ThinkAi: CHAT WITH YOUR NOTES AND IDEAS

A Project Report

Submitted to the APJ Abdul Kalam Technological University

in partial fulfillment of requirements for the award of degree

Bachelor of Technology

in

Computer Science and Engineering

by

HUBAIB P(KSD20CS0756)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

LBS COLLEGE OF ENGINEERING KASARAGOD

KERALA

December 2023

LBS COLLEGE OF ENGINEERING, KASARAGOD

MULIYAR – 671 542

DEPT. OF COMPUTER SCIENCE & ENGINEERING

Vision of Department

To be a renowned centre for education, research, and innovation in the frontier areas of Computer Science and Engineering.

Mission of Department

- Establish and maintain an operational environment to acquire, impart, create and apply knowledge in Computer Science and Engineering and inter-disciplinary.
- Serve as a resource centre for innovation in design & development of hardware and software.
- Inculcate leadership qualities, professional ethics and a sense of social commitment.

DEPT. OF COMPUTER SCIENCE & ENGINEERING

LBS COLLEGE OF ENGINEERING

KASARAGOD

2023 - 24

CERTIFICATE

This is to certify that the report entitled **ThinkAi: CHAT WITH YOUR NOTES AND IDEAS** submitted by **HUBAIB P** (KSD20CS0756), to the APJ Abdul Kalam Technological University in partial fulfillment of the B.Tech. degree in Computer Science and Engineering is a bonafide record of the project work carried out by him under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Dr. Anver S.R
(Project Guide)
Associate Professor
Dept.of CSE
LBS College of Engineering
Kasaragod

Dr. Vinod George
(Project Coordinator)
Professor
Dept.of CSE
LBS College of Engineering
kasaragod

Prof . Anver S R
Professor and Head
Dept.of CSE
LBS College of Engineering
Kasaragod

DECLARATION

We hereby declare that the project report **ThinkAi: CHAT WITH YOUR NOTES AND IDEAS**, submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by us under supervision of Dr. Anver S.R

This submission represents our ideas in our own words and where ideas or words of others have been included, we have adequately and accurately cited and referenced the original sources.

We also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. We understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

HUBAIB P

Kasaragod
23-12-2023

Abstract

ThinkAI is an innovative AI-powered chatbot solution that aims to transform how users interact with and extract value from their documents, notes, and other textual materials. At its core, ThinkAI addresses the significant challenge of inefficient and labor-intensive manual context creation required by existing tools to analyze research corpora.

ThinkAI instead utilizes cutting-edge natural language processing techniques to automatically understand documents and answer natural language user queries. Specific techniques leveraged include LangChain for dialog management, the 7 billion parameter Mistral language model for deep language understanding, and BGE embeddings for representing semantic similarities. This enables ThinkAI to ingest diverse document types like PDFs, Word files, and text notes to build a contextual understanding of a user's corpus across sources.

A key advantage of ThinkAI is its emphasis on local execution without cloud dependencies. This provides greater privacy and control for users. Additionally, ThinkAI is built on open source technologies to enable customizability to specific use cases.

From an architecture standpoint, ThinkAI combines a knowledge base built using Qdrant with multi-layered neural networks for response generation powered by Mistral. This allows for quick access to relevant information extracted from source materials. The solution also incorporates productivity features like search, summaries, and contextual recommendations to enhance user workflows.

Acknowledgement

We take this opportunity to express our deepest sense of gratitude. We express our sincere thanks to our project guide Prof . Anver S R, Associate professor Head of Department,Computer Science and Engineering, LBS College of Engineering Kasaragod for providing us with all the necessary facilities, guidance, mentorship and support.

We would like to express our sincere gratitude to the project coordinator Dr. Vinod George, professor department of Computer Science and Engineering, LBS College of Engineering Kasaragod for the support and co-operation.

Finally I thank my family, and friends who contributed to the succesful fulfilment of this seminar work.

HUBAIB P

List of Figures

Chapter 1

INTRODUCTION

The accelerating pace of knowledge creation places immense demands on today's students and researchers. As the volume of published research explodes, scholars struggle under the burden of information overload. Critical challenges arise in managing, synthesizing, and extracting key insights from ever-growing corpora of literature, notes, and other unstructured materials. Yet existing solutions fail to adequately empower users, instead requiring intensive manual effort to provide structure and context. This breeds frustration, wastes precious time, and ultimately impedes the research process itself.

To overcome these limitations, we present ThinkAI - an AI-powered chatbot that fundamentally transforms how users interact with and derive value from their documents, notes, and ideas. ThinkAI pioneers a conversational interface powered by cutting-edge natural language processing that removes the need for manual context creation. Users can simply chat with their materials, asking questions, retrieving key passages, and uncovering connections as easily as messaging a colleague.

Under the hood, ThinkAI ingests papers, annotations, margins notes, and more into an intelligent knowledge base. Advanced neural networks encode both semantic and contextual representations using techniques like the 7B parameter Mistral language model and BGE embeddings. This allows ThinkAI to parse everything from handwritten notes to journal articles with human-like comprehension. Users are freed from laborious pre-processing and metadata creation. They can focus entirely on critical thinking and knowledge discovery.

The implications are profound. ThinkAI promises to radically augment scholars' productivity, accelerate research, foster deeper insights, and democratize access to impactful AI. It represents a new paradigm in human-computer interaction that overcomes the constraints of existing tools. The future of research powered by AI is here - and it converses fluently in natural language.

1.1 Problem statement

"Develope a privacy-focused AI chatbot using Langchain, Mistral 7B instruct model, and BGE embedding model to extract information from diverse documents to improve the inefficiencies in existing solutions for managing and analysing research materials."

1.2 Features

1. **Open-source tech stack:** No vendor lock-in, allowing for flexibility and customization
2. **Multi-contextual understanding:** Analyzes information from multiple documents (PDFs, Word files, etc.)
3. **Natural language processing:** Answers user queries with accuracy and fluency
4. **Local execution:** Operates entirely on the user's device for data security and privacy
5. **qdrant integration:** Efficiently stores and manages document embeddings
6. **Langchain integration:** Streamlines document processing and information retrieval
7. **Mistral 7B instruct model:** Generates informative and comprehensive responses
8. **BGE embedding model:** Enables semantic analysis for contextual understanding

9. **Empowered user control:** Open-source nature allows users to modify and adapt the chatbot to their specific needs
10. **Enhanced research and learning:** Seamless access and comprehension of information from diverse sources
11. **Improved productivity:** Natural language interaction facilitates efficient information retrieval and analysis
12. **Privacy-conscious research:** User data and documents remain secure and private on their own devices
13. **Offline accessibility:** Operates without the need for an internet connection, ensuring uninterrupted access

1.3 Existing work

1.3.1 Manual Context Creation:

- Many current solutions in the realm of information management require users to manually input context and structure to their notes and documents.
- This manual process is not only time-consuming but also prone to errors, potentially leading to inconsistencies in data organization.

1.3.2 Limited Contextual Understanding:

- Existing chatbots often struggle to grasp the full context of user queries.
- Difficulty arises, especially when dealing with information from various sources, resulting in responses that may be irrelevant or inaccurate due to a lack of comprehensive understanding.

1.3.3 Centralized Data Storage:

- A prevalent concern with numerous solutions is the centralized storage of user data and documents.

- This approach raises privacy concerns as user information is vulnerable to potential breaches, and it also introduces security risks associated with centralized data storage systems.

1.3.4 Lack of Open-Source Options:

- Many currently available solutions are proprietary, limiting user control and customization.
- The absence of open-source alternatives restricts users from tailoring the software to their specific needs, hindering innovation and collaborative development.

In addressing these limitations, future developments could focus on automating context creation, enhancing contextual understanding through advanced AI models, exploring decentralized data storage solutions for improved privacy, and fostering the development of open-source options to empower users with greater control and customization capabilities.

1.4 Limitations

1.4.1 Inefficient:

Manual context creation is a significant bottleneck, hindering user productivity and research efficiency. The time-consuming nature of this process can impede the swift organization and retrieval of information, affecting the overall workflow of users engaged in research or knowledge management tasks.

1.4.2 Inaccurate:

Limited contextual understanding can lead to irrelevant or inaccurate responses, frustrating users and hindering research progress. Chatbots and information systems that lack a comprehensive grasp of user queries and context may provide information that is not pertinent or may even be incorrect, negatively impacting the quality of research outcomes.

1.4.3 Unsecure:

Centralized data storage poses privacy risks and security vulnerabilities. Storing user data on centralized servers makes it susceptible to potential breaches, raising concerns about the privacy of sensitive information and exposing the system to security threats. This compromises the integrity and confidentiality of user data.

1.4.4 Limited Control:

Proprietary solutions restrict user control and customization, hindering flexibility and adaptability. Users are often constrained by the functionalities and configurations predetermined by proprietary software, limiting their ability to tailor the system to their specific needs. This lack of control impedes adaptability to diverse workflows and preferences.

In addressing these limitations, future developments should aim to streamline context creation processes, enhance contextual understanding through advanced AI models, explore secure and decentralized data storage solutions, and promote open-source options to empower users with greater control and customization capabilities.

Chapter 2

LITERATURE REVIEW

2.1 Extractive Text Summarization Using Word Vector Embedding

Extractive Text Summarization Using Word Vector Embedding, A. Jain et al. [1], 2017. The provided study focuses on text summarization, a vibrant area of research aimed at distilling relevant information from large documents across various domains such as finance, news media, academics, and politics. The authors propose an approach for supervised extractive summarization, leveraging a combination of feature extraction and neural network techniques. The study evaluates the effectiveness of their method using the Document Understanding Conferences 2002 dataset and compares it against various online extractive text summarizers. Text summarization, particularly based on either abstractive or extractive methods, is identified as a popular approach. The paper leans towards extractive summarization, which involves gathering relevant sentences from documents. The distinction between abstractive and extractive summarization methods is outlined, emphasizing the simplicity of the latter. The methodology section details the approach for text summarization as a binary classification problem. The explored text is categorized as either relevant for inclusion in the summary or irrelevant. The document is broken down into sentences, and features are extracted. These features are then used to train a neural network for predicting the inclusion of sentences in the summary. The proposed method incorporates both standard features and word vector embedding-based features to enhance summarization accuracy. The paper identifies four major challenges in extractive text summarization, including the identification of important information, removal of irrelevant details, minimizing unnecessary information, and assembling relevant information into a coherent report. The study evaluates its proposed method against challenges by employing a good set of features followed by a neural network for supervised extractive summarization. The inclusion of word vector embedding-based features contributes to higher accuracy, as demonstrated through testing against various online extractive text summarizers using the DUC 2002 dataset. The paper concludes by summarizing the proposed methodology's effectiveness and suggests future research directions. The combination of feature extraction and neural networks for supervised extractive summarization proves promising, highlighting the potential for further advancements in the field.

2.2 KBot: A Knowledge Graph Based ChatBot for Natural Language Understanding Over Linked Data

Literature Review KBot: A Knowledge Graph Based ChatBot for Natural Language Understanding Over Linked Data, A. Ait-Mlouk et al. [2], 2020. The paper addresses the growing significance of chatbots in leveraging linked data, specifically knowledge bases (KBs), to make structured data accessible and useful for end-users. The authors highlight the challenges associated with building chatbots over linked data, emphasizing the importance of user query understanding, support for multiple knowledge bases, and handling multilingual aspects. The review of related works in the field is crucial to understanding the context and evolution of chatbots over linked data. The paper traces the historical evolution of chatbots from their inception in the 1960s with systems like Eliza, Parry, and Alice, which were primarily based on text conversation. It notes the significant progress made over the decades, leading to the development of sophisticated AI chatbots such as Siri, Cortana, Google Assistant, and others by major companies. The overview provides context for the reader by highlighting the trajectory of chatbot development and its integration into various platforms and applications. In the context of linked data, the authors emphasize the primary goal of chatbot systems—to retrieve relevant information from one or multiple knowledge bases using natural language understanding (NLU) and semantic web technologies. They underscore the transformation of natural language into SPARQL queries as a key mechanism for achieving this goal. The literature review acknowledges the progress in chatbot research within the linked data domain but identifies persistent challenges, including user query understanding, intent classification, multilingual support, handling multiple knowledge bases, and understanding analytical queries. The authors discuss the challenges faced by existing linked data chatbots, emphasizing the need for substantial training data, which is often expensive and challenging to obtain. They recognize the recent growth in linked data development and its impact on chatbot advancements in both research and industry. Despite this progress, the paper asserts that challenges such as user query understanding, intent classification, multilingual aspects, support for multiple knowledge bases, and analytical query comprehension persist in the field. The literature review introduces the proposed solution, KBot, as a chatbot that addresses several challenges in the linked data domain. It emphasizes KBot's ability to compete with existing linked data chatbots in terms of performance. The authors outline key contributions, including the design and implementation of KBot, a machine learning model (SVM) for intent classification, an analytical queries engine for data exploration, and scalability features that allow the addition of new knowledge bases, support for multiple languages, and flexibility for diverse tasks. In summary, the literature review provides a comprehensive overview of the historical evolution of chatbots, their integration with linked data, and the persisting challenges in the field. It sets the stage for the proposed solution, KBot, by establishing the context and underscoring the need for advancements in linked data chatbots.

2.3 DocChat: An Information Retrieval Approach for Chatbot Engines Using Unstructured Documents

DocChat: An Information Retrieval Approach for Chatbot Engines Using Unstructured Documents, Zhao Yan et al. [3], 2016. The paper addresses a critical challenge in the development of chatbot engines, specifically focusing on the limitation of existing engines that rely on predefined utterance-response (Q-R) pairs. The study introduces ”DocChat,” a novel information retrieval approach that leverages unstructured documents instead of Q-R pairs to respond to user utterances. To understand the context and significance of this novel approach, a literature review covers key themes related to chatbot development, existing methods, and challenges in the field. Building chatbot engines capable of natural language interaction with humans represents a formidable challenge in artificial intelligence. The paper acknowledges the complexity of this problem and emphasizes the rapid growth of social media platforms, community question answering (CQA) websites, and the vast amount of Q-R pairs that have become available. The explosion of data-driven chatbot approaches is discussed as a response to this growing corpus of Q-R pairs. The paper categorizes existing methods for short text conversation (STC) into two main types: retrieval-based methods and generation-based methods. Retrieval-based methods involve matching the current utterance with existing Q-R pairs, and generation-based methods use an encoder-decoder framework to generate responses. However, both approaches have drawbacks, such as intractability in collecting Q-R pairs for specific domains and limitations in the fluency and naturality of machine-generated text. To address the limitations of existing methods, the paper introduces ”DocChat” as a response retrieval approach based on unstructured documents. Unlike traditional Q-R pair-based methods, DocChat selects a response sentence directly from given documents by ranking all possible sentences using features designed at different levels of granularity. This innovative approach aims to improve the adaptability of chatbot engines to various topics and ensures the fluency and naturality of responses since they are drawn from existing documents. The literature review highlights the promising results obtained through experiments, emphasizing the effectiveness of DocChat in both question-answering (QA) and chatbot scenarios. The approach’s adaptability and the natural fluency of responses are identified as key advantages. Additionally, the paper emphasizes the contributions of DocChat, positioning it as a solution that complements chatbot engines using Q-R pairs as their primary source of responses.

In summary, the literature review provides a comprehensive background on the challenges associated with chatbot development, the limitations of existing methods, and the introduction of DocChat as an innovative response retrieval approach. It sets the stage for the paper’s contributions and showcases the need for advancements in chatbot technology.

Chapter 3

METHODOLOGY

1.Input Text Preprocessing:

- PDFs, Word docs, etc. are ingested.
- Text is extracted, cleaned and tokenized into sentences/paragraphs.

2.Embedding Generation with GRE:

- GRE model encodes each sentence/paragraph into 768-dimensional embeddings.
- Encodes semantics and relationships between concepts in vectors.

3.qdrant Indexing:

- GRE embeddings indexed into a column store.
- Enables quick KNN and vector similarity searches.
- Scales to billions of embeddings.

4.Context Embedding Indexes:

- Langchain tracks which texts map to which embeddings.
- Creates indexes to identify origin contexts during retrieval.

5.User Query Preprocessing:

- User questions/queries tokenized and cleaned.
- Encoded into GRE embeddings.

6.Context Retrieval:

- GRE embedded query used for vector similarity search in qdrant.
- Most relevant contexts to the query retrieved using HNSW(Hierarchical Navigable Small World graphs).

7. Response Generation with Mistral 7B:

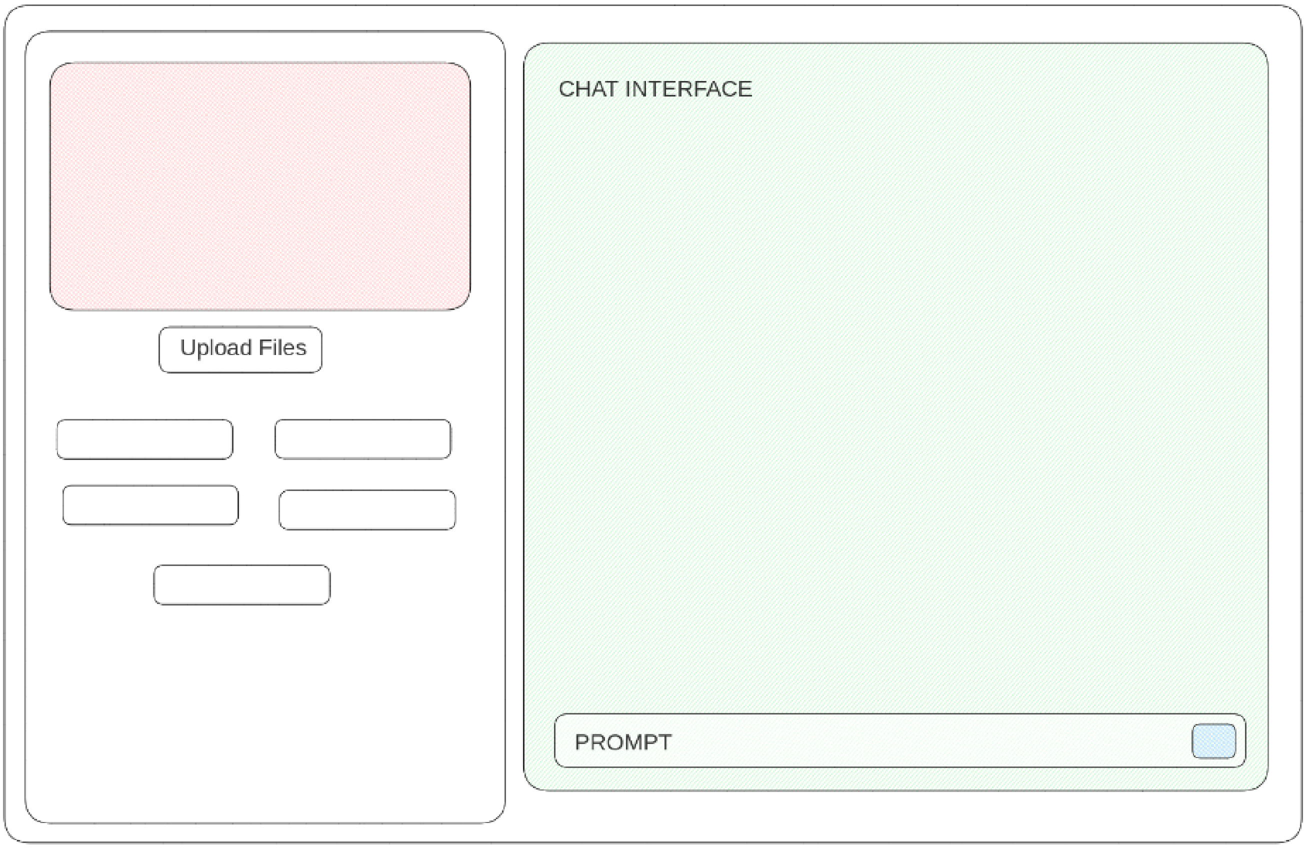


Figure 3.1: Low Fedility Wireframe

- Retrieved contexts fused and fed as prompt to 7B parameter LM.

- Models long-term dependencies in text to generate informative response.

8. Return Response to User:

- Fluent, accurate Mistral 7B generated responses.

- Further iterates on user feedback.

3.1 Proposed system

3.1.1 Tech stack

Qdrant (Vectore store DB): Open-source database for efficient storage and management of document embeddings.

LangChain (Ai Pipeline): Open-source framework for building AI pipelines and tools for document processing.

Mistral-7B-Instruct-v0.1 (LLM):Open-source large language model for generating

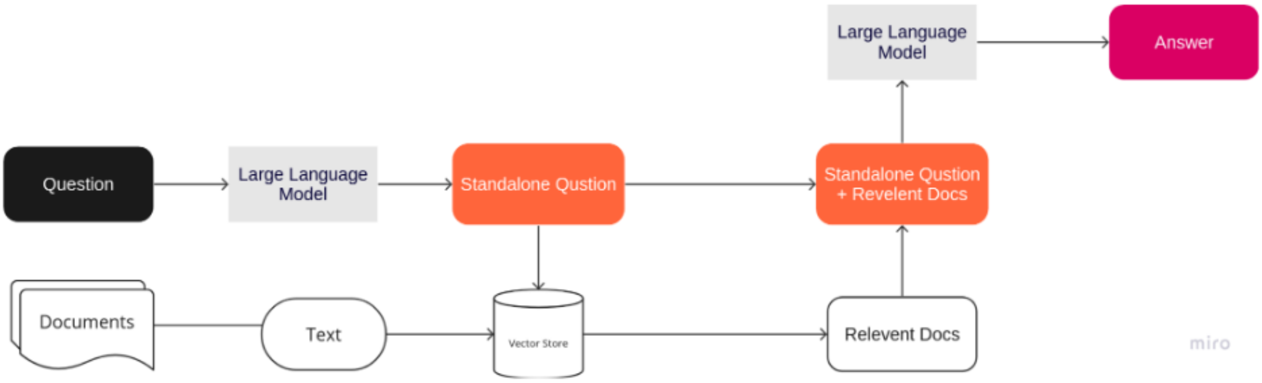


Figure 3.2: Overall Diagram

accurate and informative responses.

BAAI-bge-large-en (Embedding):Open-source library for efficient text encoding and semantic analysis.

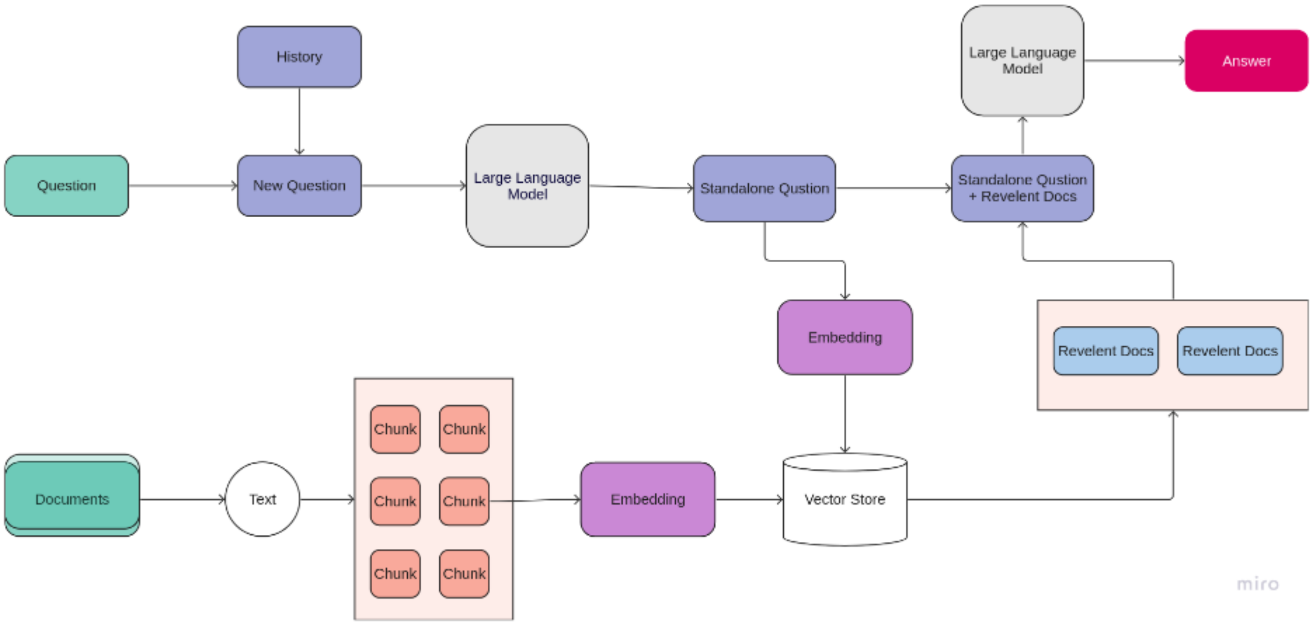
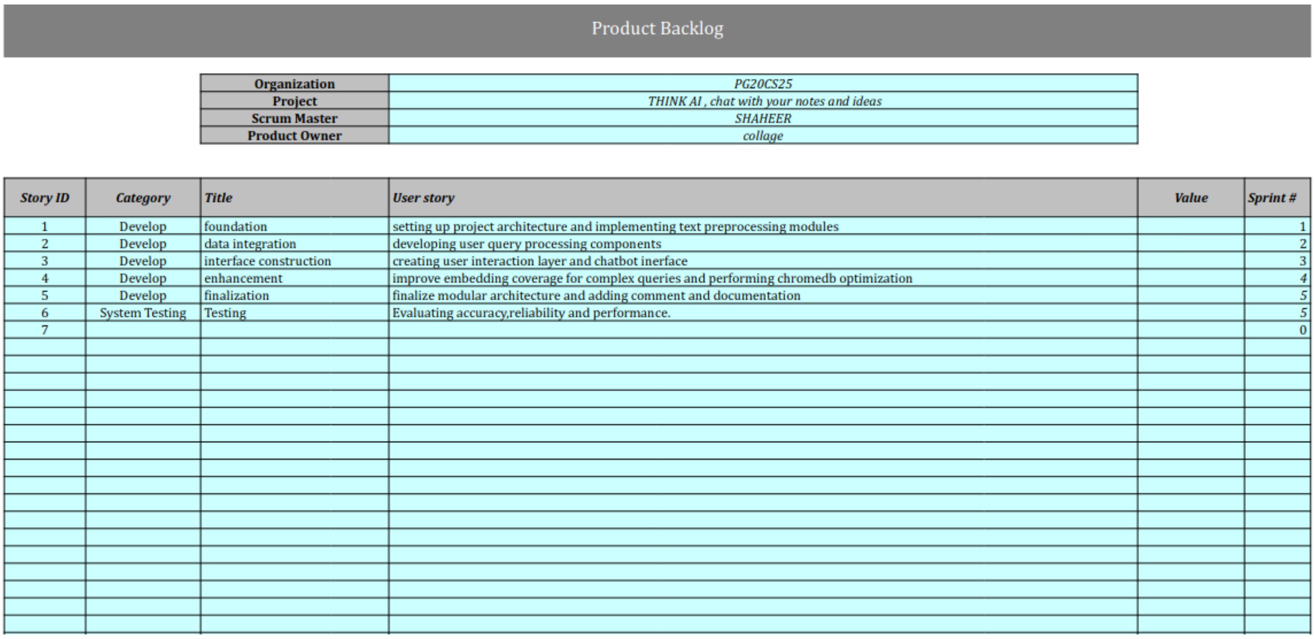


Figure 3.3: Detailed Architecture

Chapter 4

SCRUM

4.1 Backlog



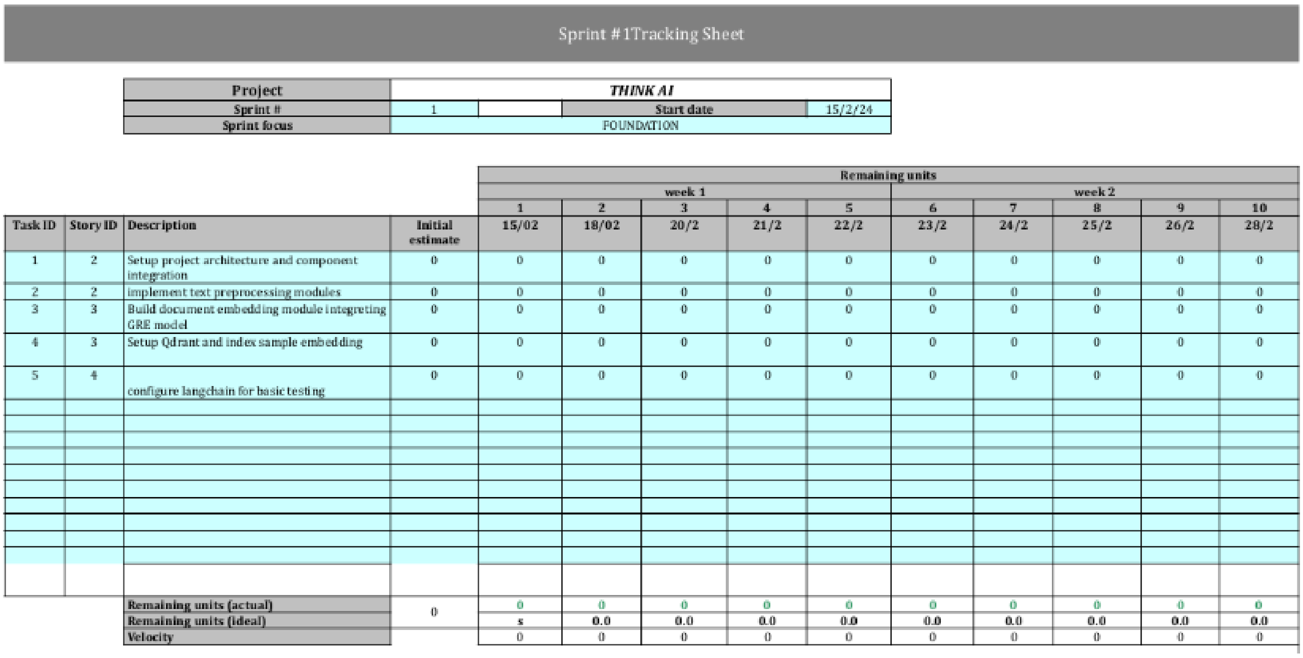
Product Backlog

Organization	*PG20CS25*
Project	*THINK AI , chat with your notes and ideas*
Scrum Master	*SHAHEER*
Product Owner	*collage*

Story ID	***Category***	***Title***	***User story***	***Value***	***Sprint #***
1	Develop	foundation	setting up project architecture and implementing text preprocessing modules		1
2	Develop	data integration	developing user query processing components		2
3	Develop	interface construction	creating user interaction layer and chatbot inerface		3
4	Develop	enhancement	improve embedding coverage for complex queries and performing chromedb optimization		*4*
5	Develop	finalization	finalize modular architecture and adding comment and documentation		*5*
6	System Testing	Testing	Evaluating accuracy,reliability and performance.		*5*
7					0

Figure 4.1: Product Backlog

4.2 Sprint1



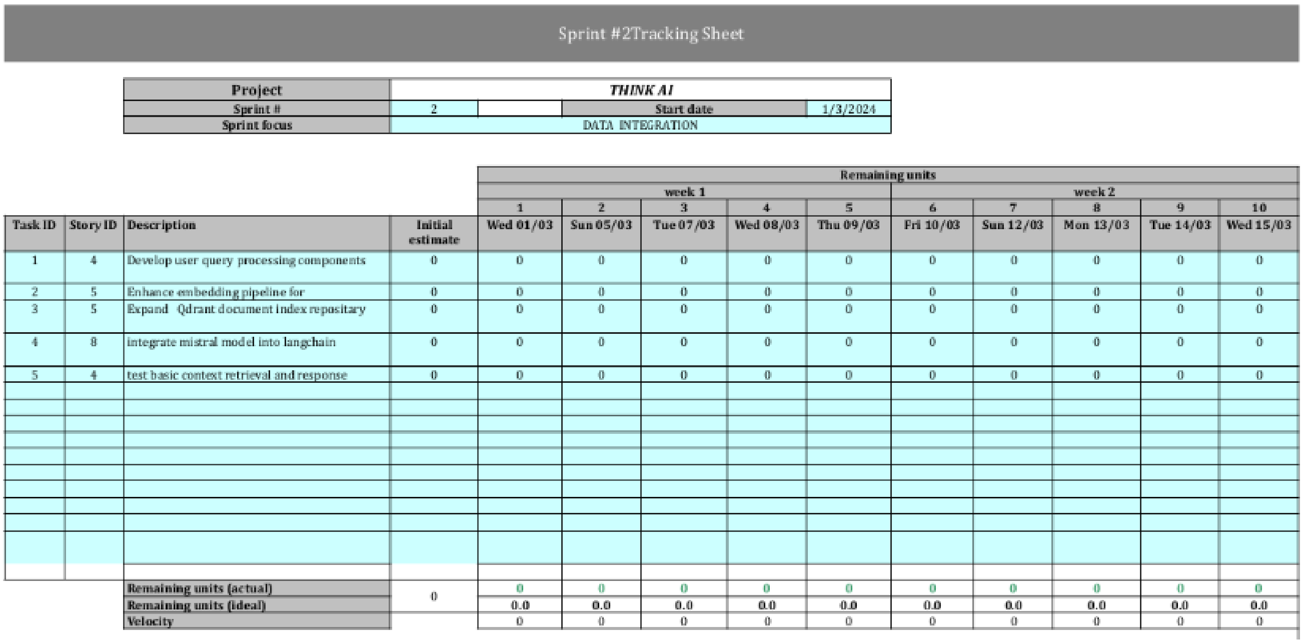
Sprint #1Tracking Sheet

Project	*THINK AI*			
Sprint #	1		Start date	15/2/24
Sprint focus	FOUNDATION			

				Remaining units									
				week 1					week 2				
				1	2	3	4	5	6	7	8	9	10
Task ID	Story ID	Description	Initial estimate	15/02	18/02	20/2	21/2	22/2	23/2	24/2	25/2	26/2	28/2
1	2	Setup project architecture and component integration	0	0	0	0	0	0	0	0	0	0	0
2	2	implement text preprocessing modules	0	0	0	0	0	0	0	0	0	0	0
3	3	Build document embedding module integreting GRE model	0	0	0	0	0	0	0	0	0	0	0
4	3	Setup Qdrant and index sample embedding	0	0	0	0	0	0	0	0	0	0	0
5	4	configure langchain for basic testing	0	0	0	0	0	0	0	0	0	0	0
		Remaining units (actual)	0	0	0	0	0	0	0	0	0	0	0
		Remaining units (ideal)		s	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
		Velocity		0	0	0	0	0	0	0	0	0	0

Figure 4.2: Sprint1

4.3 Sprint2



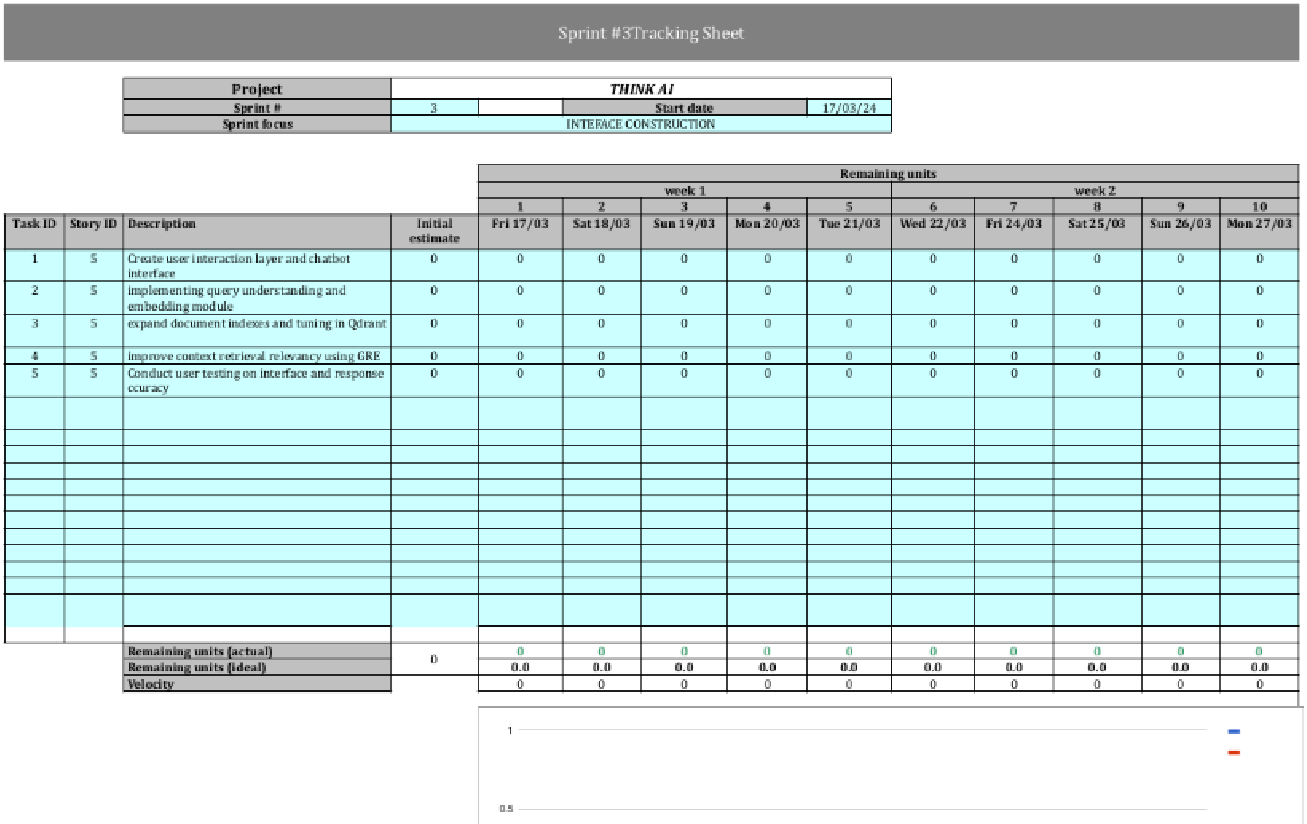
Sprint #2Tracking Sheet

Project	THINK AI		
Sprint #	2	Start date	1/3/2024
Sprint focus	DATA INTEGRATION		

				Remaining units									
				week 1					week 2				
				1	2	3	4	5	6	7	8	9	10
Task ID	Story ID	Description	Initial estimate	Wed 01/03	Sun 05/03	Tue 07/03	Wed 08/03	Thu 09/03	Fri 10/03	Sun 12/03	Mon 13/03	Tue 14/03	Wed 15/03
1	4	Develop user query processing components	0	0	0	0	0	0	0	0	0	0	0
2	5	Enhance embedding pipeline for	0	0	0	0	0	0	0	0	0	0	0
3	5	Expand Qdrant document index repositary	0	0	0	0	0	0	0	0	0	0	0
4	8	integrate mistral model into langchain	0	0	0	0	0	0	0	0	0	0	0
5	4	test basic context retrieval and response	0	0	0	0	0	0	0	0	0	0	0
		Remaining units (actual)	0	0	0	0	0	0	0	0	0	0	0
		Remaining units (ideal)		0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
		Velocity		0	0	0	0	0	0	0	0	0	0

Figure 4.3: Sprint2

4.4 Sprint3



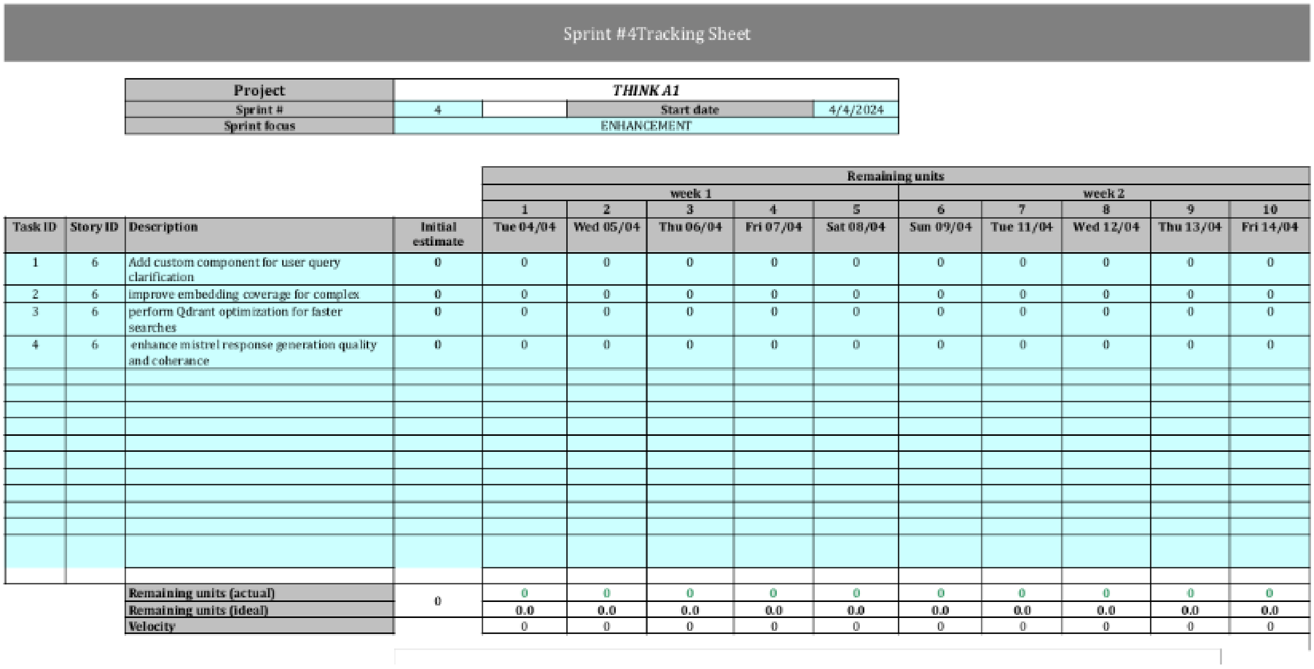
Sprint #3Tracking Sheet

Project	*THINK AI*		
Sprint #	3	Start date	17/03/24
Sprint focus	INTERFACE CONSTRUCTION		

				Remaining units									
				week 1					week 2				
				1	2	3	4	5	6	7	8	9	10
Task ID	Story ID	Description	Initial estimate	Fri 17/03	Sat 18/03	Sun 19/03	Mon 20/03	Tue 21/03	Wed 22/03	Fri 24/03	Sat 25/03	Sun 26/03	Mon 27/03
1	5	Create user interaction layer and chatbot interface	0	0	0	0	0	0	0	0	0	0	0
2	5	implementing query understanding and embedding module	0	0	0	0	0	0	0	0	0	0	0
3	5	expand document indexes and tuning in Qdrant	0	0	0	0	0	0	0	0	0	0	0
4	5	improve context retrieval relevancy using GRE	0	0	0	0	0	0	0	0	0	0	0
5	5	Conduct user testing on interface and response ccuracy	0	0	0	0	0	0	0	0	0	0	0
		Remaining units (actual)	0	0	0	0	0	0	0	0	0	0	0
		Remaining units (ideal)		0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
		Velocity		0	0	0	0	0	0	0	0	0	0

Figure 4.4: Sprint3

4.5 Sprint4



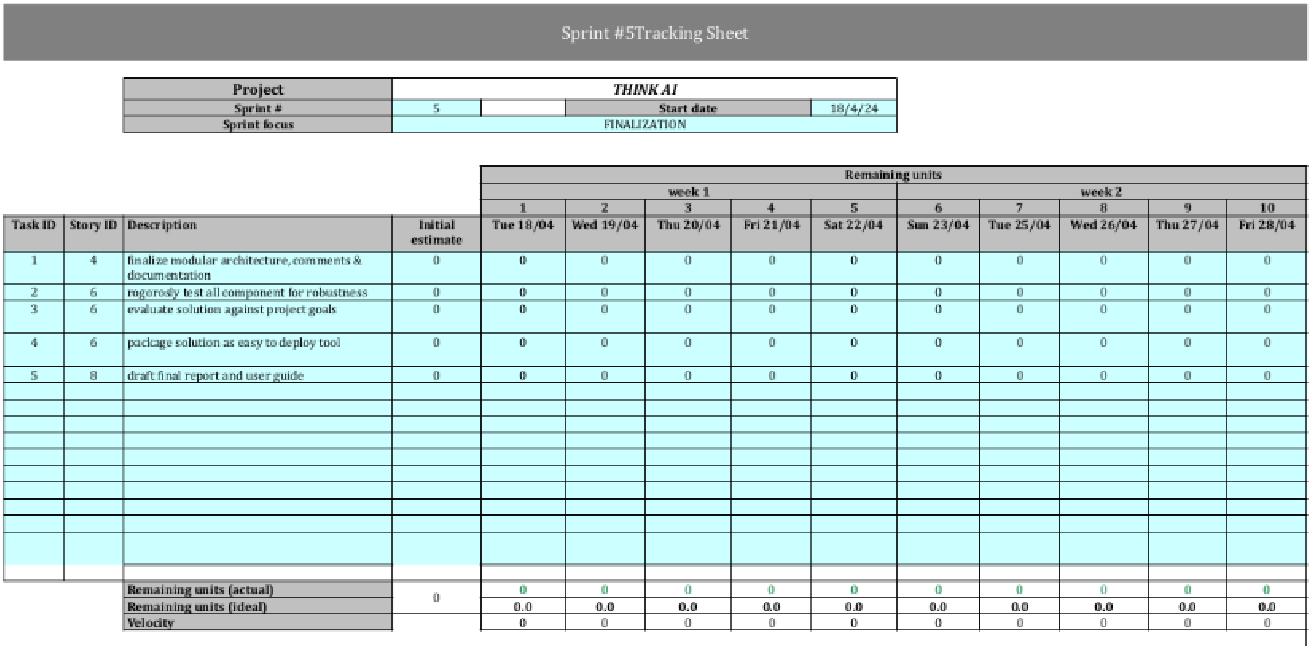
Sprint #4Tracking Sheet

Project	*THINK A1*		
Sprint #	4	Start date	4/4/2024
Sprint focus	ENHANCEMENT		

				Remaining units									
				week 1					week 2				
				1	2	3	4	5	6	7	8	9	10
Task ID	Story ID	Description	Initial estimate	Tue 04/04	Wed 05/04	Thu 06/04	Fri 07/04	Sat 08/04	Sun 09/04	Tue 11/04	Wed 12/04	Thu 13/04	Fri 14/04
1	6	Add custom component for user query clarification	0	0	0	0	0	0	0	0	0	0	0
2	6	improve embedding coverage for complex	0	0	0	0	0	0	0	0	0	0	0
3	6	perform Qdrant optimization for faster searches	0	0	0	0	0	0	0	0	0	0	0
4	6	enhance mistrel response generation quality and coherance	0	0	0	0	0	0	0	0	0	0	0
		Remaining units (actual)	0	0	0	0	0	0	0	0	0	0	0
		Remaining units (ideal)		0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
		Velocity		0	0	0	0	0	0	0	0	0	0

Figure 4.5: Sprint4

4.6 Sprint5



Sprint #5Tracking Sheet

Project	***THINK AI***		
Sprint #	5	**Start date**	18/4/24
Sprint focus	FINALIZATION		

				Remaining units									
				week 1					**week 2**				
				1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Task ID	**Story ID**	**Description**	**Initial estimate**	**Tue 18/04**	**Wed 19/04**	**Thu 20/04**	**Fri 21/04**	**Sat 22/04**	**Sun 23/04**	**Tue 25/04**	**Wed 26/04**	**Thu 27/04**	**Fri 28/04**
1	4	finalize modular architecture, comments & documentation	0	0	0	0	0	0	0	0	0	0	0
2	6	rogorosly test all component for robustness	0	0	0	0	0	0	0	0	0	0	0
3	6	evaluate solution against project goals	0	0	0	0	0	0	0	0	0	0	0
4	6	package solution as easy to deploy tool	0	0	0	0	0	0	0	0	0	0	0
5	8	draft final report and user guide	0	0	0	0	0	0	0	0	0	0	0
		Remaining units (actual)	0	0	0	0	0	0	0	0	0	0	0
		Remaining units (ideal)		0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
		Velocity		0	0	0	0	0	0	0	0	0	0

Figure 4.6: Sprint5

Chapter 5

CONCLUSION

Each chapter is to begin with a brief introduction (in 4 or 5 sentences) about its contents. The contents can then be presented below organised into sections and subsections. In conclusion, the development of an AI-powered research assistant chatbot represents a significant leap forward in enhancing the accessibility and comprehension of research information. By integrating cutting-edge NLP techniques like Langchain, Mistral, and GRE, the chatbot has successfully eliminated the cumbersome process of manual context creation. This achievement, coupled with the implementation of multi-document understanding, enables the chatbot to provide accurate and semantic responses to user queries. One of the standout features of this research assistant chatbot is its commitment to data privacy and security. The decision to process data locally without relying on cloud dependencies ensures that users have complete control over their information, addressing concerns related to privacy and security in existing solutions. The provision of an open-source stack further adds to the chatbot's appeal by empowering users with complete control and customization options based on their specific needs. Through this initiative, the chatbot not only addresses the limitations observed in existing solutions, such as poor context, lack of privacy, and restricted control but also showcases the immense potential of AI in revolutionizing how users access and comprehend research information. Overall, this AI-powered research assistant chatbot stands as a testament to the possibilities and advancements achievable through ethical and innovative applications of artificial intelligence in academia.

References

[1] A. Jain, D. Bhatia and M. K. Thakur, ”Extractive Text Summarization Using Word Vector Embedding,” 2017 International Conference on Machine Learning and Data Science (MLDS), Noida, India, 2017, pp. 51-55, doi: 10.1109/MLDS.2017.12.

[2] A. Ait-Mlouk and L. Jiang, ”KBot: A Knowledge Graph Based ChatBot for Natural Language Understanding Over Linked Data,” in IEEE Access, vol. 8, pp. 149220-149230, 2020, doi: 10.1109/ACCESS.2020.3016142.

[3] Zhao Yan, Nan Duan, Junwei Bao, Peng Chen, Ming Zhou, Zhoujun Li, and Jianshe Zhou. 2016. DocChat: An Information Retrieval Approach for Chatbot Engines Using Unstructured Documents. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, https://aclanthology.org/P16-1049.

[4] Arjun Pesaru, Taranveer Singh Gill, Archit Reddy Tangella (2023). AI ASSISTANT FOR DOCUMENT MANAGEMENT USING LANG CHAIN AND PINECONE. *https://www.irjmets.com*

[5] Haritha Akkineni, P. V. S. Lakshmi, and Lasya Sarada (2022). Design and Development of Retrieval-Based Chatbot Using Sentence Similarity. *https://link.springer.com*

[6] Oguzhan Topsakal1, and Tahir Cetin Akinci (2023). Creating Large Language Model Applications Utilizing LangChain: A Primer on Developing LLM Apps Fast. *https://as-proceeding.com*

[7] Andreas Lommatzsch and Jonas Katins (2019). An Information Retrieval-based Approach for Building Intuitive Chatbots for Large Knowledge Bases. *https://ceur-ws.org/Vol-2454/paper*$_{6}0.pdf$

[8] Norbert Braunschweiler and Rama Doddipatla and Simon Keizer and Svetlana Stoyanchev (2023). Evaluating Large Language Models for Document-grounded Response Generation in Information-Seeking Dialogues. *https://arxiv.org*

[9] Arjun Pesaru, Taranveer Singh Gill, Archit Reddy Tangella (2023). AI ASSISTANT FOR DOCUMENT MANAGEMENT USING LANG CHAIN AND PINECONE. *https://www.irjmets.com*

[10] Pinky Sitikhu, Kritish Pahi, Pujan Thapa, Subarna Shakya (2019). A Comparison of Semantic Similarity Methods for Maximum Human Interpretability. *https://arxiv.org/abs*