Popular, Simple, Fast, (*Note: Can look up some other methods online)*

Methodology:

1. Created 10 diverse questions spanning all 4 CelcomDigi documents
2. Manually identified the correct source document for each test question
3. Built an evaluation script to systematically test retrieval of top 3 chunks for each question
4. Measured Recall@3 and retrieval speed for quantitative assessment



```
====== EVALUATION RESULTS ======
Total questions: 10
Recall@3: 1.00 (10/10)
Average retrieval time: 0.0305s
```

Note: Can explore more robust evaluations like comparison between response semantics and ground truth answer semantics.

Limitations and Future Improvements

Current Limitations

1. Document-Specific Processing: Our chunking and cleaning approaches are highly tailored to CelcomDigi's FAQ-style documents, potentially limiting generalizability to other document formats
2. Metadata Underutilization: Although we extract rich metadata (titles, questions, dates), the current retrieval pipeline doesn't fully leverage this information for filtering or re-ranking
3. Limited Document Scope: The system has only been tested with four documents and may face challenges scaling to hundreds or thousands of documents
4. Evaluation Simplicity: Our evaluation focused on basic Recall@k metrics rather than comprehensive user experience assessment

Future Enhancements

1. Scalable Processing Pipeline: Implement more generalizable chunking strategies using transformer-based segmentation models to handle diverse document types and structures
2. Enhanced Metadata Utilization: Develop hybrid retrieval that combines vector similarity with metadata filtering to improve precision and enable faceted search

Conclusion

In this Project I have:

1. Built an AI-powered knowledge assistant using key technologies:
    a. OCR Engine: Tesseract via pypdfium2
    b. Vector Database: ChromaDB
    c. Embedding Model: sentence-transformers/multi-qa-MiniLM-L6-cos-v1
    d. LLM: Qwen3:0.6b via Ollama
    e. Frontend: Streamlit
    f. Languages/Frameworks: Python, LangChain
2. Achieved performance metrics:
    a. Recall@3: 1.00 (100% accuracy)
    b. Average retrieval time: 33.9ms
    c. Total response time: ~1.5s

Contact me Via:

Email: bernardbyy@gmail.com

Phone Number: 018-3759772

linkedin: https://www.linkedin.com/in/boon-yong-yeow/

Source Codes: https://github.com/Bernardbyy/RAG