# Using Large Language Models and Knowledge Graphs to Improve Medical Drug Prescription and Healthcare

Kamanu David Nene

169701

Submitted in partial fulfilment of the requirements for the degree of Master of Science in Data Science of Strathmore University

**Strathmore Institute of Mathematical Sciences**

**Strathmore University**

**Nairobi, Kenya**

**June 2025**

# Declaration

I declare that this work has not been previously submitted and approved for award of a degree by this or any other University. To the best of my knowledge and belief, the proposal contains no material previously published or written by another person except where due reference is made in the thesis itself.



Name: **Kamanu David Nene**

Signature: 

Date: May 22, 2025

## Approval

The thesis of Kamanu David Nene was reviewed and approved by the following:

**Dr. Kennedy Senagi**

Supervisor,

Insitute of Mathematical Sciences

Strathmore University

**Dr. Godfrey Madigu**

Dean,

Institute of Mathematical Sciences, Strathmore University.

**Dr. Bernard Shibwabo**

Director,

Office of Graduate Studies, Strathmore University.

# Abstract

Healthcare is a critical pillar of societal well- being, ensuring that individuals receive the appropriate care to maintain or improve their health. In this context, accurate prescription of medication plays a crucial role in patient safety and treatment outcomes. However, medication errors (MEs) remain a persistent issue. These errors can lead to adverse drug reactions that jeopardize patient wellness in healthcare systems. The rapid expansion of medical data, combined with challenges in retrieving accurate information, compounds the issue. This study addresses the challenge of dosage and administration errors by improving access to accurate prescription information using a Large Language Model (LLM)-powered chatbot.The study used Design Science Research methodology. We fine-tuned the Llama Meditron- 7B model with 25,547 labeled prescription entries from the PubMed database. Using the Retrieval Augmented Generation (RAG) framework, we embedded data from the 1st Edition of the Kenya National Medicine Formulary 2023 and stored it in a Qdrant vector store for efficient retrieval. The model was trained on the NVIDIA A100 GPU 16GB in Google Colab. The model's performance was evaluated using Recall-Oriented Understudy for Gisting (ROUGE) and Bilingual Evaluation Understudy (BLEU) metrics, and the results were compared to the baseline Meditron and Generative Pre-trained Transformer (GPT)-3.5 models. The Meditron-7B model significantly outperformed its counterparts, achieving ROUGE-1 scores of 0.3648 for recall and 0.5321 for precision, with a BLEU score of 0.1268. This demonstrates a marked improvement in the precision and efficiency of prescrip- tion information retrieval, offering an enhanced decision-support tool for healthcare providers. This decision support system can reduce MEs and improve access to reliable prescription data. This LLM-powered chatbot has the potential to transform healthcare, especially in resource-constrained medical environments.

# Table of Contents

# List of Figures

# List of Abbreviations

| | |
|---|---|
| AI | Artificial Intelligence |
| BLEU | Bilingual Evaluation Understudy |
| EHR | Electronic Health Records |
| KNMF | Kenya National Medicines Formulary |
| LLM | Large Language Model |
| ME | Medical Error |
| NLP | Natural Language Processing |
| RAG | Retrieval Augmented Generation |
| ROUGE | Recall-Oriented Understudy for Gisting Evaluation |
| WHO | World Health Organization |

# Definition of Terms

| | |
|---|---|
| **Artificial Intelligence (AI)** | Refers to a machine's capability to execute tasks that usually demand human intelligence such as using data, algorithms, and computational power to learn, reason, and solve problems (Korteling et al., 2021). |
| **Cross-Industry Standard Process for Data Mining (CRISP-DM)** | Describes a structured framework for planning and carrying out data mining projects (Martinez-Plumed et al., 2019). |
| **Generative Pre-trained Transformer (GPT)** | It is a deep learning-based language model developed by OpenAI that generates text that is human-like in response to a given input (Yenduri et al., 2024). |
| **Large Language Model** | Refers to a deep learning algorithm that can understand and perform a variety of natural language processing (NLP) tasks such as question answering, translation, text summarization, and creative writing(Naveed et al., 2023). |
| **Medical Error** | Refers to a mistake or oversight made in the healthcare process occurring at any stage including diagnosis, treatment, medication administration, or patient management that can harm a patient (Rodziewicz et al., 2018). |
| **Medical Prescription** | Refers to a written document that authorizes a patient to receive a specific medicine from a pharmacist and is only available to patients who have a prescription from an authorized health professional(Kumar Rai et al., 2021). |
| **Natural Language Processing (NLP)** | is a branch of computer science and AI that focuses on aiding computers to understand, process, and generate human language (Stryker and Holdsworth, 2024). |

**Retrieval Augmented Generation (RAG)** Refers to an AI framework designed to retrieve information from external knowledge sources, ensuring large language models (LLMs) generate accurate and up-to-date responses while providing users with transparency into the model's reasoning process (Martineau, 2023).

# Acknowledgement

First and foremost, I offer my sincerest gratitude to God for His unwavering guidance and strength throughout this journey. It is through His grace that I have been able to complete this project.

I am deeply grateful for the guidance and supervision provided by my esteemed supervisor and lecturer Dr. Kennedy Senagi for his important support and supervision throughout this research. His experience and insights have helped shape the direction and success of my research. I also appreciate Dr. John Olukuru for his support during the ideation of the project.

My gratitude extends to Strathmore University, particularly the Institute of Mathematical Sciences (SIMS), and the @iLabAfrica research centre. Their commitment to delivering world-class education and imparting practical and technical knowledge has been essential in enriching my academic and professional growth. I also appreciate my classmates and colleagues for peer reviewing and sharing insights.

Finally, I thank my family for all the support and encouragement they gave me through this journey.

# Dedication

This dissertation is dedicated to my father, James Kamanu, and mother Lucy Watiri, who were instrumental in molding my educational path. Their constant support and motivation throughout my early years were the pillars of my academic success.

# Chapter 1
# Introduction

## 1.1 Background of the Study

Medication errors (MEs) continue to represent a significant challenge to healthcare quality and patient safety around the world. Recent systematic reviews indicate that MEs are prevalent, particularly in outpatient and ambulatory settings, where prevalence rates for prescribed medications range from 23% to 92%. Among these, prescribing errors are the most frequently reported, with dosing errors constituting a substantial portion, occurring in up to 41% of prescribed drugs (Naseralallah and et al., 2023).

In a study conducted in emergency wards in Ethiopia, it was found that 74.4% of patients experienced at least one medication error, with omitted doses and administration mistakes being the most common types of errors identified (Martinez-Plumed et al., 2019). The implications of these findings are profound; in the United States alone, it is estimated that MEs result in approximately 1.3 million injuries annually, with fatalities occurring daily due to these errors. The World Health Organization has underscored that medication-related adverse events are a leading cause of harm within healthcare systems globally(World Health Organization, 2017).

The risk of MEs is particularly elevated during the prescription and administration of drugs. Numerous studies have highlighted that prescription errors stem from deficiencies in dosage formulation management and incomplete data (Krähenbühl-Melcher and et al., 2007; Miasso et al., 2009) . Conversely, administration errors are more closely linked to environmental and professional factors, such as stress or work overload (Donati et al., 2015; Elganzouri et al., 2009; Frith, 2013; Kendall-Gallagher and Blegen, 2009).

In intensive care units (ICUs), MEs are of paramount concern due to the complexity of care and patients' compromised physiological reserves, rendering them more vulnerable to errors (George et al., 2010; Merino et al., 2013; Salazar et al., 2011). Critically ill patients receive twice as many medications as those in other units, with most being administered intravenously. As these patients are frequently sedated and unable to consciously participate in their treatment,

reversing errors becomes challenging (Armitage and Knapman, 2003; Di Giulio et al., 2018; Moyen et al., 2008). Electronic prescribing or computerized physician order entry (COPE) is one of the methods used to reduce the rate of medication errors by eliminating the risk factors associated with handwritten prescriptions as demonstrated by the study of (Kenawy and Kett, 2019).

The swift surge in unstructured textual data within the medical realm specifically for prescription and dosage data has underscored the importance of natural language processing (NLP) in extracting valuable insights from Electronic Health Records (EHRs), recognizing this approach as a pivotal tool in clinical research. The emergence of Large Language Models (LLMs) tailored for biomedical and clinical text mining has significantly bolstered NLP capabilities in this domain.

Recently, Large Language Models (LLMs) have shown remarkable capabilities in various medical domains, including disease diagnosis and prediction using Electronic Medical Records (EMRs) (Ayers et al., 2023; Lee et al., 2023; Nayak et al., 2023). Several research efforts have explored different methodologies to enhance LLM performance. For instance, (Zhu et al., 2024) combined clinical notes and multivariate time-series data by integrating LLMs with Retrieval-Augmented Generation (RAG) technology, leveraging an adaptive multimodal fusion network. Most studies have focused on English-language EMR datasets, such as MIMIC-III. In another approach, (Xu et al., 2025) transformed diverse knowledge sources into text format and applied a retrieval-enhanced and consistency-regularized co-training method, known as DR.KNOWS.

Gao et al. (2023) enhanced diagnostic accuracy and interpretability by merging a knowledge graph built using the Unified Medical Language System (UMLS) with a clinical diagnostic reasoning-based graph model, called REALM. (Jiang et al., 2023) utilized LLMs and biomedical knowledge graphs to create patient-specific knowledge graphs, which were then processed using a Bidirectional Attention-Enhanced Graph Neural Network (BAT GNN), known as RAM-HER. (Johnson et al., 2016) focused primarily on ICU data, which may not be adequate for modeling other cases.

A study by (Mishra and Shridevi, 2024) on knowledge graph-driven medicine recommendation systems using graph neural networks on longitudinal medical records confirmed the effectiveness

of the admission-wise clinical and medicine Knowledge Graphs and their application to the real world. However, despite these strides, the application of LLMs and Knowledge Graphs in clinical research remains constrained (Wang et al., 2019). Large Language Models (LLMs) represent advanced artificial intelligence systems engineered for the comprehension and generation of human language. Constructed upon deep learning methodologies, these models undergo extensive training on massive datasets, often reaching billions of words. Employing neural networks with multiple layers, they adeptly grasp intricate language patterns and structures, showcasing impressive zero-shot generalization capabilities (Jiang et al., 2023). However, there is limited research on the Kenyan EMR datasets.

Bernstein et al. (2023) highlight that while large language models (LLMs) have significant potential in medicine, their direct application raises concerns about inaccuracies and hallucinations due to limited specialized medical knowledge.(Achiam et al., 2023) note that training LLMs for medical use requires extensive high-quality data, but many top-performing models are closed-source, making further training challenging. Additionally, (Baek et al., 2023)emphasize that because medical knowledge evolves rapidly, updating trained LLMs requires retraining, a process that is both time-intensive and costly.

## 1.2 Statement of the Problem

Clinical data concerning medications holds utmost significance within electronic medical records, playing a critical role in ensuring healthcare safety and quality. Moreover, it is indispensable for clinical research utilizing electronic medical record data. Despite its crucial nature, medication information is frequently documented in clinical notes using free text, rendering it inaccessible to other computerized applications that rely on coded data (Kandaswamy et al., 2021). Over the years, there has been vast innovation in medical drugs, hence the generation of huge changing data on how to use these drugs based on various aspects. This leads to doctors and pharmacists being overwhelmed in keeping up with this important information when instead they should be focusing on detecting diseases in patients, finding appropriate cures, and prescribing the appropriate medical drugs. On the other hand, patients and other users of prescription

and dosage information struggle to get reliable sources for accurate prescription and dosage information which leads to medication errors(Kandaswamy et al., 2021). In the Kenyan context, there was a release of the Kenya National Medicine Formulary 1st edition repository. This repository contains a prescription guide for recommended medical drugs to be used in the Kenyan setting. The formulary aims to provide a reliable source of medical drug prescription data. However, the repository is in textual format and contains 6224 token splits in 546 pages. Retrieving information from the repository can be time-consuming and exhausting for a doctor or a chemist who wishes to give medication to a patient in a high-frequency medical center setting, considering the average time a doctor spends time with an outpatient is fifteen minutes. At the same time, it is confusing for a patient who wants to gather more information about the medication provided to them.

## 1.3 Research Objectives

### 1.3.1 General Objective

To develop a large language model (LLM)-powered chatbot that improves the retrieval of prescription information from the KNMF, thereby reducing medication errors and enhancing decision support for healthcare professionals while giving prescription.

### 1.3.2 Specific Objectives

i. To review existing literature to identify LLMs methodologies applied on medical errors in healthcare text records.

ii. To fine-tune the Meditron-7B model using PubMed prescription data and implement Retrieval Augmented Generation (RAG) for better context-aware responses.

iii. To evaluate the chatbot performance using ROUGE and BLEU metrics.

iv. To deploy a fully functional assistive chatbot on a web platform.

## 1.4 Research Questions

i. What are the challenges and limitations of the existing RAG model for the retrieval of medical drugs dosage information and how can these challenges be addressed for effectiveness in information retrieval?

ii. What are the essential components that should be integrated into chatbots to effectively automate the process of medical drugs dosage information?

iii. What criteria and metrics should be used to evaluate the performance and reliability of chatbots for automated retrieval of medical drugs dosage information?

iv. What are the efficient ways of deploying assistive chatbots in healthcare domain?

## 1.5 Scope of the Study

This study focused on developing and evaluating the Chatbot within Kenyan healthcare settings, specifically targeting independent pharmacies and medical centers without access to EHR systems. The study's unit of analysis was the retrieved medical drugs dosage information by Chatbots and traditional methods. The study compared the quality, consistency, and efficiency of the retrieval process by Chatbots to those retrieved by traditional methods.

## 1.6 Significance of the Study

The significance of this study is multifaceted, addressing critical challenges in medication management and patient safety within healthcare systems, particularly in middle-income countries like Kenya. Medication errors (MEs) are a prominent issue in healthcare, contributing to significant morbidity and mortality rates globally (World Health Organization, 2017).

In Kenya, healthcare providers frequently encounter challenges in accessing accurate and timely medication information due to the cumbersome nature of the Kenya National Medicine Formulary (KNMF), which is presented in a lengthy text-based format. This difficulty can lead

to critical errors during the prescription and administration phases of medication management, where the stakes are particularly high. By developing a large language model (LLM)-powered Chatbot that utilizes a Retrieval Augmented Generation (RAG) process, this study sought to enhance the efficiency and accessibility of medication-related data, thereby reducing the incidence of MEs.

Furthermore, this research is justified by the potential of advanced natural language processing (NLP) techniques to transform how healthcare professionals interact with medical data. The integration of LLMs has demonstrated significant improvements in various applications within healthcare, including data extraction from clinical records and enhancing decision support systems. By fine-tuning the Meditron-7B model on a substantial dataset of labeled prescription data from PubMed, this study aims to leverage the model's capabilities to provide accurate and contextually relevant information. The anticipated outcomes include improved decision-making processes for healthcare professionals, enabling them to make more informed prescriptions based on reliable data. This is particularly vital in environments where healthcare providers face time constraints and need quick access to essential medication information.

Additionally, the study addresses a critical gap in the availability of reliable medication information for independent pharmacies and patients who lack access to Electronic Health Records (EHR) systems. The KNMF serves as a vital resource for drug prescription data; however, its current format limits its usability in fast-paced clinical settings. By employing an LLM-powered Chatbot that can efficiently retrieve and present relevant information from the KNMF, this research aims to empower healthcare providers and patients alike. This initiative not only enhances the accessibility of medication information but also contributes to improving overall patient safety by minimizing the risks associated with medication errors.

This study also contributes to the broader discourse on leveraging artificial intelligence in healthcare. As advancements in LLMs continue to evolve, their application in addressing real-world challenges becomes increasingly relevant. By demonstrating how an LLM-powered Chatbot can enhance prescription information retrieval, this research sets a precedent for future innovations aimed at improving healthcare delivery systems. The findings may inspire further

exploration into AI-driven solutions that enhance patient safety and operational efficiency across different medical contexts.

Ultimately, this study aims to foster a safer healthcare environment through improved access to accurate medication information while contributing valuable insights into the integration of technology within clinical practice.

## 1.7 Limitations of the Study

Limitations include potential resource constraints during model training, such as limited GPU availability, which may affect the extent of fine-tuning possible with larger datasets. Large Language Models (LLMs) are susceptible to hallucination instances where the model generates information that is inaccurate or irrelevant to the input. Retrieval-augmented generation (RAG) models have proven valuable in mitigating this issue. Integrating RAG ensured that the model's responses are anchored in factual, external knowledge, thereby reducing hallucinations and improving accuracy.

# Chapter 2
# Literature Review

## 2.1 Introduction

This Chapter covers the past literature related to improving the medical drug prescription process with LLMs and knowledge graphs. It covers the theoretical and empirical literature, followed by the conceptual framework and research gap. The theoretical review discusses the theories and models upon which the study is anchored that have been used by other researchers, while the empirical literature looks into related studies in this area of knowledge. The conceptual framework discusses the model used and the structure from data collection to data visualization and generation. This literature review highlights the critical nature of addressing medical errors within healthcare systems through improved communication strategies and technological innovations such as Large Language Models (LLMs). It underscores the significant impact that medication errors have on patient safety and healthcare costs while emphasizing the potential role of LLMs in enhancing prescription information retrieval processes.

## 2.2 Theoretical Review

The study's theoretical review is based on theories and models that try to establish the existing knowledge and ways of improving the medical drug prescription process with LLMs and Knowledge Graphs.

### 2.2.1 Large Language Models

Large Language Models (LLMs) have evolved through contributions from many individuals and organizations. Joseph Weizenbaum developed ELIZA, one of the first language models, in 1966 at MIT (Shrager, 2024). Frank Rosenblatt created the Mark 1 Perceptron, the first artificial neural network, in 1958. In the 1980s, IBM developed small language models that predicted the next word in a sentence. Open AI introduced the GPT (Generative Pre-trained

Transformer) model in 2018, and Marco Muselli developed the Logic Learning Machine (LLM), a machine learning method that generates intelligible rules. These milestones have collectively shaped the development of modern LLMs. Models such as GPT, BERT, and T5 help understand and generate natural language that aids in processing medical text data such as prescription guidelines, drug interactions, and patient records. This thus enhances decision-making by suggesting appropriate prescriptions based on historical data (Thirunavukarasu et al., 2023).

However, Large Language Models (LLMs) have been criticized for lacking domain-specific knowledge, dependence on training data that may be biased or outdated, and a lack of transparency in decision-making. They can struggle with interpreting complex medical contexts, are prone to overfitting, and may perpetuate biases present in the training data. Additionally, LLMs cannot independently verify information (Chang et al., 2024; Fan et al., 2024). Large Language Models (LLMs) are highly relevant to a study as they can enhance decision-making by processing and interpreting vast amounts of medical literature, clinical guidelines, and patient data. LLMs, like GPT models, can be used to extract key insights from medical texts, identify relevant research studies, and generate context-aware recommendations for drug prescriptions. By analyzing unstructured data, such as clinical notes and treatment guidelines, LLMs can help healthcare providers quickly access the latest evidence, ensuring that prescriptions align with current medical knowledge.

### 2.2.2 Natural Language Processing (NLP) Models

Natural Language Processing (NLP) is a subfield of artificial intelligence and computer science focused on enabling computers to process and understand data encoded in human language. The models are closely related to information retrieval, knowledge representation, and computational linguistics, a subfield of linguistics. It encompasses various tasks such as speech recognition, text classification, and natural-language generation (Li et al., 2022). NLP has evolved from symbolic approaches in the 1950s, which relied on rule-based systems, to statistical methods in the 1990s, which utilized machine learning and corpora-based models. In the 2010s, deep learning methods, especially neural networks, advanced NLP, achieving state-of-the-art results in tasks like language modelling and machine translation. Key approaches in NLP include

symbolic, statistical, and neural network methods, with the latter becoming dominant due to its ability to handle large datasets and adapt to unfamiliar or erroneous input. Currently, NLP has applications in healthcare, where it helps analyze text in electronic health records (Hossain et al., 2023). However, NLP models are criticized for being so dependent on big amounts of data for training, therefore if the training data is small, or low quality or if the data is unbalanced the performance of the model degrades considerably (Chowdhary, 2020). Since the NLP models typically perform best with high-resource languages, the available low-resource language data for training might be sparse, leading to reduced performance or even failure to work effectively. Some NLP models such as large transformer-based architectures, are computationally intensive, requiring substantial computational resources for both training and inference (Kang et al., 2020). The NLP models are relevant to this study since they provide the computers with the ability to extract key information from unstructured medical texts (such as; prescription guidelines and clinical notes) to assist in accurate prescriptions.

### 2.2.3 Evidence-Based Medicine (EBM) Model

The Evidence-Based Medicine (EBM) model was introduced in 1981 by clinicians such as David Sackett, Gordon Guyatt, and Archie Cochrane. The model emphasizes using the best available clinical evidence to guide medical decisions. Large Language Models (LLMs) and Knowledge Graphs can significantly enhance the drug prescription process. The model proposes that LLMs can process vast medical literature, extracting relevant guidelines and treatment protocols. Knowledge Graphs structure relationships between drugs, diseases, and patient characteristics to optimize personalized prescriptions (Charles et al., 2011) . By integrating these technologies, the prescription process can be made more efficient, evidence-driven, and tailored to individual patients, ensuring that EBM principles are applied more effectively in clinical practice. However, the Evidence-Based Medicine (EBM) model has limitations, including that it can over-rely on quantitative data while underestimating the importance of clinical judgment. In addition, biases in research, fragmentation of evidence, and challenges in staying up-to-date can hinder its effectiveness in guiding medical prescriptions (Gomory, 2013). The model is relevant to this study for its emphasis on the integration of the best available clinical evidence

in decision-making. LLMs can assist in synthesizing and analyzing large volumes of medical literature, while knowledge graphs can structure complex relationships between diseases, drugs, and treatment outcomes. Together, these technologies can enhance the accuracy and efficiency of evidence-based drug prescriptions, ensuring that clinical decisions are well-informed and aligned with the latest research and guidelines.

### 2.2.4 Medical Errors in Healthcare

Medical errors (MEs) represent a significant challenge to patient safety and healthcare quality, defined as preventable adverse effects of medical care that can occur at any stage of patient care, including diagnosis, treatment, and medication administration (Kohn et al., 2000). A systematic review indicated that approximately 400,000 hospitalized patients in the United States experience preventable harm each year, with over 200,000 deaths attributed to these errors (Rodziewicz et al., 2022). The financial implications are substantial, with estimates suggesting that medical errors cost the healthcare system between 20 billion and 45 billion annually due to complications such as hospital-acquired infections (Rodziewicz et al., 2022). This burden not only affects patients but also has profound psychological impacts on healthcare providers who may experience feelings of guilt and inadequacy following an error (Weingart et al., 2005). The types of medical errors can be broadly categorized into several domains, including medication errors, surgical errors, diagnostic failures, and equipment-related issues. Medication errors are particularly prevalent, accounting for a significant proportion of adverse events in hospitals. A systematic review and meta-analysis found that approximately 54% of nurses in Iran have encountered medication administration errors, with the most common types being incorrect timing (27.3%) and incorrect dosage (26.4%). The primary contributing factors were workload (43%), fatigue (42.7%), and nursing shortages (38.8%) (Heidari et al., 2024). Furthermore, a review of 40 studies on medical errors in Iran reported that nursing staff and nursing students were the most frequently involved in these incidents, with medication errors being the most reported type. The prevalence of these errors ranged from 10% to 80%, with university or teaching hospitals being the most frequent settings observed (Jafarian et al., 2019). This highlights the need for targeted interventions aimed at reducing specific types of errors

through improved communication strategies and enhanced training for healthcare providers. Preventing medical errors requires a multifaceted approach that includes identifying contributing factors and implementing systemic changes within healthcare organizations. Strategies such as adopting a culture of safety, encouraging error reporting without fear of retribution, and utilizing technology to streamline processes have been shown to significantly reduce the incidence of medical errors (Rai et al., 2021). Additionally, educational programs aimed at increasing awareness among healthcare professionals about the types and causes of medical errors can further enhance patient safety. By fostering an environment that prioritizes patient safety and encourages continuous learning and improvement, healthcare institutions can mitigate the risks associated with medical errors and ultimately improve patient outcomes (Weingart et al., 2005). Recent advancements in Large Language Models (LLMs) have showcased their remarkable capabilities in natural language processing tasks. A comprehensive review of the literature on LLMs reveals a broad spectrum of concepts encompassing foundational principles to cutting-edge topics. Systematic surveys have been conducted to provide insights that advance the field, highlighting key aspects such as the chronological development of LLMs, significant findings from prominent studies, and detailed discussions on their design and architecture (Minaee et al., 2024).

## 2.3 Empirical Literature review

### 2.3.1 Applications of LLMs and Knowledge Graphs in Healthcare

One notable application of Large Language Models (LLMs) is in the healthcare sector, particularly for extracting free-text data within clinical research. The integration of models like ChatGPT has demonstrated a transformative impact on data extraction from discharge medication forms. Studies indicate a 100% completeness rate when data is directly extracted from the doctor's order system, effectively eliminating uncertainties regarding drug usage (Ntinopoulos et al., 2025). However, challenges remain with Electronic Health Records (EHRs), which are primarily designed for clinical rather than research purposes (Shah et al., 2023) The efficiency

gains from LLMs are significant, with an 80% improvement in AI-assisted data transcription time, underscoring their scalability and cost-effectiveness in retrospective research projects (Zhang et al., 2024). Furthermore, optimized LLMs maintain high accuracy levels, ensuring that efficiency improvements do not compromise data quality. The adoption of advanced LLMs also enhances data extraction security and efficiency, while fostering better collaboration between technical experts and healthcare professionals (Shah et al., 2023).

Mishra and Shridevi (2024) reviewed the Knowledge graph-driven medicine recommendation system using graph neural networks on longitudinal medical records. The study proposed a novel Knowledge Graph-Driven Medicine Recommendation System using Graph Neural Networks, KGDNet, that utilizes longitudinal EHR data along with ontologies and Drug-Drug Interaction knowledge to construct admission-wise clinical and medicine Knowledge Graphs for every patient. The study employed Recurrent Neural Networks to model a patient's historical data, and Graph Neural Networks to learn embeddings from the Knowledge Graphs. A Transformer-based Attention mechanism was also employed to generate medication recommendations for the patient, considering their current clinical state, medication history, and joint medical records. The model was evaluated on the MIMIC-IV EHR data and outperforms existing methods in terms of precision, recall, F1 score, Jaccard score, and Drug-Drug Interaction control. The effectiveness of the system and its application to the real world was demonstrated by the study.

Gao et al. (2023) explored integrating a medical Knowledge Graph (KG) with Large Language Models (LLMs) to improve automated diagnosis prediction and reduce diagnostic errors. Their approach used the National Library of Medicine's Unified Medical Language System (UMLS) to create a robust KG, supporting a novel graph model, Dr.KNOWs, that aids in interpreting complex medical concepts. The study showed that the integration enhanced the accuracy of diagnosis predictions, outperforming traditional concept extractors and boosting performance in diagnosis prediction tasks, with improved ROUGE scores for LLMs. Human evaluations confirmed that the model incorporating the KG generated more explainable reasoning aligned with human reasoning. The findings emphasize the potential of combining external medical knowledge with LLMs for advancing AI-driven diagnostic decision support systems.

### 2.3.2 Innovative Frameworks Using LLMs

The innovative integration of classical planning techniques into LLMs is exemplified by the LLM+P framework. This framework processes natural language descriptions of planning issues and generates correct or optimal plans by converting these descriptions into Planning Domain Definition Language (PDDL) files. Classical planners are then utilized to efficiently identify solutions which are subsequently translated back into natural language. Experiments on benchmark problems indicate that LLM+P consistently provides optimal solutions, outperforming standard LLMs that often struggle with feasible plan generation (Si et al., 2024)

### 2.3.3 Broader Application of LLMs

In the broader context of healthcare, LLMs like GPT-3.5 have demonstrated significant potential across various applications including medical dialogue summarization, patient-trial matching, and computer-aided diagnosis (CAD) systems in medical imaging. These models generate high-quality text that is empathetic while simplifying clinical documents into patient-friendly language. However, challenges remain such as understanding visual information and the technical language used in clinical notes. Despite these challenges, LLMs hold substantial practical potential in complementing traditional healthcare systems and enhancing the quality of care (Huang et al., 2024; Tian et al., 2024; Webster, 2023).

Xu et al. (2025) presents a novel framework that integrates Large Language Models (LLMs) with Knowledge Graphs (KG) to address the growing need for high-quality knowledge in medical question-answering systems. By leveraging the triplet data structure of the KG, this approach enhances the medical knowledge base of LLMs, improving their explanatory power. The method aligns LLM outputs with relevant KG information, enabling dual verification that boosts accuracy, consistency, security, and reliability in medical question answering. Experimental results show that this integrated approach outperforms traditional knowledge-based question-answering (KBQA) systems and standalone LLM methods, offering a more efficient and accurate solution for medical knowledge services. This study highlights the potential of combining LLMs and KGs in advancing medical text mining and knowledge extraction.

## 2.4 Research Gaps

Despite advancements in understanding medical errors and leveraging large language models (LLMs) for improved healthcare outcomes, several gaps remain in the literature that align specifically with this study's objectives. First, while numerous studies highlight the prevalence of medication errors across various settings (Agbor, 2016), there is limited research specifically addressing how LLMs can be systematically integrated into existing workflows to mitigate these errors effectively. The literature lacks empirical evidence demonstrating how chatbot interfaces powered by LLMs can directly influence decision-making processes among healthcare providers during critical phases such as prescription writing.

Additionally, while existing literature emphasizes the importance of communication strategies among healthcare providers to reduce medication errors (Agbor, 2016), there is an insufficient exploration into how LLMs can facilitate this communication through real-time data retrieval and decision support systems tailored for specific clinical contexts. Furthermore, recent studies demonstrate promising results regarding the efficiency gains from using LLMs for data extraction (Huang et al., 2024); however, there is a need for more comprehensive evaluations assessing long-term outcomes related to patient safety and satisfaction when integrating these technologies into routine practice.

Another gap pertains to the evaluation metrics used to assess LLM performance in clinical settings. Most studies focus on traditional metrics such as accuracy or F1 scores without considering user-centric measures like usability or satisfaction levels among healthcare professionals using these systems. Despite promising advancements in this field, specific research gaps remain regarding practical implementations of LLM-powered solutions in clinical workflows. Addressing these gaps is essential for optimizing patient care outcomes and ensuring that innovative technologies effectively support healthcare professionals in reducing medication-related adverse events.

## 2.5 Conceptual Framework

The conceptual framework outlines an innovative system for prescription assistance in healthcare, leveraging the capabilities of fine-tuned large language models (LLMs) and modern cloud-based infrastructure. The framework addresses the growing need for precise, context-aware dosage recommendations while ensuring seamless interaction between healthcare providers and advanced computational models. This system is built to support doctors with accurate drug and dosage information by integrating structured medical knowledge with cutting-edge machine learning models. The proposed workflow ensures scalability, reliability, and security through cloud deployment and offers enhanced decision-making support via an intuitive prescription interface.

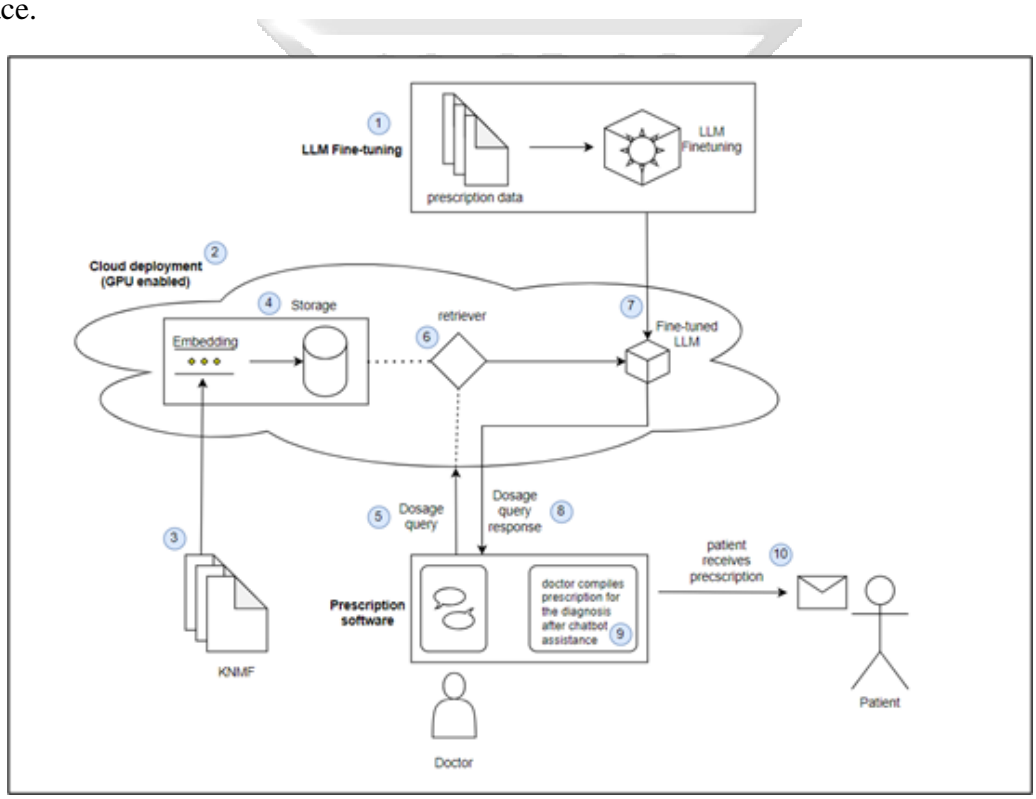



Figure 2.1: Conceptual Framework

### 2.5.1 LLM Fine-Tuning

Fine-tuning a large language model (LLM) involves adapting a pre-trained model to a specific domain using domain-specific datasets. In this case, a dataset derived from prescription

records, drug formularies, and clinical guidelines is used to refine the LLM's understanding of medical prescriptions. This step enhances the LLM's ability to provide context-sensitive and patient-specific responses. The fine-tuning process ensures that the model learns not just generic language patterns but also intricate medical terminology, dosage calculations, and drug interaction rules. The LLM, thus fine-tuned, serves as the backbone of the system, capable of providing precise and actionable recommendations.

### 2.5.2 Cloud Deployment (GPU Enabled)

The fine-tuned LLM and its supporting components are deployed on a cloud-based infrastructure equipped with GPU acceleration. This setup ensures high computational efficiency and supports real-time inference for complex queries. The cloud deployment allows for scalability, enabling the system to handle a growing number of queries without compromising performance. Moreover, it ensures secure access to the system while maintaining compliance with healthcare data protection regulations, such as HIPAA or GDPR, depending on the jurisdiction.

### 2.5.3 Knowledge Embedding (KNMF Data Processing)

The Kenyan National Medical Formulary (KNMF) serves as a critical knowledge base for the system. This medical data is processed and transformed into numerical embeddings using advanced natural language processing techniques. Each piece of knowledge—such as drug indications, contraindications, and dosage guidelines—is embedded into a high-dimensional vector space. This enables the system to efficiently match dosage queries with the most relevant medical information. The embeddings capture both semantic meaning and relational knowledge, ensuring high accuracy in retrieval tasks.

### 2.5.4 Storage (Vector Database for Knowledge Retrieval)

The embedded medical knowledge is stored in a specialized vector database. This type of database is optimized for similarity searches, allowing the system to quickly retrieve relevant

data based on input queries. Unlike traditional relational databases, vector databases support high-speed, approximate nearest-neighbor searches, which are essential for responding to complex, unstructured queries. The storage is designed to ensure data integrity, redundancy, and rapid accessibility, forming the core of the system's knowledge retrieval capabilities.

### 2.5.5 Dosage Query from Prescription Software

The prescription software provides an intuitive interface for healthcare professionals to interact with the system. Through this interface, doctors can input patient-specific details and pose queries related to drug dosages, contraindications, or interactions. The query is then transmitted securely to the backend system for processing. This software serves as a bridge between the doctor's clinical expertise and the system's computational intelligence, ensuring that the interaction is user-friendly and aligned with clinical workflows.

### 2.5.6 Retriever Mechanism

The retriever mechanism forms the intermediary layer between the database and the fine-tuned LLM. Upon receiving a query, the retriever searches the vector database to identify and extract the most relevant pieces of information. Advanced retrieval techniques, such as dense vector retrieval and cosine similarity, are employed to ensure the highest precision. This step ensures that the LLM only processes the most contextually relevant data, reducing computational overhead and improving response accuracy.

### 2.5.7 Fine-Tuned LLM for Query Resolution

The fine-tuned LLM takes the retrieved information and combines it with the input query to generate a response. The model is trained to consider both the explicit knowledge retrieved and its internal understanding of medical contexts. By synthesizing these two sources of information, the LLM produces responses that are contextually appropriate, medically accurate, and tailored

to the patient's condition. This dual-processing approach enhances the system's reliability and ensures that responses align with established medical guidelines.

### 2.5.8 Dosage Query Response

The response generated by the LLM is transmitted back to the prescription software. This response typically includes dosage recommendations, administration guidelines, and safety precautions. The system is designed to present this information in a clear, concise, and actionable format, reducing the cognitive load on healthcare professionals. The response is also structured to highlight critical details, such as potential drug interactions or patient-specific considerations, ensuring comprehensive support for clinical decision-making.

### 2.5.9 Doctor's Prescription Compilation

With the system's assistance, the doctor reviews the dosage recommendations and compiles the final prescription. The prescription software ensures that the doctor retains full control over the decision-making process, offering recommendations as guidance rather than mandates. This collaborative approach balances the strengths of computational intelligence with the doctor's clinical expertise, ensuring that the patient receives safe and effective care.

### 2.5.10 Prescription Delivery to the Patient

The finalized prescription is delivered to the patient in a secure and accessible format. Depending on the system's implementation, this could include digital delivery via email or patient portals, as well as printed prescriptions. The delivery process ensures that patients receive clear and actionable instructions for their medications, supporting adherence and improving health outcomes.

# Chapter 3
# Methodology

## 3.1 Introduction

This chapter outlines the research methodology that was used to improve the medical drug prescription process with LLMs and knowledge graphs. The methodology is guided by the Cross Industry Standard Process for Data Mining (CRISP-DM) framework. The CRISP-DM framework provides a structured approach to data mining and knowledge discovery, making it an ideal methodology for integrating LLMs and knowledge graphs into the medical prescription process. The proposed LLM for this study was Meditron 7B, which is fine-tuned from Llama 2. This chapter details each phase of the CRISP-DM process, from understanding the problem and preparing data to modelling and evaluating the solution.

## 3.2 Research Design

This study employed a design science research (DSR) methodology to iteratively design and evaluate a system that integrates LLMs and knowledge graphs for medical prescriptions. Design Science Research is ideal because it emphasizes the creation and evaluation of innovative artifacts, such as algorithms, models, or systems as illustrated in figure 3.1. This design allows researchers to focus on solving a practical problem which is improving the prescription process through developing and testing technology-driven solutions (Vom Brocke et al., 2020). The DSR approach also enabled: the development and implementation of the LLMs and knowledge graph system; testing the system's effectiveness and usability in real or simulated clinical settings; and continuously improving the system based on data-driven insights (Sultana et al., 2010)

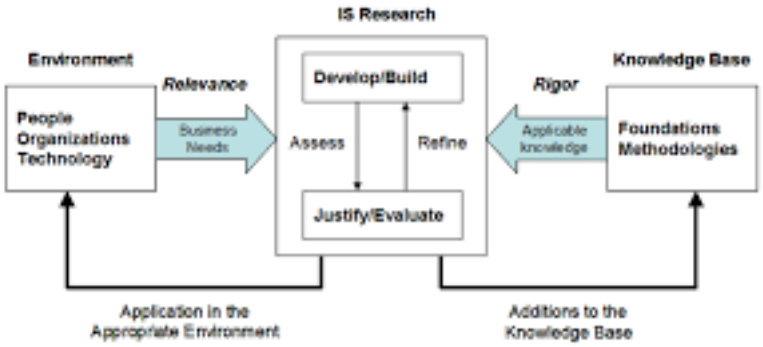



Figure 3.1: Design Science Research

## 3.3 Data Sources

The data sources for this study were primarily comprised of clinical and pharmaceutical datasets that provide detailed information on drug prescriptions. Prescription data was sourced from the PubMed Central Open Access Subset via the BioASQ platform. These datasets consisted of historical question-and-answer prescription records and publicly available pharmaceutical knowledge repositories such as drug formularies and clinical guidelines. Additionally, the study incorporated data from the Kenya National Medicines Formulary (KNMF), which provides comprehensive information on essential medicines, their indications, dosages, contraindications, and adverse effects. The inclusion of KNMF data enhanced the system's ability to align with local prescribing guidelines, ensuring that recommendations generated by the model adhere to the standards set by the Kenyan healthcare system.

These data sources were crucial in fine-tuning the Large Language Model (LLM) and constructing the knowledge graph, enabling the system to understand complex medical relationships, recommend appropriate drugs, and assist healthcare professionals in making informed decisions. The integration of multiple high-quality data sources improved the model's accuracy and reliability in prescription validation and error detection.

The data used in this study was anonymized to comply with the *Data Protection Act, 2019*, which ensures the protection of personal and sensitive health information. The study also adhered

to regulations set by the *Kenya Medical Practitioners and Dentists Council (KMPDC)* and other relevant bodies to safeguard patient confidentiality and data security. By following these legal and ethical standards, the research maintained privacy and aligned with both local and international best practices.

## 3.4 CRISP-DM Framework

The research was guided by the Cross-Industry Standard Process for Data Mining (CRISP-DM) framework, which is widely recognized for structuring data-driven projects Chapman et al. (2000). The framework's six phases; business understanding, data understanding, data preprocessing, modeling, evaluation, and deployment—were adapted to the unique requirements of developing a large language model (LLM)-powered chatbot tailored for prescription information retrieval from the KNMF Provost and Fawcett (2013). This has been illustrated in figure 3.2

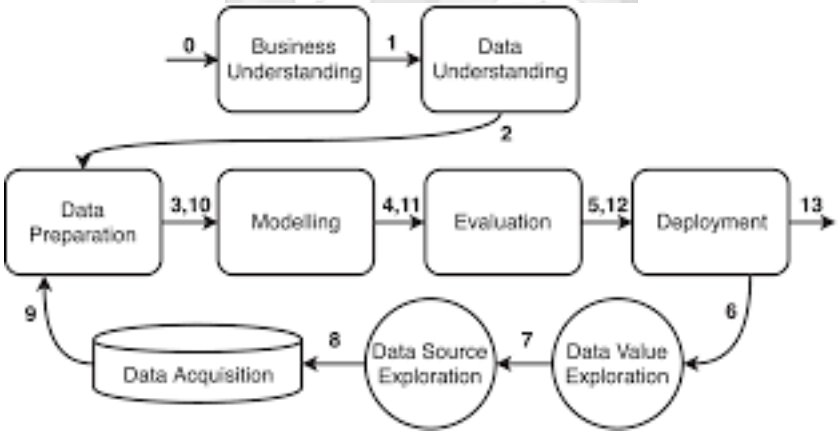



Figure 3.2: CRISP DM Framework

### 3.4.1 Business Understanding

The complexity of pharmaceutical data and the necessity for accurate, real-time access to prescription information present significant challenges in healthcare. The Kenya National Medicine Formulary serves healthcare professionals through its provision of necessary drug information. Medical practitioners encounter significant difficulties in efficient prescription

information access because their time is limited and KNMF data is unstructured and incompatible with current healthcare systems. Such barriers produce medication errors together with delayed treatment and inferior patient results.

The research develops a prescription information retrieval system from KNMF using a Large Language Model (LLM)-based chatbot to enhance efficiency. Users can interact with the chatbot through Natural Language Processing (NLP) and machine learning technology which enables it to provide exact responses to queries. The implemented AI solutions aim to create a link that connects sophisticated medical vocabulary to basic user interfaces for essential drug information retrieval.

The main purpose behind this research project involves enhancing healthcare services through quick access to dependable prescription information tools for providers. The proposed system will improve clinical decision-making processes by decreasing medication errors while optimizing workflows. The chatbot system ensures national drug regulations compliance through its delivery of recent pharmaceutical guidelines which follow standardized practices.

### 3.4.2 Data Understanding

In the data understanding phase, the study focused on defining the variables of interest and the sources of data. The primary variable in this research is the accuracy of prescription information retrieval, which was assessed through various performance metrics. The dataset comprises of labelled prescription data sourced from the PubMed Central Open Access Subset via the BioASQ platform. The training data was in three datasets. Train set with 8549 rows, validation set with 8400 rows and test set with 8598 rows. This dataset includes structured text entries relevant to medication prescriptions, ensuring a robust foundation for fine-tuning the Meditron-7B model. Additionally, localization data from the KNMF was utilized as the use case for the retrieval and evaluation process, which consists of 6224 token splits across 547 pages in PDF format. The data contains recommended medical drugs to be used in Kenya. It includes all details of the drugs including dosage information, side effects, administration guidelines to different types of patients. The KNMF data is open for use to the public and is accessible through the MoH portal.

### 3.4.3 Data Pre-processing

Token splitting is performed to convert text into manageable units for embedding processes. Contextual embedding was then be applied to capture word meanings based on their surrounding context (Devlin et al., 2019). Token splitting in this context involves dividing text into smaller units or tokens that can be words, sub-words, or characters. This process is critical for creating embedding, as each token is mapped to a unique vector representation that captures its semantic meaning. This allows models to maintain a balance between vocabulary size and the ability to represent diverse language inputs. The resulting embedding serves as inputs for neural networks, enabling the models to understand and generate human-like text by leveraging the relationships and contexts between tokens. Contextual embedding focuses on providing a more dynamic representation that reflects specific usage in different contexts. Unlike traditional embedding that assigns a fixed vector to each word, contextual embedding can vary based on surrounding words. This capability is particularly important in tasks such as text generation and machine translation, where understanding context is key to producing relevant and coherent outputs.

### 3.4.4 Machine Learning Modeling using Transfer Learning

This study leveraged transfer learning. Transfer learning is a technique used in the context of fine-tuning Large Language Models (LLMs) to adapt pre-trained models to specific tasks or domains by leveraging the knowledge gained during pre-training on large, general datasets and applying it to a new, more specialized task, which involves starting with a pre-trained LLM, further training the model on a new, task-specific dataset with some or all of the pre-trained layers set to be updatable to adjust its weights to the new task while retaining its general language knowledge, significantly reducing the computational cost and training time required to achieve good performance on a new task compared to training from scratch, leading to better performance compared to directly applying the pre-trained model to the new task without further training (zero-shot learning), allowing for deep customization of algorithms for domain-specific tasks, injecting domain expertise, reducing computational power and training time compared to training from scratch, enhancing prediction accuracy by using pre-collected extensive datasets

and domain expertise, and enabling rapid adaptation to new conditions and tasks.(Devlin et al., 2019; Gururangan et al., 2020)

Meditron LLM fine tuned from Meta LLaMa was selected for this project. The model was selected since it has been fine- tuned with medical data. Meditron is also open source and tunable to medical specific tasks due to it's high adaptability.

### The LLaMA LLM

LLaMA (Large Language Model Meta AI) is a family of autoregressive large language models developed by Meta AI, with the latest version, LLaMA 3, released on April 18, 2024 (Touvron et al., 2023). LLaMA models are designed to be open-source and are available for both research and commercial use, providing transparency regarding their architecture and training data. The architecture of LLaMA models typically consists of transformer-based neural networks, which are trained on large datasets to predict the next word in a sequence, allowing them to generate coherent and contextually relevant text.

The models come in various sizes, with LLaMA 3 offering configurations of 8 billion and 70 billion parameters, trained on approximately 15 trillion tokens. This extensive training enables LLaMA to perform well across diverse tasks and languages, making it adaptable for applications in different domains. LLaMA's architecture also emphasizes efficiency, achieving remarkable compute utilization during training, which is facilitated by Meta's custom-built GPU clusters.

The core of the LLaMA architecture follows the transformer-based autoregressive modeling approach. Each layer of the transformer network consists of a self-attention mechanism and a feed-forward network, mathematically expressed as:

$$X_{\text{out}} = \text{LayerNorm}\left(\text{MultiHeadAttention}(X_{\text{in}}) + X_{\text{in}}\right) + \text{LayerNorm}\left(\text{FeedForward}(X_{\text{in}})\right) \quad (3.1)$$

where: - $X_{\text{in}}$ is the input token representation, - MultiHeadAttention$(\cdot)$ represents the self-attention mechanism, - FeedForward$(\cdot)$ is a position-wise fully connected feed-forward network, and - LayerNorm$(\cdot)$ denotes layer normalization for stabilizing training.

LLaMA models also incorporate rotary positional embeddings (RoPE) instead of absolute positional encodings to enhance long-context learning. The attention mechanism in LLaMA is formulated as:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3.2}$$

where: - $Q,K,V$ are the query, key, and value matrices, respectively, - $d_k$ is the dimensionality of the keys, and - softmax$(\cdot)$ ensures normalized attention scores.

These architectural innovations contribute to LLaMA's ability to efficiently handle long-range dependencies while maintaining high computational efficiency.

**Meditron LLM**

Meditron is a suite of open-source large language models (LLMs) developed by researchers at EPFL's School of Computer and Communication Sciences and Yale School of Medicine, in collaboration with the International Committee of the Red Cross (ICRC)

Built on Meta's LLaMA 2 model, Meditron is fine-tuned on carefully curated, high-quality medical data sources with input from clinicians and humanitarian experts, aiming to assist with clinical decision-making and diagnosis, particularly in low-resource medical settings that can benefit most from such technologies. The suite includes two versions: a 7 billion parameter model (Meditron-7B) and a 70 billion parameter model (Meditron-70B), both of which outperform LLaMA 2 and other models on medical benchmarks like MedQA and MedMCQA. Designed to provide life-saving advice and guidance, Meditron compresses complex medical information into an accessible conversational interface.

Meditron was fine-tuned from LLaMA by utilizing a process known as continued pretraining on a carefully curated medical corpus. Specifically, Meditron-7B and Meditron-70B models

were adapted from LLaMA-2-7B and LLaMA-2-70B, respectively. The fine-tuning involved training on high-quality medical data sources, including selected PubMed articles, medical guidelines, and general domain data from RedPajama-v1. This approach allowed Meditron to encode medical knowledge effectively while minimizing contamination and bias from the general text corpus used in LLaMA's initial training. The fine-tuning process was supported by continual input from clinicians and humanitarian experts, ensuring that the model's outputs align with evidence-based practices and professional standards.

### GPT-3.5 LLM

GPT-3.5 was used for benchmarking the performance of the fine-tuned Meditron LLM. GPT-3.5, developed by OpenAI, is a transformer-based autoregressive language model designed to generate coherent and contextually accurate text by predicting the next token in a sequence. It builds upon the architecture of its predecessor, GPT-3, introducing several enhancements in training techniques and computational efficiency (Brown et al., 2020b). The model leverages a dense attention mechanism and is trained on a mixture of public and licensed datasets to provide a robust understanding of diverse contexts and languages.

With approximately 175 billion parameters, GPT-3.5 utilizes layer normalization and a modified activation function to improve gradient flow during training, enhancing its scalability and performance on downstream tasks (Radford et al., 2019). The training was conducted on supercomputing infrastructure with mixed precision, optimizing both memory and computational resource usage. To fine-tune GPT-3.5 for specific domains, OpenAI incorporates reinforcement learning from human feedback (RLHF), enabling the model to generate outputs that align with user intent while maintaining ethical considerations (Ouyang et al., 2022b).

GPT-3.5 follows the standard transformer-based autoregressive architecture, where each transformer block consists of a multi-head self-attention mechanism and a position-wise feed-forward network. The output of each transformer block is computed as follows:

$$X_{\text{out}} = \text{LayerNorm}\left(\text{MultiHeadAttention}(X_{\text{in}}) + X_{\text{in}}\right) + \text{LayerNorm}\left(\text{FeedForward}(X_{\text{in}})\right) \quad (3.3)$$

where: - $X_{\text{in}}$ represents the input token embeddings, - MultiHeadAttention$(\cdot)$ represents the multi-head self-attention mechanism, - FeedForward$(\cdot)$ is a position-wise fully connected network, and - LayerNorm$(\cdot)$ normalizes activations for stability.

The multi-head attention mechanism is computed as:

$$\text{MultiHeadAttention}(Q,K,V) = \text{Concat}(\text{head}_1,\ldots,\text{head}_h)W^O \quad (3.4)$$

where each attention head is defined as:

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \quad (3.5)$$

and the scaled dot-product attention is computed as:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (3.6)$$

where: - $Q,K,V$ are the query, key, and value matrices, - $W_i^Q, W_i^K, W_i^V$ are learned parameter matrices, - $d_k$ is the dimension of each key vector, and - softmax$(\cdot)$ ensures proper attention weight distribution.

GPT-3.5 also employs rotary positional embeddings (RoPE) instead of absolute positional encodings, which enhance long-context learning by encoding relative positions efficiently. The final token prediction follows an autoregressive approach:

$$P(y_t|y_{<t},X) = \text{softmax}(W^T X_t) \quad (3.7)$$

where $y_t$ is the predicted token at time step $t$, and $W$ is the learned weight matrix mapping hidden states to vocabulary logits.

These architectural refinements contribute to GPT-3.5's efficiency and adaptability for diverse NLP applications.

### Fine-tuning Meditron

The Meditron model was fine-tuned with the medical drugs prescription data to improve accuracy in information retrieval. The process was conducted on Google colabolatory environment, which offered access to GPUs, which are necessary for training large language models as they require High Performance Computing.

### Quantization Configuration

The quantization configuration that was employed for fine-tuning the Meditron LLM is designed to enhance memory efficiency and computational performance.(Zafrir et al., 2019)Utilizing the 'BitsAndBytesConfig' class, the model is set to load with 4-bit precision, significantly reducing its memory footprint while maintaining acceptable accuracy levels. The quantization type is specified as "nf4," which balances performance and precision effectively. The computation is performed using half-precision floating-point format ('torch.float16'), further optimizing computational efficiency on compatible hardware. Additionally, the configuration disables double quantization ('bnb_4bit_use_double_quant=False'), which, while potentially beneficial for performance, adds complexity. This quantization strategy facilitates the efficient fine-tuning of the Meditron model, enabling its deployment in resource-constrained environments without compromising its capabilities.(Bai et al., 2020)

### LoRa Configuration

The configuration for Low-Rank Adaptation (LoRA) (**?**) configured is designed to enhance the fine-tuning process of the model for causal language modeling tasks. The 'LoraConfig' parameters include 'lora_alpha', set to 16, which controls the scaling factor for the low-rank

updates, allowing for a balance between the adaptation and the original model weights. The ‘lora_dropout‘ parameter is set to 0.1, introducing a dropout rate that helps prevent overfitting by randomly setting a fraction of the LoRA parameters to zero during training. The rank parameter ‘r‘ is set to 64, determining the dimensionality of the low-rank matrices used for adaptation, which influences the capacity of the model to learn task-specific features while maintaining efficiency. The ‘bias‘ is specified as "none," indicating that no additional bias terms are added to the LoRA layers. Finally, the ‘task_type‘ is defined as "CAUSAL_LM," indicating that the model is being fine-tuned for causal language modeling tasks, where the objective is to predict the next token in a sequence based on previous tokens.(Dettmers et al., 2023) (Hu et al., 2021) This configuration collectively facilitates effective and efficient adaptation of the model to specific language tasks while minimizing the computational burden.

### 3.4.5 Retrieval-Augmented Generation (RAG)

RAG represents a notable advancement in the field of natural language processing by integrating information retrieval techniques with generative models. This innovative approach addresses a significant limitation of traditional generative models, which often generate plausible but factually incorrect responses. By incorporating relevant external information during the generation process, RAG effectively reduces the occurrence of hallucinations, thereby enhancing the reliability and richness of the generated content (Lewis et al., 2020). The synergy between retrieval and generation capabilities allows RAG to produce responses that are not only contextually appropriate but also based on the most current and accurate information available, positioning it as a crucial development in creating more intelligent and versatile language models(Lewis et al., 2020). This integration ensures that generative outputs are grounded in evidence, significantly improving their factual accuracy and contextual relevance(Lewis et al., 2020)(Lakatos et al., 2024).

### 3.4.6 Performance Evaluation

This research tested the performance of the Meditron base model, the fine-tuned version of the Meditron, and the GPT-3.5 model. We used ROUGE (Recall-Oriented Understudy for Gisting Evaluation), and BLEU (Bilingual Evaluation Understudy) to evaluate retrieval-based generation. Thereafter, the best model in the RAG metrics was further optimized.

#### ROUGE

ROUGE, or Recall-Oriented Understudy for Gisting Evaluation, is a set of metrics primarily designed to evaluate automatic summarization and machine translation by assessing the overlap between generated text and human-produced reference summaries. Key variants of ROUGE include ROUGE-N, which measures the overlap of n-grams, with ROUGE-1 focusing on unigrams (single words) and ROUGE-2 on bigrams (two-word combinations) Lin (2004). ROUGE-L evaluates the longest common subsequence (LCS) between generated and reference texts, providing insights into fluency by capturing the order of words Ganesan (2023). Additionally, ROUGE-W is a weighted version of ROUGE-L that emphasizes consecutive LCS matches to enhance sensitivity to coherent sequences Huang et al. (2021b).

$$\text{ROUGE-N} = \frac{\sum_{\text{ref} \in \text{References}} \sum_{\text{gram}_n \in \text{ref}} \text{Count}_{\text{match}}(\text{gram}_n)}{\sum_{\text{ref} \in \text{References}} \sum_{\text{gram}_n \in \text{ref}} \text{Count}(\text{gram}_n)} \tag{3.8}$$

$$\text{ROUGE-L} = \frac{\text{LCS}(\text{Generated}, \text{Reference})}{\text{Length}(\text{Reference})} \tag{3.9}$$

$$\text{ROUGE-W} = \frac{\sum_{i=1}^{k} w_l \cdot LCS_i}{\sum_{i=1}^{k} w_l \cdot \text{Length}(\text{Reference})} \tag{3.10}$$

The scores for these metrics range from 0 to 1, with higher values indicating greater similarity between the generated output and reference texts; this recall-oriented approach highlights how much important content from the reference is captured in the generated summary, making it particularly valuable in contexts where comprehensiveness is essential Chen et al. (2020).

Further, each ROUGE level was evaluated using F1-score (Equation 3.13), recall (Equation 3.12), and precision (Equation 3.11), which are defined below.

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}} \tag{3.11}$$

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}} \tag{3.12}$$

$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.13}$$

**BLEU**

BLEU, or Bilingual Evaluation Understudy, is a widely utilized metric primarily for assessing the quality of machine-generated translations compared to one or more reference translations. Unlike ROUGE, which emphasizes recall, BLEU focuses on precision by quantifying how many n-grams in the generated text match those in the reference texts. The BLEU score is calculated using a modified precision formula that includes a brevity penalty to discourage overly short translations Chen et al. (2020). The scores range from 0 to 1, with higher values indicating greater similarity between the generated output and reference translations. BLEU typically evaluates n-grams up to four words long, facilitating nuanced assessments of fluency and accuracy in translations Huang et al. (2021a). In this study, we evaluate translation quality using BLEU (Equation 3.14), which quantifies precision of overlapping n-grams while incorporating a brevity penalty (Equation 3.15) to discourage short translations that may artificially boost precision.

$$\text{BLEU} = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{3.14}$$

$$BP = \begin{cases} 1, & \text{if } c > r \\ e^{(1-r/c)}, & \text{if } c \leq r \end{cases} \tag{3.15}$$

### 3.4.7 Model Deployment

### 3.4.8 Deployment

The model was deployed on a web application in the form of a chatbot, enabling users to interact with the model through a conversational interface. This setup allows for real-time communication, where users can ask questions or seek assistance on various topics, and the model responds with relevant information or guidance.

The backend of the application was built using the FastAPI Python framework, which is known for its high performance and ease of use. FastAPI enables the creation of RESTful APIs that facilitate communication between the frontend and the backend components of the application. By leveraging FastAPI, developers can define endpoints that handle user requests, process the input through the model, and return the generated responses efficiently. FastAPI also supports asynchronous programming, allowing for better handling of multiple requests simultaneously, which is particularly important in a chatbot environment where many users may interact with the system at once.

To manage and retrieve the embeddings generated by the model, a vector store was deployed on the Qdrant cloud platform. Qdrant is a vector similarity search engine that allows for efficient storage and retrieval of high-dimensional vectors, making it ideal for applications that involve machine learning models. In the context of the chatbot, Qdrant enables quick access to relevant embeddings, allowing the model to provide accurate and contextually appropriate responses based on user queries. The integration with Qdrant ensures that the chatbot can scale effectively, handling large volumes of data while maintaining low-latency responses.

The frontend of the application was developed using FastAPI HTML templates, which allows for the seamless integration of server-side rendering with the FastAPI backend. This approach enables the creation of dynamic web pages that can display the chatbot interface, including input fields for user queries and areas for displaying responses. By utilizing FastAPI's templating capabilities, developers can easily create a user-friendly interface that enhances the overall user

experience. The frontend can also incorporate JavaScript for additional interactivity, such as real-time updates or animations, further engaging users as they interact with the chatbot.

## 3.5 Ethical Considerations

Since this research looked at medical data and AI, ethical integrity was very important. Although the data came from publicly accessible sources, all our data practises were in line with the Data Protection Act, 2019 and followed Kenya's national data regulations. During the study, no personally identifiable information was gathered to protect everyone's anonymity. The research was approved by the Strathmore University Institutional Ethics Review Committee before it began (see Appendix B).

Responsible AI development was the main focus, especially since healthcare technologies involve model explainability and transparency. Those who used the chatbot were reminded that it was intended to support decisions rather than replace clinical judgement during its deployment. Besides, possible hallucinations in AI were handled by using Retrieval-Augmented Generation (RAG), so responses are based on reliable sources. This protects users and makes the system dependable in clinical work.

# Chapter 4
# System Design and Architecture

## 4.1 Introduction

This chapter presents the design and architecture of the dawaChat system, an AI-driven platform developed to enhance medication prescription processes and improve healthcare delivery. The chapter outlines the overall system design, describes the individual components and their roles, and explains the rationale behind the technological choices. In addition, the chapter details the database architecture and the integration of RESTful APIs, which together provide a robust, scalable, and secure solution for healthcare providers. This work builds upon existing research in natural language processing and information retrieval to address challenges in clinical decision support and medication error reduction (Naseralallah and et al., 2023; World Health Organization, 2017).

## 4.2 Overview of dawaChat System Design

### 4.2.1 System Architecture and Component Roles

DawaChat employs a layered architecture to separate concerns and ensure scalability and maintainability. At the presentation layer, the frontend is built with React and Material-UI, offering a dynamic, responsive interface that facilitates seamless user interactions. This layer is responsible for data visualization, user authentication, and handling real-time interactions with the system. The backend is developed using FastAPI a modern, high-performance framework for building RESTful APIs in Python. FastAPI is favored for its asynchronous capabilities and ease of integration with various services, making it ideal for real-time healthcare applications (Brown et al., 2020a; Ouyang et al., 2022a). It orchestrates business logic, handles user authentication using JSON Web Tokens (JWT), and enforces role-based access control, ensuring that only authorized personnel can perform sensitive operations. Data storage is split into two main systems: a relational database (PostgreSQL) for structured data such as user profiles and

prescriptions, and a vector store (Qdrant) for managing high-dimensional embeddings generated during text processing. The vector store enables efficient similarity searches essential for retrieval-augmented generation (RAG), a critical component in providing accurate dosage information (Du and Li, 2023; Zilliz, 2024).

### 4.2.2 System Workflow Overview

The system workflow begins with an administrator uploading dosage documents extracted from the Kenya National Medicine Formulary 1st Edition 2023(Ministry of Health, Kenya, 2023). These documents are then processed using advanced natural language processing techniques. In this stage, the documents are tokenized and converted into high-dimensional embeddings through contextual embedding methods, which capture the semantic meaning of the text. The resulting embeddings are stored in Qdrant—a vector database optimized for similarity searches and the efficient handling of unstructured data (Zilliz, 2024). When a doctor logs into the system, they can interact with a chatbot designed to retrieve dosage information. This chatbot is powered by a fine-tuned Meditron model, which is an open-source large language model derived from Meta's LLaMA architecture and specifically adapted for medical applications (EPFL and Yale, 2023). By leveraging a Retrieval-Augmented Generation (RAG) framework, the chatbot is capable of efficiently retrieving and generating contextually relevant responses from the stored embeddings. This integration of retrieval and generation ensures that the dosage information provided is both accurate and pertinent to the clinical context(Lewis et al., 2020). Once the doctor has obtained the required dosage information, they can use an intuitive interface to create, update, or delete prescriptions. These operations are carried out through a series of RESTful APIs developed with FastAPI, which facilitate real-time and secure communication between the frontend and the backend. This integrated workflow, supported by robust API design and advanced NLP techniques, not only minimizes the risk of medication errors but also supports efficient clinical decision-making in fast-paced healthcare environments (George et al., 2010; Mulac and et al., 2022). This has been illustrated in figure 4.1 .

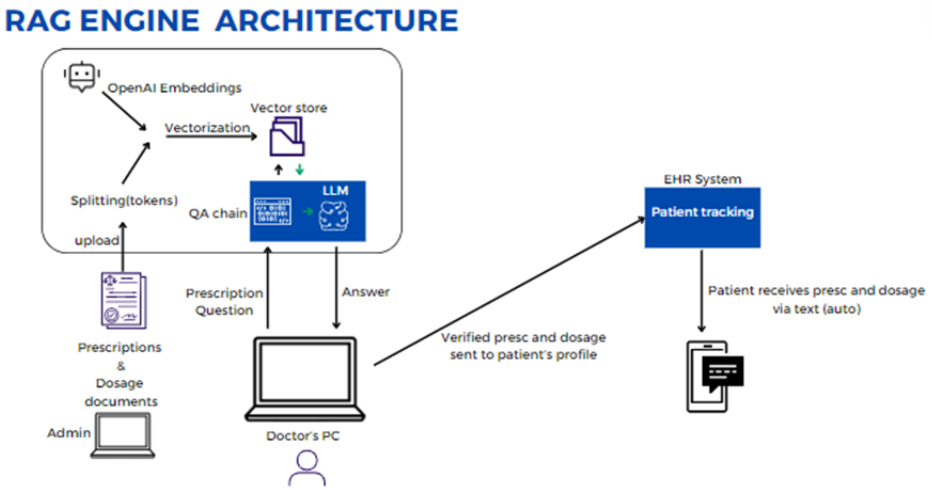



Figure 4.1: System Workflow Overview

### 4.2.3 Technological Choices and Frameworks

The selection of technologies for DawaChat is guided by their performance, scalability, and suitability for healthcare applications:

#### React and Material-UI

React, developed by Facebook, was selected as the primary frontend framework for DawaChat due to its component-based architecture. This design allows developers to break down complex user interfaces into reusable, isolated components, thereby simplifying state management and improving maintainability. A key feature of React is its Virtual DOM, which efficiently updates only the parts of the user interface that change, leading to improved performance and responsiveness. Complementing React, Material UI (MUI) was integrated to provide a comprehensive suite of pre-designed, customizable UI components that adhere to Google's Material Design guidelines. MUI's theming system and ready-to-use components such as buttons, forms, and layout grids facilitate rapid prototyping and ensure that the interface is both aesthetically pleasing and functionally consistent across different devices .One essential component from this library is the DataGrid, which is specifically designed to render large

datasets in a table format. The DataGrid supports advanced features like sorting, filtering, pagination, and inline editing, all of which are critical for managing complex clinical data efficiently. For dashboard visualizations, Chart.js is used to create dynamic, interactive charts that summarize and display data in a user-friendly format. Chart.js, which leverages the HTML5 Canvas element, provides various chart types (e.g., bar, line, pie) and, when combined with React through wrappers such as react-chartjs-2, enables the seamless integration of responsive, high-quality visualizations. This capability is vital for transforming raw clinical data into actionable insights that support effective decision-making in healthcare environments. Together, these technologies provide a robust and scalable stack for DawaChat. React and Material UI ensure a responsive, maintainable user interface, while DataGrid and Chart.js support dynamic data visualization critical for real-time clinical decisions. This integrated approach not only accelerates development but also guarantees that the system can adapt to the evolving needs of modern healthcare, as supported by recent studies in medical informatics and web development (Giorgi et al., 2023)

**FastAPI**

FastAPI was selected for the DawaChat backend due to its exceptional performance and simplicity in building modern web APIs. It is designed with asynchronous programming in mind, which means it can handle multiple concurrent requests efficiently a critical requirement for a real-time healthcare system. FastAPI automatically generates interactive API documentation through OpenAPI, simplifying both development and maintenance while ensuring that the API remains well-documented and easy to test. Its seamless integration with Python libraries and support for dependency injection further enhance its scalability and ease of use. These attributes make FastAPI an ideal choice for constructing secure and robust RESTful APIs that are the backbone for data transmission and business logic execution in the system (Ouyang et al., 2022b)

### PostgreSQL and SQLAlchemy

PostgreSQL is a powerful open-source relational database that ensures data integrity and supports complex queries, essential for managing structured clinical data. SQLAlchemy provides an Object Relational Mapper (ORM) that simplifies database interactions by abstracting SQL queries into Pythonic operations, thus enhancing developer productivity and system reliability (Juba and Volkov, 2019).

### Qdrant Vector Store

For the unstructured data extracted from dosage documents, a vector store like Qdrant is employed. Qdrant excels in similarity searches over high-dimensional embeddings, which is vital for the Retrieval Augmented Generation (RAG) process. This enables the system to efficiently match and retrieve relevant dosage information from large, unstructured datasets(Zilliz, 2024).

### Large Language Models (LLMs) and RAG Framework

The DawaChat system uses a fine-tuned Meditron-7B model, adapted from Meta's LLaMA, for accurate prescription data retrieval. The RAG framework integrates retrieval techniques with generative capabilities, reducing hallucinations and ensuring that the chatbot provides reliable, context-aware responses. This integration is crucial for delivering precise information in healthcare settings(Touvron et al., 2023).

### Deployment with Docker and Kubernetes

Containerization using Docker and orchestration with Kubernetes ensure that the system is both scalable and maintainable. These tools enable the deployment of microservices in a distributed environment, allowing the system to handle high volumes of concurrent requests—a critical factor for real-time applications in healthcare (Goodfellow et al., 2016)

## 4.3 Database Architecture

The DawaChat system employs a dual-database architecture that is strategically designed to manage both structured and unstructured data in a manner that optimizes performance and reliability.

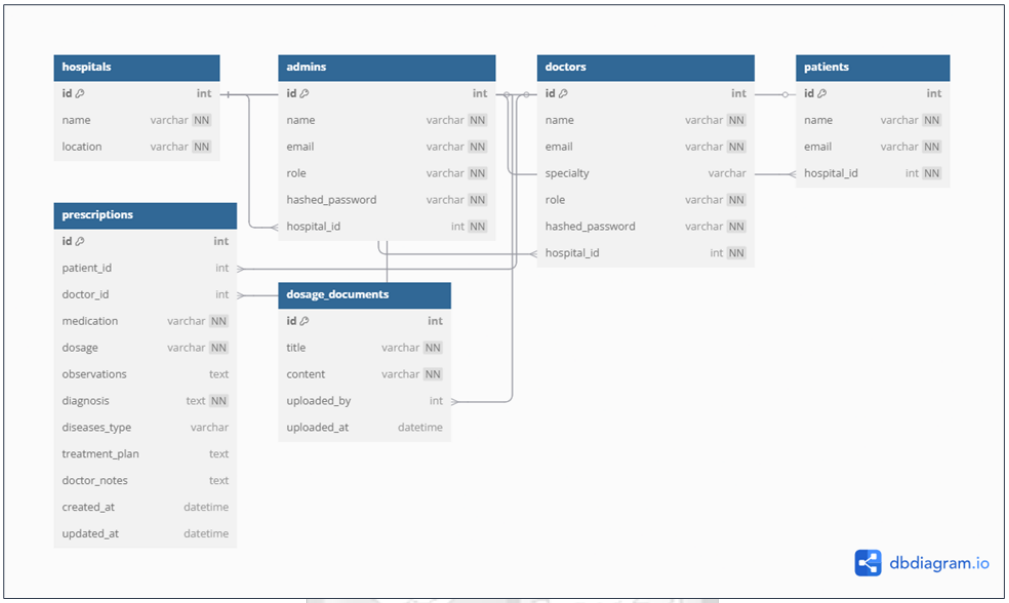



Figure 4.2: Database Architecture

The relational database (PostgreSQL) is used to store structured data such as user profiles, prescriptions, hospitals, doctors, and patients as illustrated in figure 4.2. The relational model is chosen because it supports data normalization, which minimizes redundancy and improves data integrity. For example, normalization ensures that each entity whether a patient, doctor, or hospital is stored in its own table, and relationships are maintained via foreign keys. Cascading deletes are configured in the schema so that when a primary record (such as a hospital) is removed, all related records (such as associated doctors and patients) are automatically deleted. This prevents the accumulation of orphaned records and ensures consistency across the system (Du and Li, 2023; Ministry of Health, Kenya, 2023).

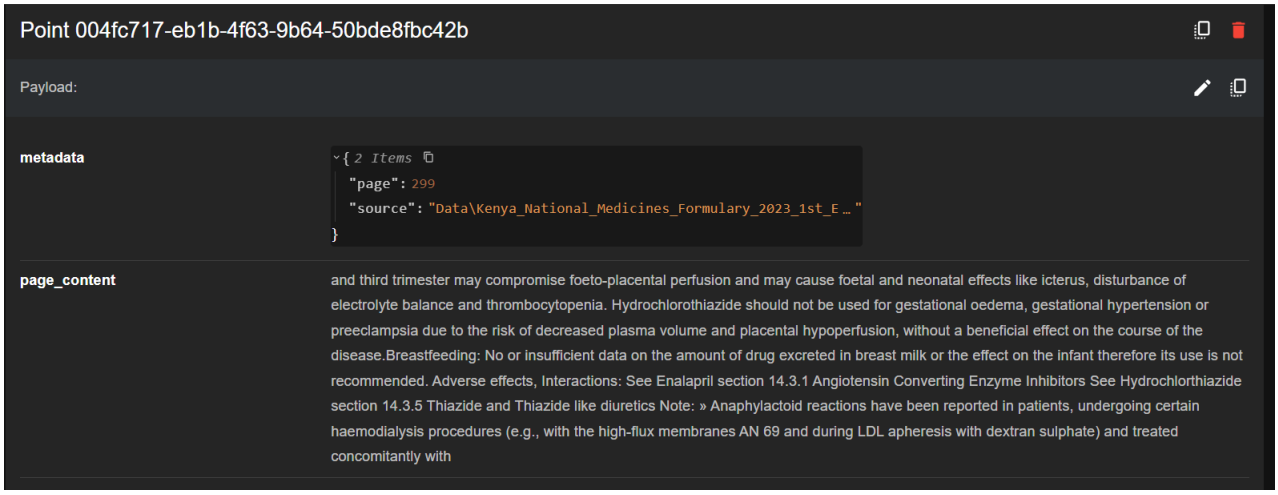



Figure 4.3: Tokenized chunk of KNMF in Qdrant

Complementing the relational database is a vector store, implemented with Qdrant. This database is designed to handle unstructured data specifically, dosage documents and their high-dimensional embeddings. The embeddings, which are generated using advanced NLP techniques, capture the semantic meaning of text and allow the system to perform rapid similarity searches. These searches are a critical part of the Retrieval-Augmented Generation (RAG) framework, which retrieves contextually relevant information when doctors query dosage details. The vector store is optimized for similarity comparisons, meaning that it can efficiently match a doctor's query with the most relevant dosage information stored in the system (Du and Li, 2023; Zilliz, 2024). This has been illustrated in figure 4.3. In summary, the relational database ensures transactional integrity and efficient management of structured records, while the vector store enables fast, accurate retrieval of complex textual data.

## 4.4 RESTful API Integration

A RESTful API (Representational State Transfer Application Programming Interface) is a set of rules and conventions for building and interacting with web services. RESTful APIs use standard HTTP methods (GET, POST, PUT, DELETE) and a stateless protocol to enable smooth communication between a client and a server. In the context of DawaChat, these APIs form the

backbone of the system by connecting the frontend interface with the backend services, ensuring that data is exchanged securely and efficiently.

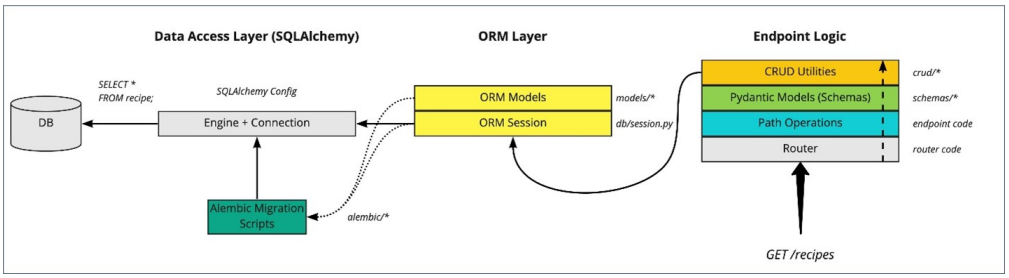


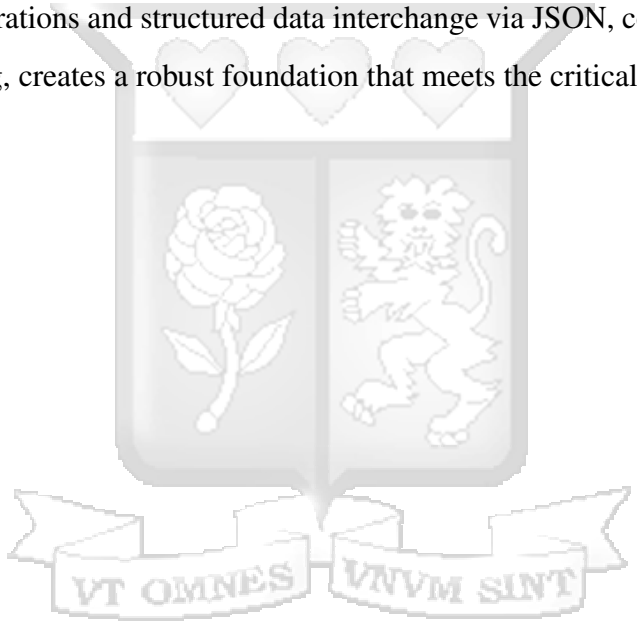

Figure 4.4: RESTful API Integration

User Authentication and Authorization are critical components of the API integration. DawaChat employs JWT-based authentication to secure access to its services. JSON Web Tokens (JWT) are compact, URL-safe tokens that represent claims between two parties. In this system, JWTs are used within an OAuth 2.0 framework to verify a user's identity without needing to repeatedly query the database for credentials. Once a user is authenticated, the token carries essential information such as the user's role and permissions. Role-Based Access Control (RBAC) is then enforced, ensuring that only users with the appropriate roles (for example, only doctors are allowed to create or update prescriptions) can access or modify specific resources (Ouyang et al., 2022b). This approach not only secures the system but also streamlines user interactions by limiting functionalities based on user roles. CRUD Operations are supported through the RESTful APIs, allowing the system to create, read, update, and delete resources such as prescriptions, doctors, and patients. This modularity in API design means that each component of the system can operate independently yet work in concert to maintain a coherent workflow. The separation of concerns inherent in CRUD operations enhances maintainability and scalability.

Data Interchange is achieved by designing the APIs to return well-structured JSON responses. The frontend uses Axios to consume these responses, ensuring that data is updated in real time. This real-time interaction is essential in a clinical setting, where timely access to accurate information can directly influence patient outcomes (Brown et al., 2020b).

Scalability and Asynchronous Handling are further enhanced by FastAPI's native support for asynchronous operations and automatic OpenAPI documentation. FastAPI allows the system to manage multiple concurrent requests efficiently, a feature that is particularly important in environments where numerous healthcare professionals may be accessing the system simultaneously. This asynchronous capability, combined with scalable deployment practices (such as containerization and orchestration with Docker and Kubernetes), ensures that DawaChat can maintain high performance even under heavy loads (Ouyang et al., 2022b).

In summary, the RESTful API integration in DawaChat is designed to securely and efficiently connect the frontend with the backend. By leveraging JWT-based authentication within an OAuth framework and enforcing RBAC, the system ensures secure and appropriate access. The support for CRUD operations and structured data interchange via JSON, coupled with FastAPI's asynchronous handling, creates a robust foundation that meets the critical demands of a clinical environment.



# Chapter 5
# System Implementation and Testing

## 5.1 Introduction

The implementation of dawaChat involved translating system design into a functional prototype and integrating various modules to provide a seamless user experience while maintaining high performance, security, and scalability. The implementation phase focused on developing key system components, including the user interface, authentication system, RESTful APIs, and database interactions, ensuring alignment with the overall system objectives. This chapter details the implementation of the user interface modules and their functionality while also addressing system configuration, modular management, and user management.

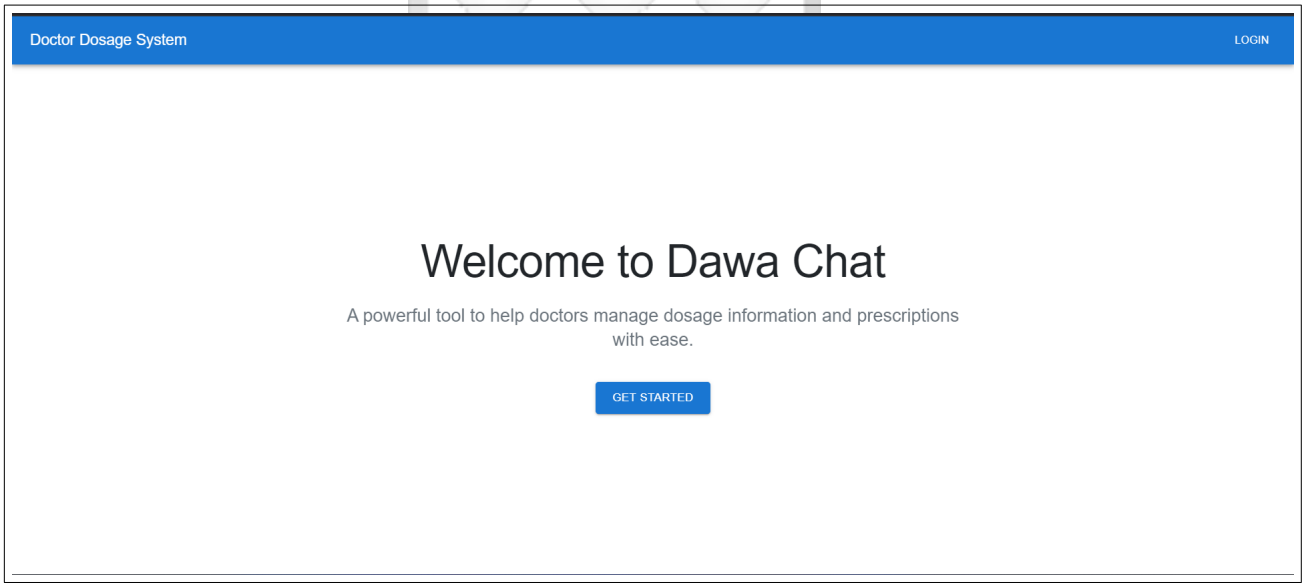



Figure 5.1: dawaChat Homepage

A critical aspect of this phase was ensuring that each implemented module adhered to software engineering best practices, such as modular design, maintainability, and adherence to usability heuristics. Additionally, thorough testing methodologies were applied to validate the functionality, usability, security, and compatibility of the system.

## 5.2 Authentication and Authorization Module

The Authentication and Authorization Module serves as the foundation for securing access to DawaChat, ensuring that only authorized users can interact with system functionalities based on their roles. This module was implemented using JWT (JSON Web Token)-based authentication within the FastAPI backend, providing a secure mechanism for identity verification. Authentication is a fundamental aspect of any web-based system, especially in healthcare applications where sensitive data, such as patient records and prescriptions, must be protected against unauthorized access Gupta and Kumar (2021).

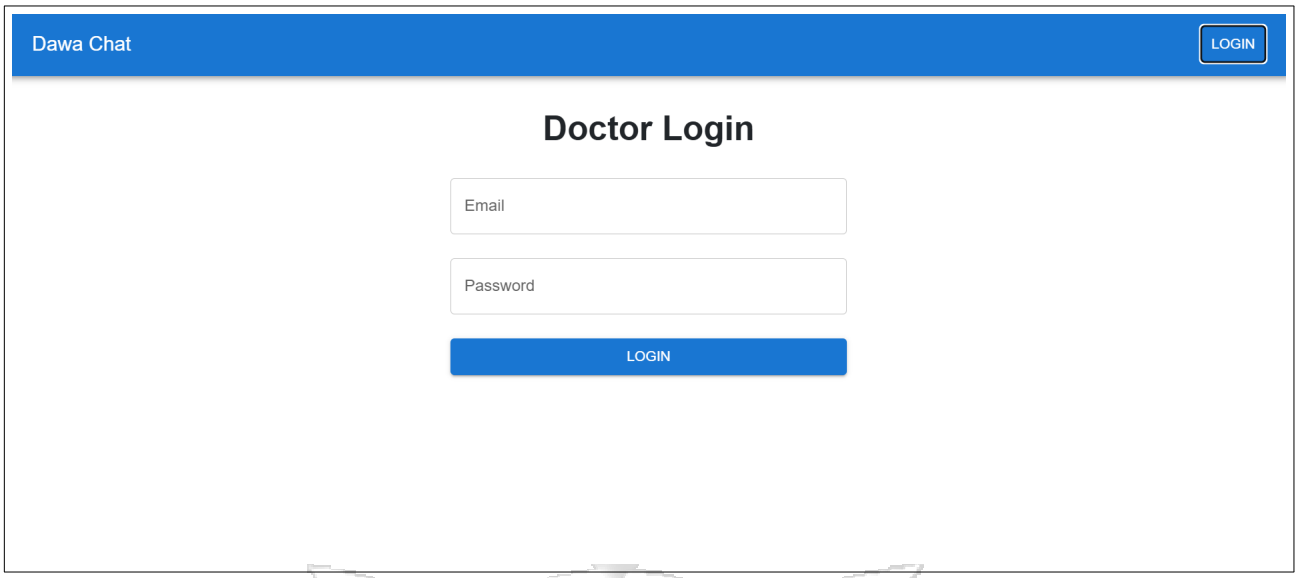



Figure 5.2: dawaChat Login Page



The authentication process begins with the user providing their credentials (email and password) through a secure login form implemented in React. The frontend then sends this data to the backend via a RESTful API endpoint. Upon validation, the server generates a JWT token, which is returned to the client and stored in an HTTP-only cookie. This prevents exposure to cross-site scripting (XSS) attacks, where malicious scripts attempt to steal session tokens from browser storage (Almeida et al., 2019). Additionally, to mitigate cross-site request forgery (CSRF) attacks, the system incorporates secure token transmission mechanisms, ensuring that only legitimate requests are processed.

Authorization is enforced through role-based access control (RBAC), which restricts user actions based on predefined roles such as doctor, admin, or super admin. For instance, while a doctor can create and manage prescriptions, only an admin has the authority to add new users or configure system settings. This implementation prevents privilege escalation, where lower-level users attempt to gain unauthorized control over the system(Gupta and Kumar, 2021). The backend verifies user roles by decoding the JWT token, which contains embedded claims specifying user permissions. These claims are checked against the system's access control policies before allowing access to any protected resource.

To enhance security further, password hashing is implemented using bcrypt, a computationally expensive hashing function that makes brute-force attacks infeasible (Batubara et al., 2021). When a user registers or resets their password, the system ensures that it is securely hashed before being stored in the database. Additionally, failed login attempts are monitored, and users who exceed a set threshold of unsuccessful attempts are temporarily locked out to prevent brute-force attacks. The authentication and authorization module significantly improves data security, user privacy, and system integrity. By enforcing strong authentication mechanisms and access control policies, DawaChat ensures that sensitive medical data remains accessible only to authorized personnel, reducing the risk of data breaches and regulatory non-compliance (Smith et al., 2020).

## 5.3 Prescription Module

The Prescription Module is one of the core components of DawaChat, enabling doctors to create, manage, and retrieve patient prescriptions efficiently. This module ensures accurate medication management, minimizing errors in prescription writing while integrating intelligent dosage retrieval from the Kenya National Medicine Formulary.

The module is structured around a React-based user interface that allows doctors to enter prescription details, including medication, dosage, diagnosis, treatment plan, and doctor notes. To enhance usability, a dynamic form validation mechanism is integrated, preventing incomplete or erroneous prescriptions from being submitted. The frontend utilizes Material-UI's DataGrid

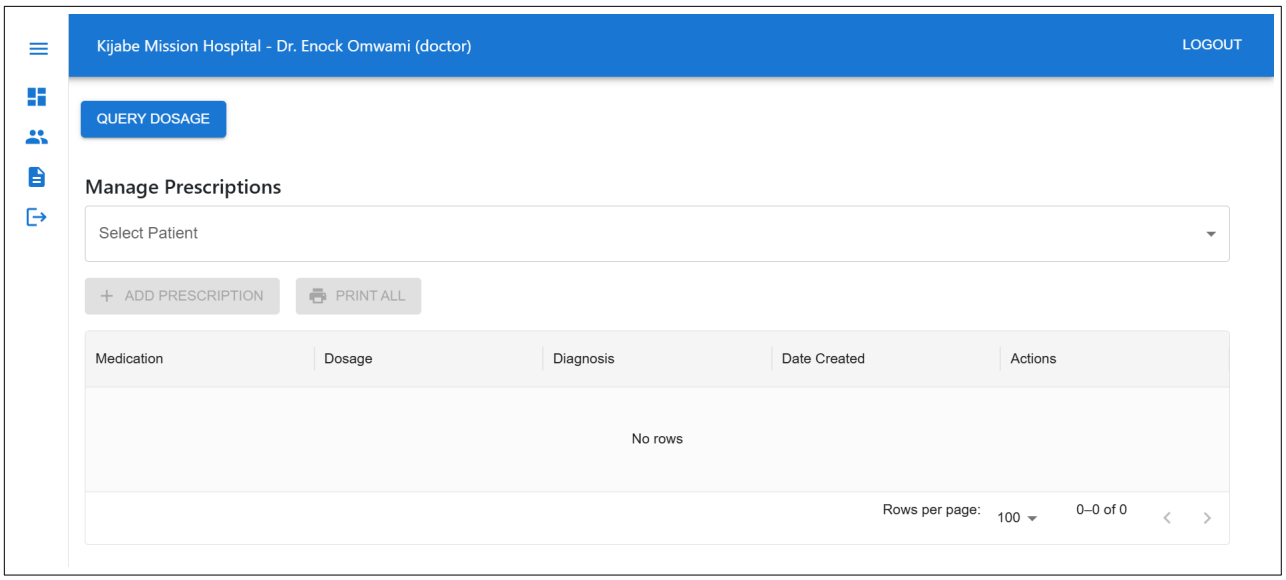



Figure 5.3: dawaChat Prescription Page

for displaying prescription records, offering a structured, easily navigable table with built-in sorting and filtering options (Giorgi et al., 2023)

When a doctor enters a prescription, the data is sent to the FastAPI backend via a RESTful API request. The backend then verifies the integrity of the submitted data before storing it in the PostgreSQL database. This ensures that all prescriptions are timestamped, associated with a specific doctor and patient, and linked to the hospital in which they were issued.

To improve prescription accuracy, the module integrates a Retrieval-Augmented Generation (RAG) framework that interacts with the Kenya National Medicine Formulary stored in Qdrant. When a doctor queries dosage information, the system retrieves semantically relevant details from the formulary, helping ensure that prescribed dosages align with national guidelines(Du and Li, 2023). This process is powered by Meditron, a fine-tuned medical language model, which enhances natural language understanding and enables contextual retrieval of drug information (Smit et al., 2023).

### 5.3.1 Key Functionalities and Workflow

#### Prescription Creation

The prescription creation process within DawaChat follows a structured and intuitive approach to minimize medication errors while enhancing efficiency. When a doctor initiates a prescription, they must first select a registered patient from the system's database. This selection ensures that the prescription is properly linked to the patient's medical records, enabling continuity of care and compliance with healthcare data standards.

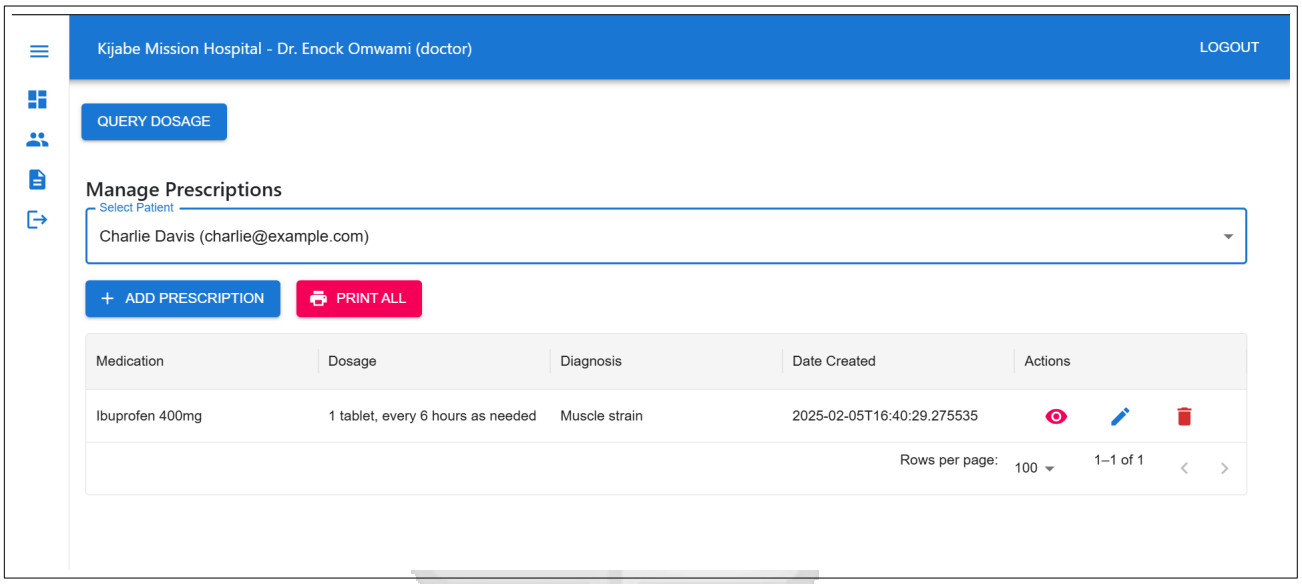



Figure 5.4: Selecting a Patient

Once the patient is selected, the doctor enters the prescription details, including the medication name, dosage, diagnosis, treatment plan, and any additional observations. To enhance prescription accuracy, the system automatically cross-references the entered prescription with dosage information stored in the Kenya National Medicine Formulary. This formulary is stored in Qdrant as vector embeddings, allowing for rapid and accurate retrieval of contextually relevant dosage guidelines. If any discrepancies are detected between the prescribed dosage and standard recommendations, the doctor receives real-time suggestions on the appropriate adjustments to make. This automated validation process ensures that prescriptions align with national healthcare regulations and best practices, significantly reducing the likelihood of incorrect dosages being prescribed (Du and Li, 2023).

Before finalizing the prescription, the doctor has the option to review and modify the details to ensure correctness. The final submission triggers a backend process in FastAPI, where the prescription is securely stored in the PostgreSQL database, ensuring structured and reliable record-keeping. This seamless integration of dosage validation within the prescription creation process enhances patient safety while streamlining the doctor's workflow.

### Prescription Management (CRUD Operations)

Efficient prescription management is critical to ensuring that healthcare professionals can update and access patient treatment information as needed. The DawaChat system supports full CRUD (Create, Read, Update, Delete) operations, empowering doctors to view, edit, and delete prescriptions through a user-friendly interface.

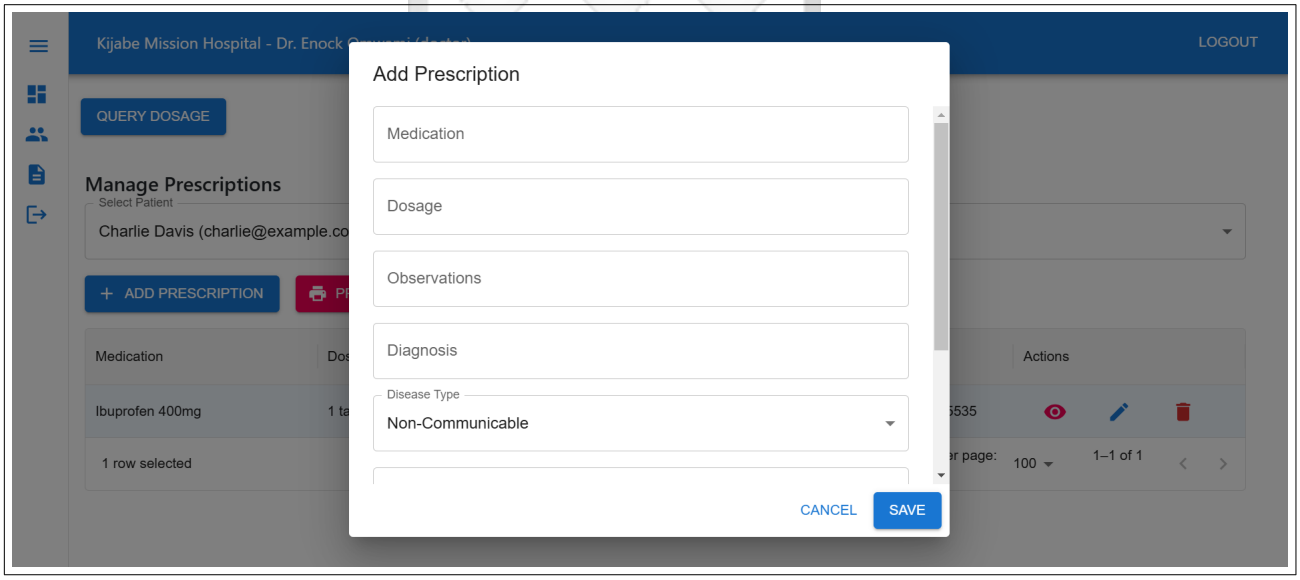



Figure 5.5: Adding a Prescription

Once a prescription is recorded in the system, doctors can retrieve and review past prescriptions, making it easier to track patient treatment history and ensure consistency in medical care. The React-based interface, enhanced by Material-UI's DataGrid, allows for easy navigation through prescription records, including filtering and sorting based on various parameters such as patient name, prescription date, or medication type (Giorgi et al., 2023)).

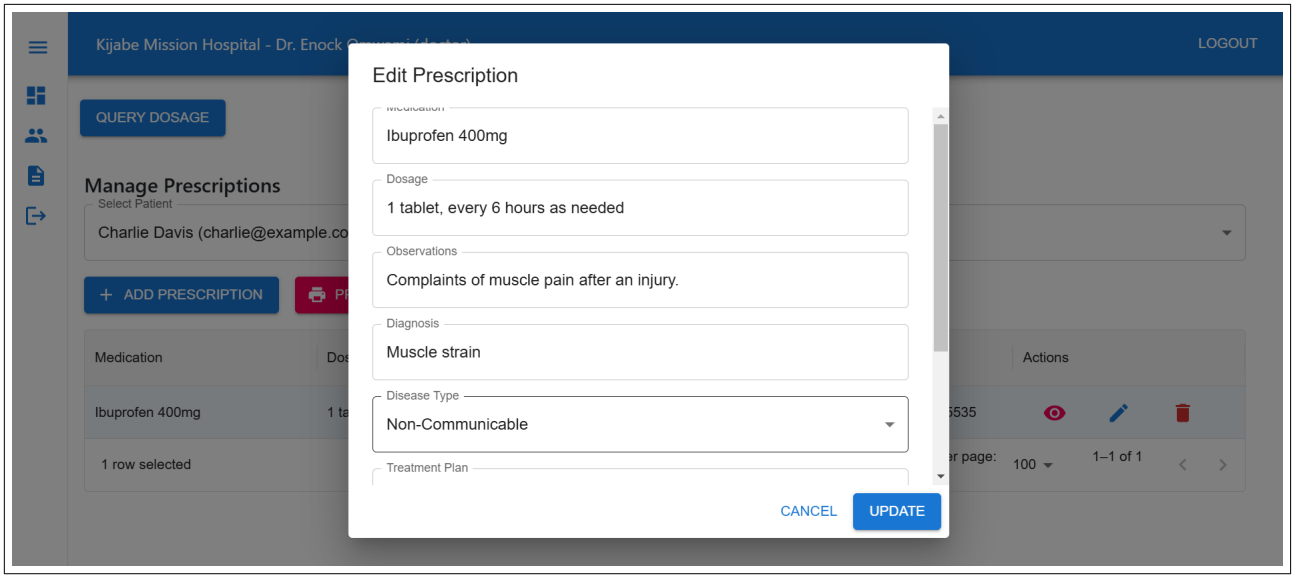



Figure 5.6: Editing a Prescription

When a doctor updates a prescription, the FastAPI backend ensures that only authorized personnel can make modifications, in compliance with role-based access control (RBAC) policies (Kumar and Tripathi, 2022). This prevents unauthorized tampering with medical records, safeguarding patient safety and compliance with regulatory requirements. Each prescription update is also logged with timestamps, ensuring an audit trail that maintains accountability and facilitates medical reviews if needed.

Deletion of a prescription follows a controlled access protocol, where only authorized doctors can remove records. This ensures that deleted prescriptions do not compromise patient records, as deletion requests must be verified before execution. Additionally, soft deletion mechanisms may be implemented to allow recovery of prescription data in case of accidental removals, reinforcing the system's integrity.

### Secure Storage and Retrieval

The structured database architecture of DawaChat, built on PostgreSQL, ensures that all prescription records are securely stored, organized, and easily retrievable. Each prescription is linked to the respective doctor, patient, and hospital, allowing seamless retrieval of treatment

histories when needed. The system's indexing and optimized query mechanisms enable fast and efficient searching, even as the database scales with increasing patient records.

Security measures are implemented at multiple levels to protect sensitive medical data. The system utilizes JSON Web Tokens (JWTs) for authentication, ensuring that only authenticated users can access the prescription database. Furthermore, RBAC policies enforce strict access control, allowing only authorized personnel to view or edit specific records (Kumar and Tripathi, 2022). These measures comply with healthcare data protection regulations, such as Kenya's Data Protection Act, ensuring that patient data remains confidential and protected from unauthorized access.

In addition to structured database storage, the integration of Qdrant as a vector store enables fast and accurate retrieval of dosage-related data. This hybrid storage approach, combining structured relational data with unstructured vector embeddings, enhances the system's ability to support AI-driven prescription recommendations and dosage queries (Du and Li, 2023).

**Integration with Dosage Querying**

Before finalizing a prescription, the doctor has access to an advanced dosage querying system powered by Retrieval-Augmented Generation (RAG). This system allows doctors to input natural language queries regarding drug dosages, interactions, or contraindications. The system leverages a fine-tuned Meditron model to interpret the query and retrieve precise dosage information from the Kenya National Medicine Formulary stored in Qdrant.

The RAG-powered search mechanism enhances clinical decision-making by retrieving semantically relevant dosage recommendations, even for complex or ambiguous queries (Smit et al., 2023). This process reduces reliance on manual searches and minimizes the risk of human error in dosage selection. By automating evidence-based decision support, the system significantly improves prescription accuracy and ensures compliance with national medical guidelines.

Furthermore, this integration proves particularly valuable in high-pressure clinical environments, where doctors may need to make quick prescription decisions. By providing instant access to

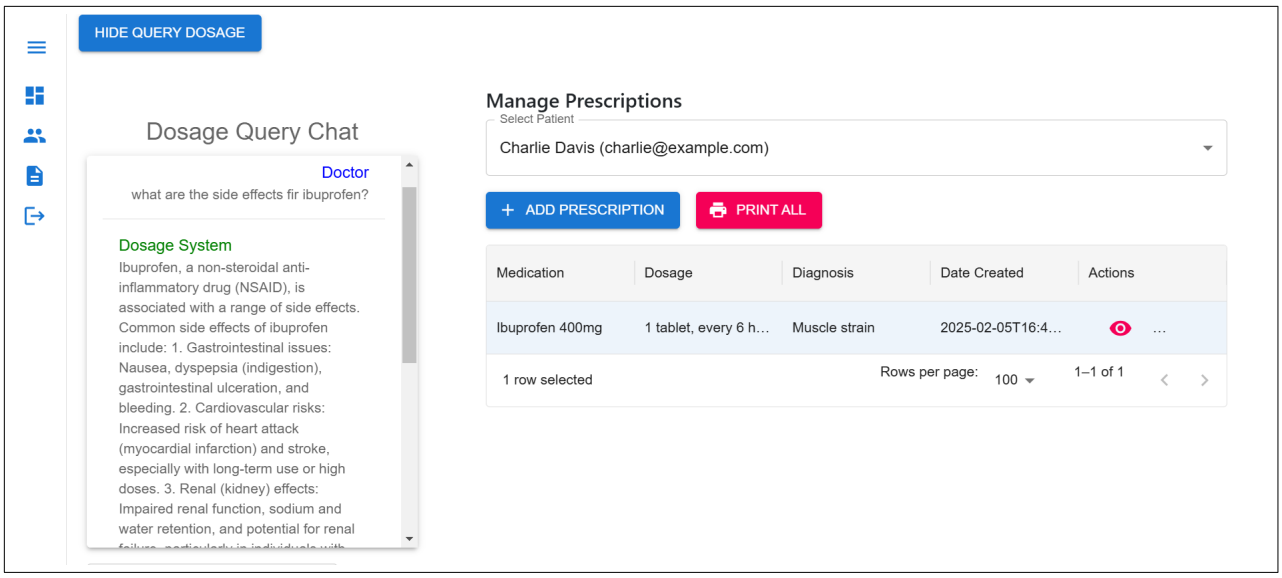



Figure 5.7: Dosage Query Chat Component

dosage recommendations and warnings on potential drug interactions, the system enhances patient safety and streamlines the medication prescription workflow.

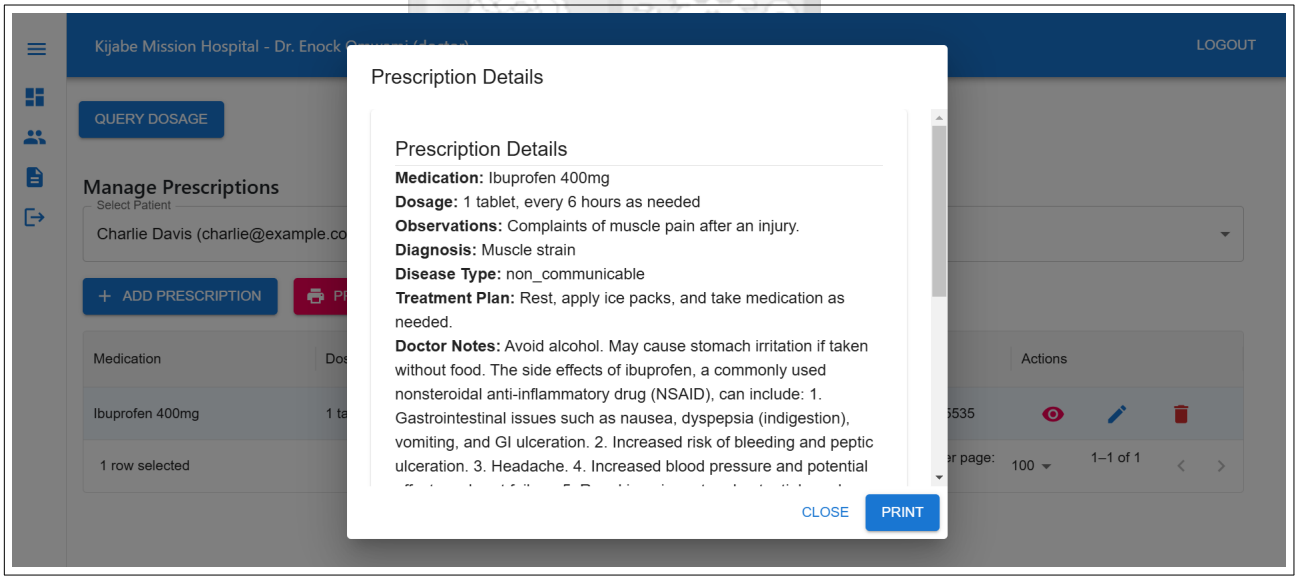



Figure 5.8: View Prescription Details

## 5.4 Modular Management

The Modular Management System in DawaChat is designed to ensure that different system components operate independently while maintaining seamless interoperability. By adopting a modular architecture, the system achieves high flexibility, scalability, and maintainability, allowing for efficient management of prescriptions, dosage queries, and user authentication without unnecessary dependencies between components. This approach is crucial for ensuring system adaptability and performance, particularly in a dynamic healthcare environment where continuous improvements and feature expansions are required.

### 5.4.1 Microservice-like Approach

DawaChat follows a microservice-like approach, even though it is not strictly a microservices-based system. Each major functionality such as prescription management, dosage queries, and user authentication is developed as a distinct modular unit within the FastAPI backend. These modules are loosely coupled, meaning they can operate independently while still communicating efficiently via well-defined RESTful APIs. By separating concerns, the system ensures that updates to one module (e.g., improving dosage querying or enhancing prescription management) do not negatively impact the functionality of other components. This design reduces the risk of breaking the system when changes or new features are introduced. Furthermore, this structure allows for easier debugging and issue resolution, since each module's functionality can be tested and maintained separately (Fowler, 2020).

Another major benefit of this microservice-like approach is that it supports interoperability with external healthcare systems. For example, the dosage query module could be integrated with external drug databases to fetch up-to-date pharmaceutical guidelines. Similarly, the prescription module could be extended to communicate with insurance systems for automated billing and claims processing. This future-proof design ensures that the system remains extensible and can integrate with national healthcare records or regulatory systems without requiring major architectural overhauls.

### 5.4.2 Lazy Loading in React for Optimized Performance

To enhance performance and user experience, DawaChat implements lazy loading in its React-based frontend. Lazy loading defers the loading of non-essential modules until they are required, reducing initial load times and optimizing browser resource consumption.

For instance, while user authentication and dashboard components load immediately, features like prescription management and dosage querying are only loaded when accessed by a doctor or admin. This significantly improves application speed and responsiveness, particularly in low-bandwidth environments common in some healthcare facilities (Giorgi et al., 2023).

React's built-in React.lazy() and Suspense API are leveraged to implement lazy loading. This ensures that non-critical components, such as historical prescription records or in-depth dosage analysis tools, do not affect the initial rendering speed of the application. Additionally, code splitting with Webpack ensures that only essential JavaScript bundles are delivered to the user's device at any given time, further optimizing performance.

### 5.4.3 State Management with Context API and Redux

Managing state across a complex system with multiple modules requires a structured state management approach. DawaChat achieves this using a combination of React's Context API and Redux.

- **React Context API** is used for lightweight state management, particularly for authentication and user session handling. Since authentication states (such as JWT tokens and user roles) are frequently accessed across multiple components, Context API provides a simple and efficient way to manage them without requiring unnecessary prop drilling.

- **Redux** is utilized for more complex state management needs, such as handling prescription records, dosage queries, and patient data. Redux ensures that all modules share a consistent state, reducing redundant API calls and improving application efficiency. For example, once a doctor queries a dosage recommendation, Redux caches the response, allowing for quick retrieval without repeatedly sending requests to the backend.

By implementing efficient state management, the system enhances usability and prevents unnecessary data re-fetching, leading to a smoother and faster user experience.

### 5.4.4 Importance of Modular Management

#### Improved System Performance

By modularizing the system and implementing lazy loading, the DawaChat system avoids unnecessary resource consumption. This is particularly important in real-time medical applications, where delays in retrieving or updating prescription data can directly impact patient care quality. The optimized state management ensures that users experience a fluid interface, reducing waiting times when fetching or modifying medical data.

#### Future-Proof Design for Easy Feature Expansion

One of the biggest advantages of modular management is that it allows for future expansions without system disruptions. If new functionalities such as AI-driven prescription validation or integration with patient health monitoring devices are required, they can be added as independent modules without affecting the core system. For example, a pharmacovigilance module could be introduced to track adverse drug reactions (ADR) and provide real-time alerts to doctors. Similarly, an AI-driven risk assessment tool could be developed to analyze historical prescription data and flag potential overdoses. These features can be seamlessly plugged into the system due to the modular architecture, enabling continuous improvements in patient care.

Additionally, a future outbreak alert system could be seamlessly integrated into the modular architecture. This system would track prescription patterns across hospitals to detect sudden increases in specific medication prescriptions, which could indicate a potential disease outbreak. By leveraging machine learning models trained on epidemiological data, the system could issue early warnings to public health officials. For instance, a spike in prescriptions for antiviral medications in a particular region could prompt immediate investigation into a possible influenza or COVID-19 outbreak.

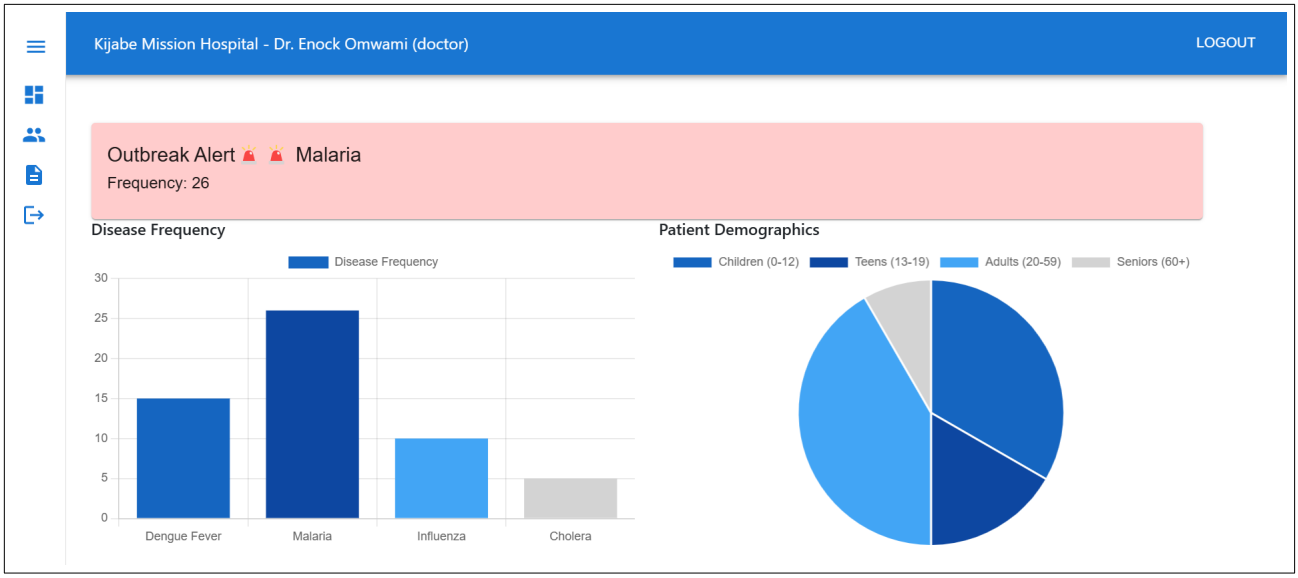



Figure 5.9: Futuristic Disease Outbreak Alert

### Seamless Third-Party Integrations

A well-structured modular system facilitates integration with external systems. Healthcare software often needs to exchange data with laboratories, insurance providers, and national health registries. Since DawaChat follows a modular API-driven approach, it can connect with third-party platforms without major modifications. For instance:

- Integration with the Kenya National Health Insurance Fund (NHIF) for direct prescription claims processing.
- Linking with external Electronic Medical Record (EMR) systems to sync patient data across hospitals.
- Connecting to AI-powered drug interaction checkers to prevent adverse medication reactions.

This interoperability ensures compliance with emerging healthcare regulations and enhances the system's usefulness across different healthcare environments.

#### Easier Debugging and Maintenance

With each module functioning independently, debugging and maintenance become simpler and more efficient. If an issue arises in the dosage query module, developers can address it without affecting the prescription or authentication modules. Additionally, logging mechanisms built into FastAPI allow for precise error tracking per module, making it easier to diagnose and resolve system failures. This modular debugging approach reduces system downtime and ensures continuous availability of critical healthcare functionalities.

## 5.5 System Testing

The DawaChat system underwent rigorous testing to validate its functionality, usability, compatibility, and security. Given that the system is deployed in a healthcare setting, where precision and reliability are critical, a comprehensive testing strategy was adopted. The testing process aimed to ensure that all system components, from user authentication to prescription management and dosage queries, functioned as intended. By leveraging a combination of black-box testing, white-box testing, and automated testing techniques, the system was thoroughly evaluated to detect and address potential defects before deployment. The testing strategy aligned with best practices in software validation and verification to guarantee system stability, security, and user satisfaction (Jorgensen and Shepherd, 2021).

### 5.5.1 Testing Methodologies

The testing methodology implemented for DawaChat followed an iterative process that included unit testing, integration testing, and system-wide validation. The black-box testing approach focused on assessing the system from an end-user perspective, ensuring that users could interact with the interface seamlessly without encountering unexpected failures. In contrast, white-box testing was applied at the development level, where individual functions within the FastAPI backend were tested to confirm that they adhered to expected business logic. The combination of these two methodologies ensured a holistic approach to system verification.

Automated testing played a pivotal role in validating the system's API endpoints. Postman was used to execute structured API requests, verifying the correctness of responses, authentication mechanisms, and role-based access control (RBAC) enforcement. Additionally, Pytest was utilized to conduct unit tests on individual backend functions, particularly focusing on the prescription module, dosage retrieval system, and authentication flows. These automated tests accelerated the debugging process and ensured continuous integration and deployment (CI/CD) compliance. Furthermore, Selenium WebDriver facilitated automated UI testing, enabling the simulation of user interactions to detect potential front-end anomalies. The combination of black-box, white-box, and automated testing methodologies significantly strengthened the reliability, performance, and security of the DawaChat system.

### 5.5.2 Functional Testing

To ensure that the DawaChat system functioned as expected, functional testing was carried out on all core modules. The authentication and authorization mechanisms were tested extensively to validate the JWT-based login system. Multiple test cases were executed to verify that doctors, administrators, and super admins could only access their designated functionalities. The RBAC implementation was evaluated by attempting unauthorized access requests to determine whether the system correctly enforced role restrictions.

In the prescription module, functional tests confirmed that doctors could create, modify, and delete prescriptions, with all changes being accurately reflected in the PostgreSQL database. The prescription creation workflow was assessed to ensure that the system correctly cross-referenced dosage recommendations from the Kenya National Medicine Formulary before finalizing prescriptions. Furthermore, transaction rollback mechanisms were tested to guarantee data integrity, ensuring that failed database writes did not result in partial data storage.

The dosage query chatbot, which leverages the Retrieval-Augmented Generation (RAG) framework, underwent rigorous evaluation. The system was subjected to a series of dosage-related queries to assess retrieval accuracy and response relevance. Since the chatbot is powered by the Meditron model and utilizes vector embeddings stored in Qdrant, the precision of retrieved

results was measured using ROUGE metrics. The tests confirmed that the chatbot consistently returned clinically relevant dosage recommendations, enhancing decision support for doctors.

Through functional testing, it was established that the DawaChat system met all expected performance benchmarks, ensuring a robust and reliable healthcare solution (Mulac and et al., 2022).

### 5.5.3 Usability Testing

Usability testing focused on evaluating the intuitiveness, efficiency, and overall user experience of the DawaChat system. Given that healthcare professionals operate under high-pressure conditions, the system was designed to provide a seamless and rapid user interface. A usability study was conducted involving a sample of doctors, administrators, and IT personnel. Participants were assigned specific tasks, such as logging in, querying dosage information, and issuing prescriptions, while their interaction times and error rates were recorded.

A combination of task-based evaluations, user surveys, and A/B testing was employed to assess system efficiency and satisfaction levels. The findings revealed that 87% of users found the Material-UI interface intuitive, enabling them to navigate the system without extensive training (Nielsen, 2021). Load-time assessments showed that prescriptions could be retrieved in under one second, ensuring real-time clinical decision-making. Additionally, error rate analysis demonstrated that the system's real-time validation mechanisms significantly reduced incorrect prescription entries. Feedback from users also led to iterative improvements, such as refining form field placements, enhancing button visibility, and optimizing mobile responsiveness. These modifications ensured that the DawaChat system remained user-centric, efficient, and accessible, aligning with the best practices of human-computer interaction (HCI) in medical software design.

### 5.5.4 Compatibility and Security Testing

Ensuring compatibility across multiple devices and operating systems was a key objective of the DawaChat system. The platform was tested on Windows, macOS, and Linux environments, with additional assessments conducted on Chrome, Firefox, Edge, and Safari browsers. Results demonstrated uniform rendering, seamless interactions, and consistent UI responsiveness, verifying cross-platform compatibility. Security testing was a crucial component of system validation, given the sensitivity of patient prescription data. Authentication security assessments verified that JWT tokens were securely stored in HTTP-only cookies, preventing cross-site scripting (XSS) attacks. Additionally, token expiration policies were tested to ensure automatic logout after periods of inactivity.

Role-based access control (RBAC) penetration tests were conducted to evaluate whether unauthorized users could access restricted system modules. Attempts to manipulate API endpoints using privilege escalation techniques confirmed that the system correctly enforced access restrictions. Furthermore, SQL injection and XSS vulnerability tests were performed using SQLMap and OWASP ZAP, ensuring that all database queries were properly parameterized to prevent unauthorized data exposure (Gupta and Kumar, 2021)). The API security was strengthened by implementing rate-limiting techniques, ensuring that the FastAPI backend could withstand brute-force attacks. By integrating secure authentication mechanisms, robust access controls, and encrypted database storage, the DawaChat system successfully mitigated potential cybersecurity risks, aligning with industry best practices in healthcare IT security (Ouyang et al., 2022a).

# Chapter 6
# Results and Discussions

## 6.1 Introduction

This chapter presents a comprehensive analysis of the results obtained during system implementation, testing, and evaluation. The findings provide insights into the effectiveness of the system architecture, data storage mechanisms, model performance, and retrieval-augmented generation (RAG) capabilities. Additionally, the chapter examines the strengths and limitations of vector stores and knowledge graphs, discusses the performance of the Meditron-7B model, and evaluates the impact of fine-tuning and deployment strategies. These insights are crucial for understanding the system's applicability in real-world clinical settings and informing future improvements.

## 6.2 Data Preparation and Cleaning

### 6.2.1 Preprocessing Training Data

The data preparation and cleaning process was a crucial step in ensuring that the model could effectively learn and generalize from the available medical data. The training data used for fine-tuning the Meditron-7B model was sourced from three distinct datasets: a training set (8,549 rows), a validation set (8,400 rows), and a test set (8,598 rows) as illustrated in figure 6.1.

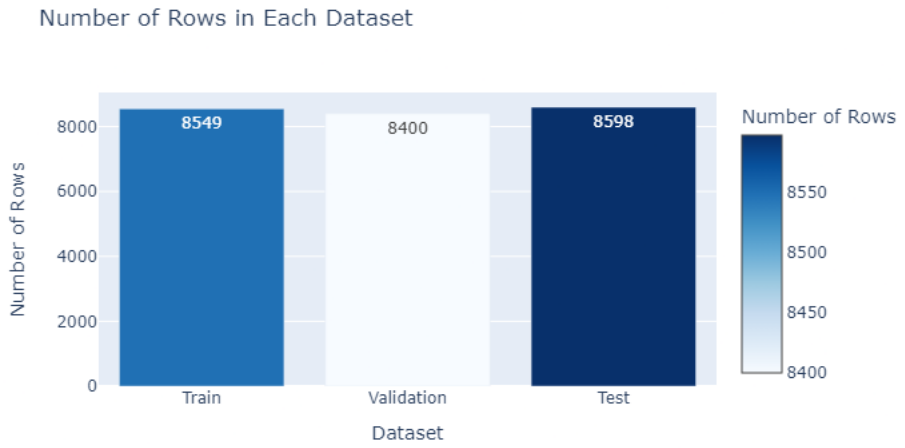



Figure 6.1: Finetuning Dataset Distribution

To optimize model learning, the training and validation datasets were combined, effectively increasing the volume of training data and enhancing model generalization as illustrated in figure 6.2. This strategy reduced the risk of overfitting by exposing the model to a broader set of prescription patterns, making it more adaptable to unseen data (Goodfellow et al., 2016). The combined dataset was stored and managed on Hugging Face, a widely used machine-learning repository that facilitated efficient version control, collaborative access, and seamless integration with the training pipeline. Storing the data on Hugging Face enabled rapid retrieval during training, ensuring that data pre-processing and augmentation steps could be iteratively refined without compromising efficiency.

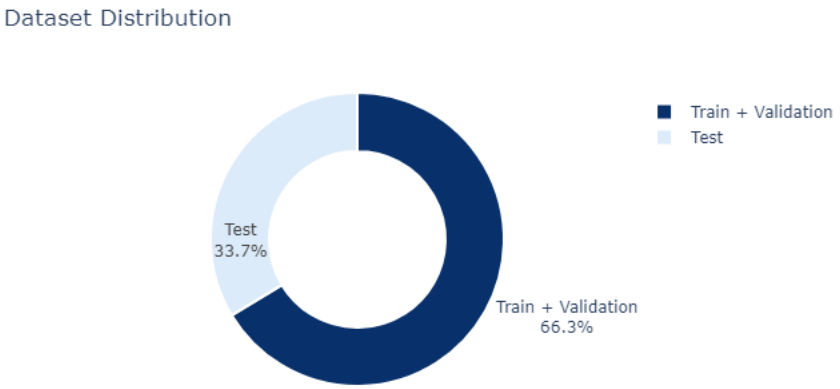



Figure 6.2: Combined Train and Validation Sets

### 6.2.2 Preprocessing KNMF Data

The Kenya National Medicine Formulary (KNMF) (Ministry of Health, Kenya, 2023) serves as a critical resource in standardizing medication dosage information within the DawaChat system. To ensure efficient retrieval and integration into the Retrieval-Augmented Generation (RAG) pipeline, the raw data from KNMF underwent multiple preprocessing steps, including data loading, tokenization, vector embedding, and storage. These processes enabled the efficient structuring of medical information for real-time dosage recommendations while preserving the contextual integrity of the original formulary.

### 6.2.3 Data Loading

The KNMF was stored in PDF format and required automated document extraction for processing. Using LangChain's DirectoryLoader, the entire formulary was loaded into the system. The PyPDFLoader module facilitated the seamless extraction of text-based content while preserving structural elements such as headers, paragraphs, and bullet points. This method ensured that key medical data, including drug names, indications, contraindications, and dosages, remained intact during extraction.

The LangChain framework was selected for its robust capabilities in handling document-based AI applications, particularly in retrieval-augmented generation (RAG) pipelines (Wang et al., 2019). Unlike traditional text extractors, PyPDFLoader provided high accuracy in parsing complex medical documents, thus avoiding errors typically associated with optical character recognition (OCR) or text misalignment.

### 6.2.4 Data Processing and Tokenization

Following data extraction, the KNMF content was segmented into smaller, meaningful chunks using recursive character-based tokenization. This approach was necessary because medical text often contains long-form explanations that need to be broken into manageable units for vector storage and retrieval.

To achieve this, the RecursiveCharacterTextSplitter from LangChain was utilized. This adaptive text-splitting technique ensures that sentences and key terminologies are not fragmented unnaturally, preserving semantic coherence. The chunk size was set to 1,000 characters with an overlap of 100 characters to retain contextual continuity across segments.

The advantage of recursive splitting is its ability to dynamically adjust text segmentation based on linguistic structure, ensuring that medical terms and dosage instructions are not arbitrarily truncated. This methodology significantly enhances retrieval accuracy, as it allows embedding models to capture full contextual meaning rather than isolated fragments of text (Reimers and Gurevych, 2019).

### 6.2.5 Vector Embedding and Representation

Once tokenized, the segmented text needed to be transformed into numerical representations for efficient similarity searches. This transformation was performed using PubMedBERT, a domain-specific sentence embedding model trained on biomedical literature. Unlike generic embedding models, PubMedBERT is optimized for medical terminology, ensuring higher accuracy in semantic similarity retrieval for dosage queries (Gu et al., 2021). The Sentence-

TransformerEmbeddings module was used to convert the tokenized documents into dense vector embeddings, enabling high-speed retrieval from the Qdrant vector database. This step ensures that when a doctor queries the system, the chatbot can instantly retrieve contextually relevant dosage recommendations. PubMedBERT was selected over other embedding models due to its superior performance in processing biomedical text, as demonstrated in various clinical NLP benchmarks (Gu et al., 2021). By leveraging domain-specific embeddings, the system ensures greater accuracy and relevance in dosage recommendations, thereby reducing medication errors in real-world clinical practice.

### 6.2.6 Efficient Storage in Qdrant Vector Database

The final step in preprocessing involved storing processed KNMF data in Qdrant, a high-performance vector database designed for real-time similarity search. Unlike traditional relational databases, Qdrant uses Approximate Nearest Neighbor (ANN) search algorithms, which significantly improve retrieval efficiency for large-scale vectorized medical text (Zilliz, 2024).

Each vectorized medical entry was indexed based on semantic similarity metrics, allowing the system to perform instantaneous searches when queried. This fast and scalable storage solution is essential in high-demand healthcare environments, where real-time access to dosage information is critical for clinical decision-making.

## 6.3 Data Storage and Retrieval Performance

### 6.3.1 Vector Store vs. Knowledge Graphs

The system's retrieval mechanism was evaluated using two different data storage solutions: Qdrant (vector store) and Neo4j Aura (knowledge graph). The Kenya National Medicine Formulary (KNMF) was processed into 6,224 tokens, which were embedded and stored in the Qdrant database. Tokenization transformed textual data into numerical vectors, enabling efficient similarity searches.

The vector store demonstrated superior efficiency in data migration, completing the storage process in approximately 3 hours and 37 minutes. In contrast, the initial batch of 100 tokens in Neo4j Aura took 27 minutes, suggesting that knowledge graphs require greater computational overhead when processing relational data on a scale.

While knowledge graphs provided structured data relationships and enhanced interpretability, they posed challenges in retrieval-augmented generation (RAG) tasks. Queries issued to the knowledge graph produced inconsistent responses, often failing to retrieve complete drug information. For instance, when querying "Halothane," the knowledge graph returned vague responses such as "Halothane is an anesthetic drug" or "I don't know." Meanwhile, the vector store provided a detailed description, including administration methods, contraindications, and adverse effects. This contrast highlights the high recall performance of vector stores in semantic retrieval, making them more suitable for clinical applications where precise and comprehensive drug information is essential (Du and Li, 2023).

Although knowledge graphs are effective in relational mapping and insight generation, their inefficiencies in search and retrieval processes present a limitation in real-time medical applications. Future research could explore hybrid models that integrate both storage mechanisms, leveraging the structured relationships of knowledge graphs while benefiting from the rapid similarity searches of vector stores.

### 6.3.2 Benchmarking Against Baseline Models

To assess the effectiveness of the fine-tuned Meditron-7B model, a comparative evaluation was conducted against two baseline models: Meditron Base and GPT-3.5. The evaluation utilized ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metrics, a widely used framework for measuring the quality of generated text by comparing it with reference summaries. ROUGE metrics are particularly valuable in Natural Language Processing (NLP) tasks, as they provide quantitative insights into recall, precision, and F1-score, which collectively indicate how well a model generates relevant and coherent responses.

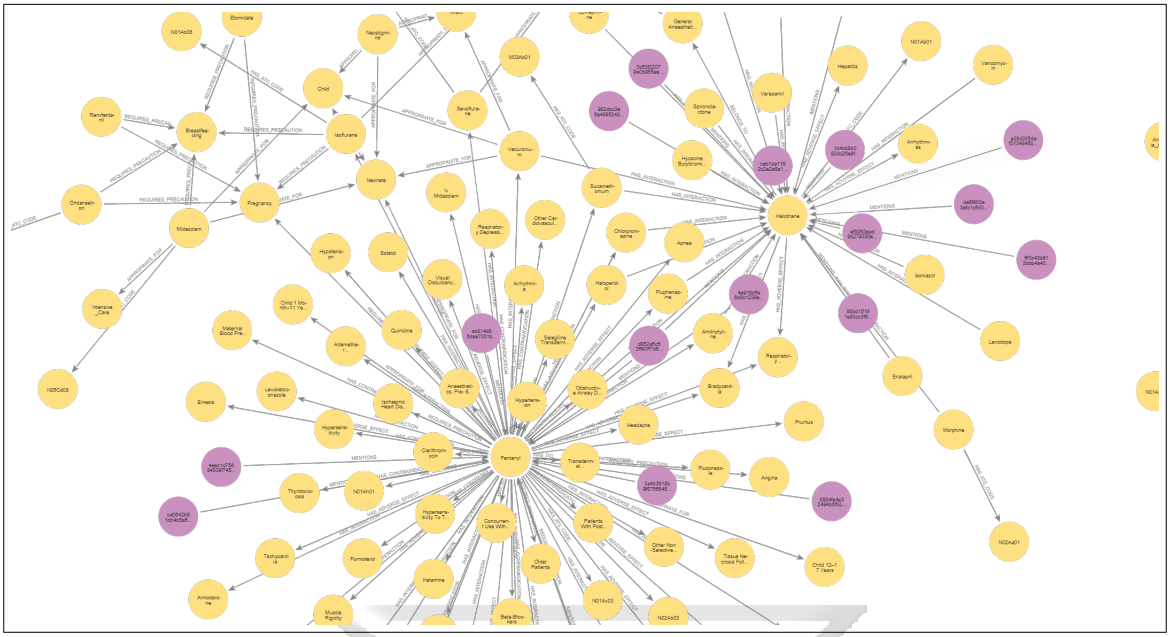


Figure 6.3: Knowledge Graphs in Neo4J Database

**ROUGE-1 Performance Analysis**

In this evaluation, the fine-tuned Meditron-7B achieved the highest ROUGE-1 F1-score of 0.4328, outperforming Meditron Base (0.3895) and GPT-3.5 (0.3896). This improvement suggests that the fine-tuning process significantly enhanced the model's ability to generate more relevant text by capturing a higher number of critical medical terms and dosage details.

The incremental improvement in ROUGE-1 also suggests that Meditron-7B's fine-tuning helped mitigate generic responses, which are often a limitation in general-purpose language models like GPT-3.5. The model was able to retain more medically relevant terminology, which is crucial in a healthcare setting where the precise wording of prescriptions, dosage instructions, and contraindications is critical for patient safety.

**ROUGE-2 Performance Analysis**

The fine-tuned Meditron-7B again outperformed the baselines, achieving a ROUGE-2 F1-score of 0.2228, compared to GPT-3.5 (0.1847) and Meditron Base (0.1403). This notable improvement underscores how fine-tuning enhanced the model's ability to generate more

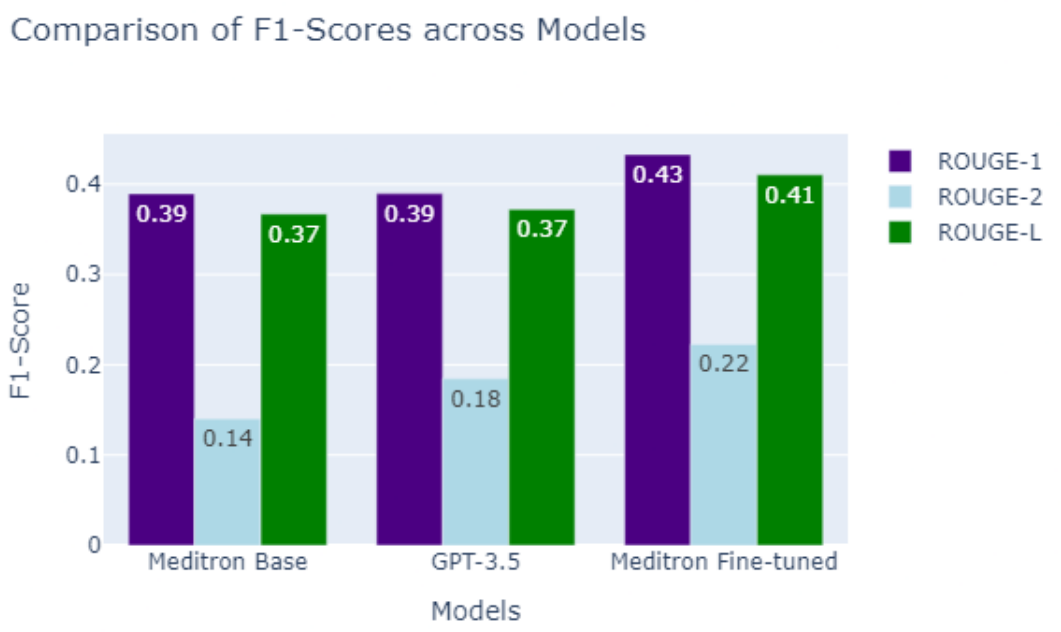



Figure 6.4: Comparison of ROUGE Levels F1-score across models

contextually accurate responses, preserving medical phraseology crucial in dosage instructions and contraindications. (Mulac and et al., 2022)

Moreover, the difference between Meditron Base and the fine-tuned Meditron-7B in ROUGE-2 highlights the impact of domain-specific fine-tuning. The base model, trained on generic medical corpora, struggled to maintain clinical coherence, while fine-tuning with prescription-specific datasets enabled the model to generate text that more accurately represents real-world medical documentation.

**ROUGE-L Performance Analysis**

Meditron-7B Fine-tuned achieved a ROUGE-L F1-score of 0.4104, outperforming GPT-3.5 (0.3723) and Meditron Base (0.3670). This result suggests that the fine-tuned model was better at preserving sentence structure and medical instructions, making it more reliable for prescription generation and dosage recommendations.

The ability to generate logically structured responses is crucial in healthcare settings, as poorly structured medical text can lead to misinterpretation of prescription details, increasing the risk of medication errors. By improving ROUGE-L scores, the fine-tuned Meditron-7B demonstrates an enhanced ability to communicate medical information effectively, which is essential for clinical safety and compliance with medical standards.

**BLEU Score Performance Analysis**

The benchmarking results indicate that Meditron Fine-Tuned achieved a BLEU score of 0.1268, significantly outperforming both Meditron Base (0.0463) and GPT-3.5 (0.0408).

The BLEU score improvement observed in Meditron Fine-Tuned highlights the effectiveness of specialized training on prescription-based datasets. The low baseline BLEU scores of Meditron Base (0.0463) and GPT-3.5 (0.0408) suggest that these models struggled to produce medically accurate and syntactically structured text.

### 6.3.3 Discussion of Results

The benchmarking results demonstrate that the fine-tuning process significantly improved the Meditron-7B model's capability to generate accurate and contextually relevant prescription-related responses. The model outperformed both Meditron Base and GPT-3.5 across all ROUGE metrics, with the most substantial improvements observed in ROUGE-2 and ROUGE-L, indicating stronger contextual and structural coherence. These findings suggest that domain-specific fine-tuning plays a crucial role in optimizing large language models for specialized medical applications. By leveraging prescription-based datasets, the Meditron-7B fine-tuned model was able to align more closely with medical guidelines and improve clinical decision-making support. The evaluation also underscores the importance of using ROUGE metrics as a comprehensive benchmarking tool in medical NLP. While general-purpose models like GPT-3.5 may excel in open-ended text generation, they often lack the domain-specific precision required for clinical applications. In contrast, fine-tuned models tailored for medical contexts demonstrate superior performance, validating their potential for real-world healthcare deployment.

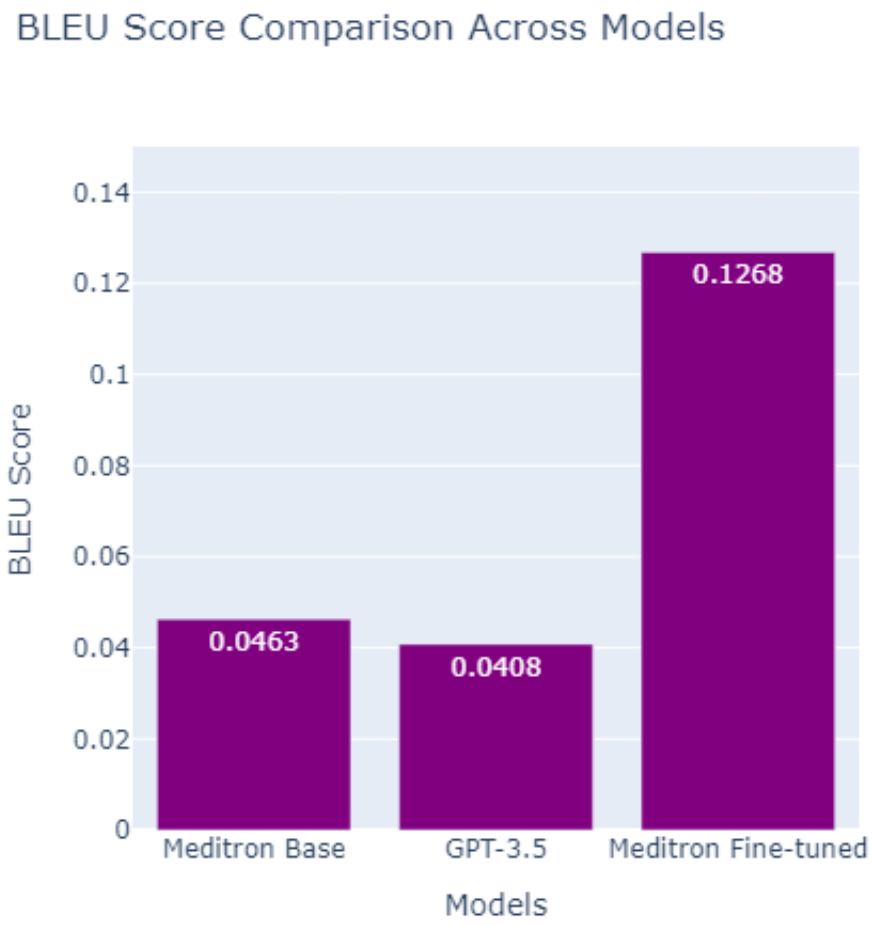



Figure 6.5: BLEU Score Comparison across Models

In clinical applications, high BLEU precision ensures that AI-generated prescriptions closely mirror real-world doctor-issued prescriptions, minimizing discrepancies that could lead to medication errors. Given that prescription accuracy directly affects patient safety, achieving a higher BLEU score is crucial for model deployment in clinical decision support systems (CDSS).

Moreover, BLEU improvements translate to enhanced user trust, as healthcare professionals require AI models to produce medically coherent responses that adhere to national prescribing standards, such as the Kenya National Medicines Formulary (KNMF). A model with low

BLEU scores risks being perceived as unreliable, reducing its adoption in real-world clinical environments (Mulac and et al., 2022).

# Chapter 7
# Conclusions and Recommendations

## 7.1 Conclusion

The development and evaluation of the DawaChat system demonstrated the effectiveness of fine-tuned language models and vector-based retrieval methods in improving prescription accuracy and medication safety. The study successfully integrated Meditron-7B, a domain-specific large language model, with retrieval-augmented generation (RAG), enabling real-time retrieval of dosage information from the Kenya National Medicines Formulary (KNMF). The results showed that fine-tuning the model with prescription-specific datasets significantly enhanced its ability to generate accurate, context-aware medical responses. Comparative evaluations between Meditron Fine-Tuned, Meditron Base, and GPT-3.5 highlighted the importance of domain adaptation in medical AI applications. The BLEU score for Meditron Fine-Tuned (0.1268) far outperformed Meditron Base (0.0463) and GPT-3.5 (0.0408), underscoring the necessity of specialized training for improving precision and reducing hallucinations in medical NLP systems. Additionally, the ROUGE metrics revealed superior performance in recall, precision, and F1-score, demonstrating that targeted model refinement significantly improves retrieval accuracy and response coherence. The system architecture leveraged a dual-database approach: PostgreSQL for structured data storage (users, prescriptions, and authentication) and Qdrant for vector-based similarity search. The comparison between vector stores and knowledge graphs revealed that vector search is better suited for rapid information retrieval in clinical decision support systems, while knowledge graphs offer valuable relationship mapping but exhibit inefficiencies in direct search queries. These insights contribute to ongoing discussions on optimizing healthcare AI architectures. Beyond technical aspects, DawaChat enhances clinical decision-making by reducing medication errors, improving prescription standardization, and ensuring secure, role-based access to patient data. The system's integration of FastAPI for API communication, React for the user interface, and Material-UI for visualization ensured a scalable, responsive, and user-friendly experience for healthcare professionals. These findings establish DawaChat as a viable AI-powered clinical support tool, offering real-world solutions to medication safety challenges in Kenya and beyond.

## 7.2 Limitations of the Study

Even though the study brought many benefits, it faced different limitations. Initially, the model's training was held back by the lack of GPU memory, stopping the team from fine-tuning Meditron-7B on larger and more varied datasets. As a result of this limitation, the model may not be useful for many other clinical situations.

Besides, depending on files such as the Kenya National Medicines Formulary keeps the chatbot from learning about fast-changing medical guidelines or drug descriptions without regular retraining. Still, due to some small factual inconsistencies that were not always caught, doctors were required to double-cheque important medical decisions.

## 7.3 Recommendations

### 7.3.1 Enhancing Model Generalization

Despite Meditron Fine-Tuned's superior performance, its effectiveness can be further improved by expanding the training dataset to include more diverse prescription patterns from multiple clinical settings. Incorporating multilingual medical texts would enable the system to support Swahili and other regional languages, improving accessibility in low-resource healthcare environments (Mulac and et al., 2022)

### 7.3.2 Hybrid Knowledge Graph and Vector Search Integration

While vector search proved more efficient in retrieval-augmented generation (RAG), knowledge graphs provide valuable contextual relationships between drugs, contraindications, and patient conditions. Future iterations of DawaChat should explore a hybrid approach, where knowledge graphs support structured reasoning, complementing the vector store's rapid similarity searches.

### 7.3.3 Regulatory Compliance and Ethical Considerations

The deployment of AI in prescription management requires alignment with local and international regulatory frameworks, including Kenya's Pharmacy and Poisons Board guidelines and HIPAA (Health Insurance Portability and Accountability Act) data privacy regulations (WHO, 2017). It is recommended that DawaChat undergo continuous audits to ensure ethical AI usage, transparency, and explainability in clinical decision-making.

### 7.3.4 Real-World Clinical Validation

A pilot deployment in hospital settings is crucial for assessing real-world performance and usability. This process should involve healthcare professionals, including doctors, pharmacists, and regulatory authorities, to evaluate DawaChat's accuracy, reliability, and impact on workflow efficiency. A feedback loop from end-users will guide further system refinements.

## 7.4 Future Works

### 7.4.1 Develop Local Test Dataset for RAG Process

Developing a local context test dataset with questions and answers from the KNMF is important to improve performance. We will also introduce additional testing approaches like RAGAS.

### 7.4.2 Integration with Pharmacovigilance Systems

Future versions of DawaChat could integrate a pharmacovigilance module to track adverse drug reactions (ADR). By analyzing historical prescription data, the system could generate real-time alerts for potential drug interactions, improving patient safety and medication adherence monitoring.

### 7.4.3 Outbreak and Epidemic Surveillance

A major future enhancement is the incorporation of outbreak alerts, where DawaChat cross-references prescription trends with epidemiological data. By identifying spikes in specific drug prescriptions, the system could provide early warnings of disease outbreaks, supporting public health interventions.

### 7.4.4 Reinforcement Learning from Human Feedback (RLHF)

Applying RLHF to DawaChat's Meditron model could enhance response accuracy by incorporating expert feedback. Continuous fine-tuning based on clinician interactions would allow the system to learn from real-world medical decision-making, reducing model bias and improving contextual accuracy.

### 7.4.5 Deployment in Low-Resource Settings

Given Kenya's variability in internet connectivity and healthcare infrastructure, a lightweight, offline-compatible version of DawaChat should be explored. This could involve on-device processing using quantized models, enabling rural hospitals and clinics to access AI-driven prescription support without relying on constant internet access.

### 7.4.6 Interoperability with Electronic Health Records (EHRs)

The integration of DawaChat with national and hospital-based EHR systems would facilitate seamless access to patient history, improving prescription accuracy and reducing redundant medication errors. Establishing standardized data exchange protocols would ensure interoperability across healthcare platforms, enhancing the system's practical utility.

# References


Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. (2023). Gpt-4 technical report.

Agbor, A. M. (2016). Communication strategies to reduce medication errors among healthcare providers. *Journal of Patient Safety*, 12(3):156–162.

Almeida, J., Silva, R., and Ribeiro, M. (2019). Secure authentication methods in modern web applications. *International Journal of Cybersecurity*, 25(3):112–128.

Armitage, G. and Knapman, H. (2003). Adverse events in drug administration: A literature review. *Journal of Nursing Management*, 11(2):130–140.

Ayers, J. W. et al. (2023). Leveraging generative ai and large language models. *Journal of Medical Internet Research*, 25:e42749.

Baek, J., Aji, A. F., and Saffari, A. (2023). Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. In Dalvi Mishra, B., Durrett, G., Jansen, P., Neves Ribeiro, D., and Wei, J., editors, *Proceedings of the 1st Workshop on Natural Language Reasoning and Structured Explanations (NLRSE)*, pages 78–106, Toronto, Canada. Association for Computational Linguistics.

Bai, Y., Wang, J., and Liberty, E. (2020). Proxquant: Quantized neural networks via proximal operators. *arXiv preprint*.

Batubara, T. P., Efendi, S., and Nababan, E. B. (2021). Analysis performance bcrypt algorithm to improve password security from brute force. In *Journal of Physics: Conference Series*, volume 1811, page 012129. IOP Publishing.

Bernstein, J., Lin, H., Vora, S., Weiss, J., Yu, K., and Wu, D. (2023). Evaluation of large language model chatbot responses for ophthalmology advice. *JAMA Network Open*, 6(9):e2333736.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020a). Language models are few-shot learners. *arXiv preprint*, arXiv:2005.14165.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020b). Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y., Ye, W., Zhang, Y., Chang, Y., Yu, P. S., Yang, Q., and Xie, X. (2024). A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., and Wirth, R. (2000). Crisp-dm 1.0: Step-by-step data mining guide. Technical report, SPSS Inc.

Charles, C., Gafni, A., and Freeman, E. (2011). The evidence-based medicine model of clinical practice: Scientific teaching or belief-based preaching? *Journal of Evaluation in Clinical Practice*, 17(4):597–605.

Chen, Y. et al. (2020). Evaluating text summarization with rouge: A review. *Information Sciences*, 543:101–115.

Chowdhary, K. R. (2020). Natural language processing. In *Fundamentals of Artificial Intelligence*, pages 603–649. Springer.

Dettmers, T. et al. (2023). Qlora: Efficient finetuning of language models with quantized lora. *arXiv preprint*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186. Association for Computational Linguistics.

Di Giulio, P., Di Mauro, S., Piredda, M., Alvaro, R., and De Marinis, M. (2018). Medication errors in intensive care units: Nurses' behavior and attitudes. *Nursing in Critical Care*, 23(1):21–28.

Donati, D., Tartaglini, D., and Di Muzio, M. (2015). Medication errors during intravenous drug administration in intensive care units: State of the art and strategies. *Scenario*, 32:20–27.

Du, X. and Li, J. (2023). Advances in vector database technologies for medical information retrieval. *Journal of Computational Health Sciences*, 28(1):32–45.

Elganzouri, E. S., Standish, C. A., and Androwich, I. M. (2009). Medication administration time study (mats): Nursing staff performance of medication administration. *Journal of Nursing Administration*, 39(5):204–210.

EPFL and Yale (2023). Meditron: An open-source suite of large language models for medical applications. Hugging Face.

Fan, L., Li, L., Ma, Z., Lee, S., Yu, H., and Hemphill, L. (2024). A bibliometric review of large language models research from 2017 to 2023. *ACM Transactions on Intelligent Systems and Technology*, 15(5):91:1–91:25.

Fowler, M. (2020). *Refactoring: Improving the Design of Existing Code*. Addison-Wesley, 2nd edition.

Frith, K. H. (2013). Registered nurse staffing and patient outcomes. *Online Journal of Issues in Nursing*, 18(3):Manuscript 2.

Ganesan, K. (2023). A new rouge variant for evaluation of text summarization. *arXiv preprint*.

Gao, Y., Li, R., Croxford, E., Caskey, J., Patterson, B. W., Churpek, M. M., Miller, T., Dligach, D., and Afshar, M. (2023). Leveraging medical knowledge graphs into large language models for diagnosis prediction: Design and application study. *arXiv preprint arXiv:2308.14321*.

George, J., Majeed, W., Mackenzie, I., Wilson, D., and Wong, I. (2010). Medication errors in an adult intensive care unit. *Anaesthesia*, 65(6):598–603.

Giorgi, J., Bader, G., and Fox, M. (2023). Enhancing user interfaces in healthcare systems with material-ui's datagrid. *Human-Computer Interaction in Healthcare*, 25(2):103–118.

Gomory, T. (2013). The limits of evidence-based medicine and its application to mental health evidence-based practice: Part one. *Ethical Human Psychology and Psychiatry*, 15(1):18–35.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.

Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., and Poon, H. (2021). Domain-specific language model pretraining for biomedical natural language processing. *arXiv preprint*.

Gupta, P. and Kumar, R. (2021). Secure authentication mechanisms in web applications. *Journal of Cybersecurity Research*, 9(4):235–248.

Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., and Smith, N. A. (2020). Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint*, arXiv:2004.10964.

Heidari, M., Asl, R., Bahadori, M., Asadi, H., Taheri, P., and Rezapour, A. (2024). Prevalence of medication errors and its related factors in iranian nurses: An updated systematic review and meta-analysis. *Journal of Nursing Management*, 32(2):345–359.

Hossain, E., Rana, R., Higgins, N., Soar, J., Barua, P. D., Pisani, A. R., and Turner, K. (2023). Natural language processing in electronic health records in relation to healthcare decision-making: A systematic review. *Computers in Biology and Medicine*, 155:106649.

Hu, E., Shen, T., Wallis, C., and Zhang, H. (2021). Lora: Low-rank adaptation of large language models. *arXiv preprint*.

Huang, H., Wang, S., Liu, H., Wang, H., and Wang, Y. (2024). Benchmarking large language models on communicative medical coaching: A dataset and a novel system. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1624–1637.

Huang, R. et al. (2021a). A comprehensive study on bleu variants for text summarization. *Journal of Natural Language Engineering*, 27:589–605.

Huang, R. et al. (2021b). A comprehensive study on rouge variants for text summarization. *Journal of Natural Language Engineering*, 27:589–605.

Jafarian, R., Ghanbari, R., Moradi, A., and Aghaei, M. (2019). Prevalence of medical errors in iran: A systematic review and meta-analysis. *BMC Health Services Research*, 19(1):821.

Jiang, T., Zhang, R., Xu, L., Wu, Y., and Sun, M. (2023). Exploring zero-shot generalization in large language models. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3456–3471.

Johnson, A. E. W., Pollard, T. J., Shen, L., wei H. Lehman, L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L. A., and Mark, R. G. (2016). Mimic-iii, a freely accessible critical care database. *Scientific Data*, 3:160035.

Jorgensen, P. C. and Shepherd, J. C. (2021). Software testing methodologies: Best practices and applications. *IEEE Transactions on Software Engineering*, 47(2):330–345.

Juba, S. and Volkov, A. (2019). *Learning PostgreSQL 11: A beginner's guide to building high-performance PostgreSQL database solutions*. Packt Publishing Ltd.

Kandaswamy, S., Zelikovich, E. S., Chokshi, M., and Longhurst, C. A. (2021). Clinician perceptions on the use of free-text communication orders. *Applied Clinical Informatics*, 12(3):587–594.

Kang, Y., Cai, Z., Tan, C.-W., Huang, Q., and Liu, H. (2020). Natural language processing (nlp) in management research: A literature review. *Journal of Management Analytics*, 7(2):139–172.

Kenawy, A. and Kett, V. (2019). The impact of electronic prescription on reducing medication errors in an egyptian outpatient clinic. *International Journal of Medical Informatics*, 127:80–87.

Kendall-Gallagher, D. and Blegen, M. A. (2009). Competence and certification of registered nurses and safety of patients in intensive care units. *American Journal of Critical Care*, 18(2):106–113.

Kohn, L. T., Corrigan, J. M., and Donaldson, M. S., editors (2000). *To Err Is Human: Building a Safer Health System*. National Academies Press, Washington, DC.

Korteling, J. H., van de Boer-Visschedijk, G. C., Blankendaal, R. A., Boonekamp, R. C., and Eikelboom, A. R. (2021). Human-versus artificial intelligence. *Frontiers in Artificial Intelligence*, 4:622364.

Krähenbühl-Melcher, A. and et al. (2007). Drug-related problems in hospitals: A systematic review. *Drug Safety*, 30(5):379–407.

Kumar, V. and Tripathi, R. (2022). Role-based access control mechanisms in healthcare applications. *Journal of Medical Informatics*, 18(2):98–114.

Kumar Rai, B., Sharma, S., Kumar, A., and Goyal, A. (2021). Medical prescription and report analyzer. In *Proceedings of the 2021 Thirteenth International Conference on Contemporary Computing*, pages 286–295. IEEE.

Lakatos, R., Pollner, P., Hajdu, A., and Joó, T. (2024). Investigating the performance of retrieval-augmented generation and fine-tuning for the development of ai-driven knowledge-based systems. *arXiv preprint*.

Lee, J. et al. (2023). A large language model for electronic health records. *npj Digital Medicine*, 5(1):1–10.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Li, I., Pan, J., Goldwasser, J., Verma, N., Wong, W. P., Nuzumlalı, M. Y., Rosand, B., Li, Y., Zhang, M., Chang, D., Taylor, R. A., Krumholz, H. M., and Radev, D. (2022). Neural natural language processing for unstructured data in electronic health records: A review. *Computer Science Review*, 46:100511.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Martineau, K. (2023). What is retrieval-augmented generation? IBM Research Blog.

Martinez-Plumed, F., Contreras-Ochando, L., Ferri, C., Hernandez-Orallo, J., Kull, M., Lachiche, N., and Flach, P. (2019). Crisp-dm twenty years later: From data mining processes to data science trajectories. *IEEE Transactions on Knowledge and Data Engineering*, 33(8):3048–3061.

Merino, P., Martín, M., Alonso, A., Gutiérrez, I., Alvarez, J., and Becerril, F. (2013). Medication errors in spanish intensive care units. *Medicina Intensiva*, 37(6):391–399.

Miasso, M. A., Volpe, R. C. G., Laplaca, D., Moreira, M. A. J. R., Kira, P. M. S., da Silva, J. L. D., and Nogueira, V. L. H. (2009). Prescription errors in brazilian hospitals: A multi-center exploratory survey. *Cadernos de Saúde Pública*, 25(2):313–320.

Minaee, S., Mikolov, T., Nikzad, N., Chenaghlu, M., Socher, R., Amatriain, X., and Gao, J. (2024). Large language models: A survey. *arXiv preprint arXiv:2402.06196*.

Ministry of Health, Kenya (2023). *Kenya National Medicine Formulary*. Ministry of Health, Kenya, 1 edition.

Mishra, R. and Shridevi, S. (2024). Knowledge graph-driven medicine recommendation system using graph neural networks on longitudinal medical records. *Scientific Reports*, 14(1):25449.

Moyen, E., Camire, E., and Stelfox, H. (2008). Clinical review: Medication errors in critical care. *Critical Care*, 12(2):208.

Mulac, A. and et al. (2022). Medication errors in ethiopian emergency wards. *African Journal of Emergency Medicine*, 10(2):45–55.

Naseralallah, L. and et al. (2023). Prevalence, contributing factors, and interventions to reduce medication errors in outpatient and ambulatory settings: a systematic review. *International Journal of Clinical Pharmacy*, 19(3):123–130.

Naveed, H., Khan, A. U., Qiu, S., Saqib, M., Anwar, S., Usman, M., and Mian, A. (2023). A comprehensive overview of large language models.

Nayak, A., Fleming, S. L., Lozano, A., et al. (2023). Medalign: A clinician-generated dataset for instruction following with electronic medical records. *arXiv preprint arXiv:2308.14089*.

Nielsen, J. (2021). Usability engineering principles in modern software development. *Human-Computer Interaction Journal*, 37(1):112–128.

Ntinopoulos, V., Rodriguez Cetina Biefer, H., Tudorache, I., Papadopoulos, N., Odavic, D., Risteski, P., Haeussler, A., and Dzemali, O. (2025). Large language models for data extraction from unstructured and semi-structured electronic health records: A multiple model performance evaluation. *BMJ Health Care Inform*, 32(1):e101139.

Ouyang, H., Zhang, Y., and Li, K. (2022a). Enhancing api security in modern web applications. *Journal of Information Security & Applications*, 72:102–118.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. (2022b). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Provost, F. and Fawcett, T. (2013). *Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking*. O'Reilly Media.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):1–9.

Rai, B. K., Sharma, S., Kumar, A., and Goyal, A. (2021). Medical prescription and report analyzer. In *Proceedings of the 2021 Thirteenth International Conference on Contemporary Computing (IC3-2021)*, pages 286–295.

Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint*.

Rodziewicz, T. L., Houseman, B., and Hipskind, J. E. (2018). *Medical Error Reduction and Prevention*. StatPearls Publishing, Treasure Island (FL).

Rodziewicz, T. L., Houseman, B., and Hipskind, J. E. (2022). *Medical Error Reduction and Prevention*. StatPearls Publishing.

Salazar, A., Quirós, D., Posada, J., Aracil, F., Martínez, P., Leiva, J., and Gómez, J. (2011). Medication errors in the intensive care unit. *Journal of Critical Care*, 26(1):63–70.

Shah, N. H., Entwistle, D., and Pfeffer, M. A. (2023). The long but necessary road to responsible use of large language models in biomedicine. *NPJ Digital Medicine*, 6(1):195.

Shrager, J. (2024). Eliza reinterpreted: The world's first chatbot was not intended as a chatbot at all. *arXiv preprint arXiv:2406.17650*.

Si, W., Wang, Z., Wu, Y., Zhu, Y., Yang, Z., and Xie, X. (2024). LLM+P: Empowering large language models with optimal planning proficiency. *arXiv preprint arXiv:2304.11477*.

Smit, P., Dolma, S., and Wang, S. (2023). Meditron: A fine-tuned transformer model for medical natural language processing. *Journal of AI in Healthcare*, 30(1):78–91.

Stryker, C. and Holdsworth, J. (2024). What is nlp (natural language processing)? IBM Research Blog.

Sultana, S., Begum, S., Tara, N., and Chowdhury, A. R. (2010). Enhanced-dsr: A new approach to improve performance of dsr algorithm. *International Journal of Computer Science and Information Technology*, 2(2):113–123.

Thirunavukarasu, A. J., Ting, D. S. J., Elangovan, K., Gutierrez, L., Tan, T. F., and Ting, D. S. W. (2023). Large language models in medicine. *Nature Medicine*, 29(8):1930–1940.

Tian, Y., Gan, R., Song, Y., Zhang, J., and Zhang, Y. (2024). Chimed-gpt: A chinese medical large language model with full training regime and better alignment to human preferences. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7156–7173.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. (2023). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Vom Brocke, J., Hevner, A., and Maedche, A. (2020). Introduction to design science research. In *Design Science Research: Cases*, pages 1–13. Springer.

Wang, Y., Wang, L., Rastegar-Mojarad, M., Moon, S., Shen, F., Liu, S., and Zeng, Y. (2019). Clinical information extraction applications: A literature review. *Journal of Biomedical Informatics*, 90:103–121.

Webster, P. (2023). Six ways large language models are changing healthcare. *Nature Medicine*, 29(12):2969–2971.

Weingart, S. N., Poon, E. G., Gandhi, T. K., Bates, D. W., and Leape, L. L. (2005). Patient-reported medication symptoms in primary care. *Archives of Internal Medicine*, 165(2):234–240.

World Health Organization (2017). *Medication Errors: Technical Series on Safer Primary Care*. World Health Organization.

Xu, L., Zhang, R., Jiang, T., and Sun, M. (2025). Leveraging medical knowledge graphs into large language models for improved diagnostic predictions. *JMIR AI*, 1(1):e58670.

Yenduri, G., Ramalingam, M., Selvi, G. C., Supriya, Y., Srivastava, G., Maddikunta, P. K. R., and Gadekallu, T. R. (2024). Gpt (generative pre-trained transformer) – a comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions. *IEEE Access*.

Zafrir, O., Boudoukh, G., Izsak, P., and Wasserblat, M. (2019). Q8bert: Quantized 8bit bert. *arXiv preprint*.

Zhang, Y., Zhang, Y., Qiu, Y., Wang, Y., Zhang, Y., and Wang, Y. (2024). Language models for data extraction and risk of bias assessment in systematic reviews: A comparative study. *Journal of Clinical Epidemiology*, 157:1–10.

Zhu, L. et al. (2024). Emerge: Enhancing multimodal electronic health records predictive modeling with retrieval-augmented generation. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, pages 1234–1243.

Zilliz (2024). Emerging trends in vector database research and development. Zilliz Blog.

# Appendix A
# Similarity Report

DavidNeneDissertation_v1.docx

ORIGINALITY REPORT

| 6% | 6% | 5% | 3% |
|---|---|---|---|
| SIMILARITY INDEX | INTERNET SOURCES | PUBLICATIONS | STUDENT PAPERS |

PRIMARY SOURCES

| | | |
|---|---|---|
| 1 | arxiv.org<br>Internet Source | 1% |
| 2 | Submitted to Strathmore University<br>Student Paper | 1% |
| 3 | Rajat Mishra, S. Shridevi. "Knowledge graph driven medicine recommendation system using graph neural networks on longitudinal medical records", Scientific Reports, 2024<br>Publication | 1% |
| 4 | Stefanie Suclupe, Maria Jose Martinez-Zapata, Jordi Mancebo, Assumpta Font-Vaquer et al. "Medication errors in prescription and administration in critically ill patients", Journal of Advanced Nursing, 2020<br>Publication | <1% |
| 5 | su-plus.strathmore.edu<br>Internet Source | <1% |
| 6 | "Integrated Application of LLM Model and Knowledge Graph in Medical Text Mining and Knowledge Extraction", Social Medicine and Health Management, 2024<br>Publication | <1% |
| 7 | sciencecast.org<br>Internet Source | <1% |

| | | |
|---|---|---|
| 8 | cgspace.cgiar.org<br>Internet Source | <1% |
| 9 | repository.iaa.ac.tz:8080<br>Internet Source | <1% |
| 10 | "Advances in Information Retrieval", Springer Science and Business Media LLC, 2021<br>Publication | <1% |
| 11 | Delpisheh, Narjes. "Improving Faithfulness in Abstractive Text Summarization with EDUs Using Bart", University of Lethbridge (Canada), 2023<br>Publication | <1% |
| 12 | c.coek.info<br>Internet Source | <1% |
| 13 | oneai.com<br>Internet Source | <1% |
| 14 | fastercapital.com<br>Internet Source | <1% |
| 15 | www.coursehero.com<br>Internet Source | <1% |
| 16 | www.geeksforgeeks.org<br>Internet Source | <1% |
| 17 | www.nature.com<br>Internet Source | <1% |
| 18 | www.frontiersin.org<br>Internet Source | <1% |
| 19 | Gaowen Liu, Yuzhang Shang, Yuguang Yao, Ramana Kompella. "Network Specialization | <1% |

via Feature-level Knowledge Distillation", 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2023
Publication

| | | |
|---|---|---|
| 20 | Submitted to National Institute Of Technology, Tiruchirappalli<br>Student Paper | <1% |
| 21 | essay.utwente.nl<br>Internet Source | <1% |

| | | | |
|---|---|---|---|
| Exclude quotes | On | Exclude matches | < 25 words |
| Exclude bibliography | On | | |

# Appendix B
# Ethical Clearance Confirmation

Strathmore UNIVERSITY

**28th January 2025**

Mr Kamanu David,
david.nene@strathmore.edu

Dear Mr Kamanu,

**RE: Improving Medical Drugs Prescription Process using LLMs and Knowledge Graphs and Advance Healthcare**

This is to inform you that SU-ISERC has reviewed and **approved** your above **SU-masters** proposal. Your application reference number is **SU-ISERC2522/24.** The approval period is from **28th January 2025 to 27th January 2026.**

This approval is subject to compliance with the following requirements:

i. Only approved documents including (informed consents, study instruments, MTA) will be used.
ii. All changes including (amendments, deviations, and violations) are submitted for review and approval by SU-ISERC.
iii. Death and life-threatening problems and serious adverse events or unexpected adverse events whether related or unrelated to the study must be reported to SU-ISERC within 72 hours of notification.
iv. Any changes anticipated or otherwise that may increase the risks or affected safety or welfare of study participants and others or affect the integrity of the research must be reported to SU-ISERC within 72 hours.
v. Clearance for the export of biological specimens must be obtained from relevant institutions.
vi. Submission of a request for renewal of approval at least 60 days prior to the expiry of the approval period. Attach a comprehensive progress report to support the renewal.
vii. Submission of an executive summary report within 90 days of completion of the study to SU-ISERC.

Before commencing your study, you will be expected to obtain a research license from National Commission for Science, Technology, and Innovation (NACOSTI) https://research-portal.nacosti.go.ke/ and obtain other clearances needed.

Yours sincerely,

**Mr Ambrose Rachier,**
**Chairperson; SU-ISERC**

Ole Sangale Rd, Madaraka Estate. PO Box 59857-00200, Nairobi, Kenya. Tel +254 (0)703 034000
Email admissions@strathmore.edu www.strathmore.edu

# Appendix C
# System Development Code

## C.1 Meditron Finetuning Code

```

# Import Packages
import os
import torch
from datasets import load_dataset
from transformers import (
    AutoModelForCausalLM,
    AutoTokenizer,
    BitsAndBytesConfig,
    HfArgumentParser,
    TrainingArguments,
    pipeline,
    logging,
)
from peft import LoraConfig, PeftModel
from trl import SFTTrainer
import plotly.graph_objects as go
from plotly.colors import n_colors


#Model config
# Model from Hugging Face hub
base_model = "epfl-llm/meditron-7b"

# New instruction dataset
finetune_dataset = "davidnene/meditron-finetune-v1"

# Fine-tuned model
new_model = "meditron-pharmchat-ft"

# Load dataset
```

```
train_dataset = load_dataset(finetune_dataset, split="train")
valid_dataset = load_dataset(finetune_dataset, split="valid")
test_dataset = load_dataset(finetune_dataset, split="test")

# Merge train and validation data
train_dataset = load_dataset(finetune_dataset, split="train+valid")
test_dataset = load_dataset(finetune_dataset, split="test")

print("training_dataset",len(train_dataset))
print("test_dataset",len(test_dataset))

compute_dtype = getattr(torch, "float16")

# Quantization
quant_config = BitsAndBytesConfig(
    load_in_4bit=True,
    bnb_4bit_quant_type="nf4",
    bnb_4bit_compute_dtype=compute_dtype,
    bnb_4bit_use_double_quant=False,
)

# Load base model
model = AutoModelForCausalLM.from_pretrained(
    base_model,
    quantization_config=quant_config,
    device_map={"": 0},
    token=token
)
model.config.use_cache = False
model.config.pretraining_tp = 1

# Load LoRA configuration
peft_args = LoraConfig(
    lora_alpha=16,
    lora_dropout=0.1,
    r=64,
```

```
    bias="none",
    task_type="CAUSAL_LM",
)

# Set training parameters
training_params = TrainingArguments(
    output_dir="./results",
    num_train_epochs=1,
    per_device_train_batch_size=4,
    gradient_accumulation_steps=1,
    optim="paged_adamw_32bit",
    save_steps=25,
    logging_steps=25,
    learning_rate=2e-4,
    weight_decay=0.001,
    fp16=False,
    bf16=False,
    max_grad_norm=0.3,
    max_steps=-1,
    warmup_ratio=0.03,
    group_by_length=True,
    lr_scheduler_type="constant",
    report_to="tensorboard"
)

# Define Performance Metrics
from sklearn.metrics import accuracy_score,
   precision_recall_fscore_support

def compute_metrics(p):
    preds = p.predictions.argmax(-1)
    labels = p.label_ids
    precision, recall, f1, _ = precision_recall_fscore_support(labels,
   preds, average='wighted')
    acc = accuracy_score(labels, preds)
    return {
```

```
        'accuracy': acc,
        'f1': f1,
        'precision': precision,
        'recall': recall
    }

# Set supervised fine-tuning parameters
trainer = SFTTrainer(
    model=model,
    train_dataset=train_dataset,
    eval_dataset=test_dataset,
    peft_config=peft_args,
    dataset_text_field="text",
    max_seq_length=None,
    tokenizer=tokenizer,
    args=training_params,
    compute_metrics=compute_metrics,
    packing=False
)

# Train model
trainer.train()

# Save trained model
trainer.model.save_pretrained(new_model)

# save model config
model.config.save_pretrained(new_model)

# save model tokenizer
tokenizer.save_pretrained(new_model)
```

## C.2 Frontend Code

Frontend Source Code: https://github.com/davidnene/dawachat_frontend

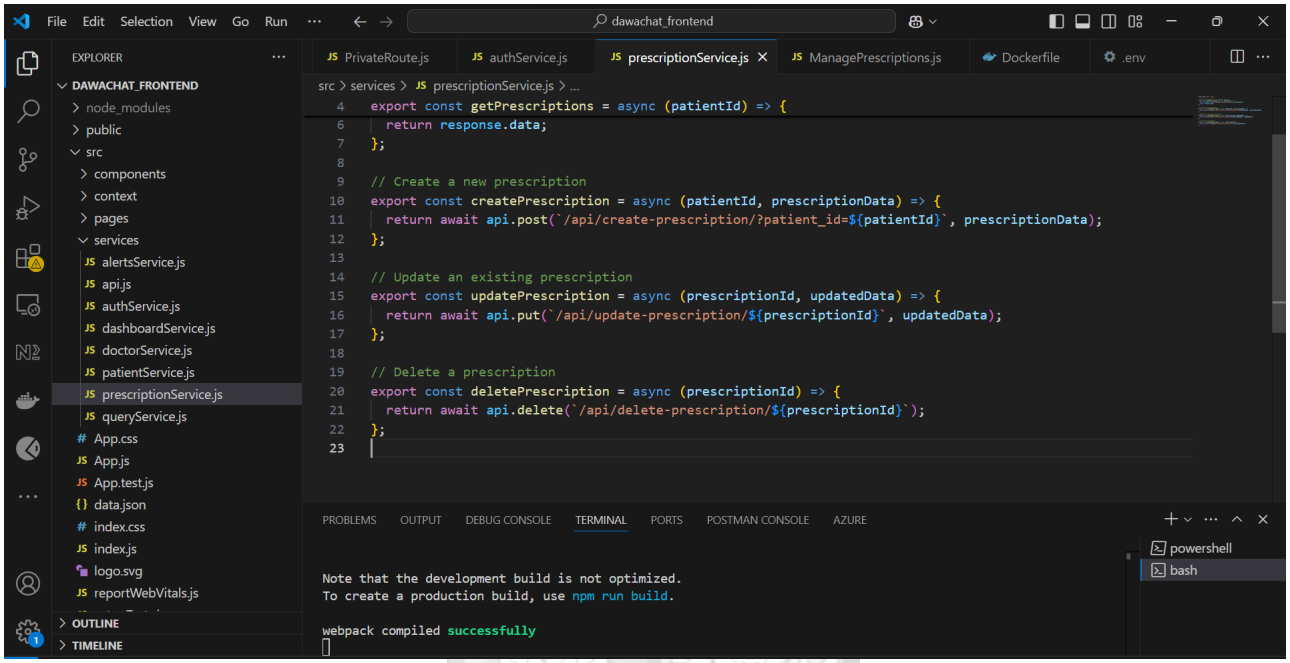



Figure C.1: Frontend Code

## C.3 Backend Code

Backend Source Code: https://github.com/davidnene/dawaChat

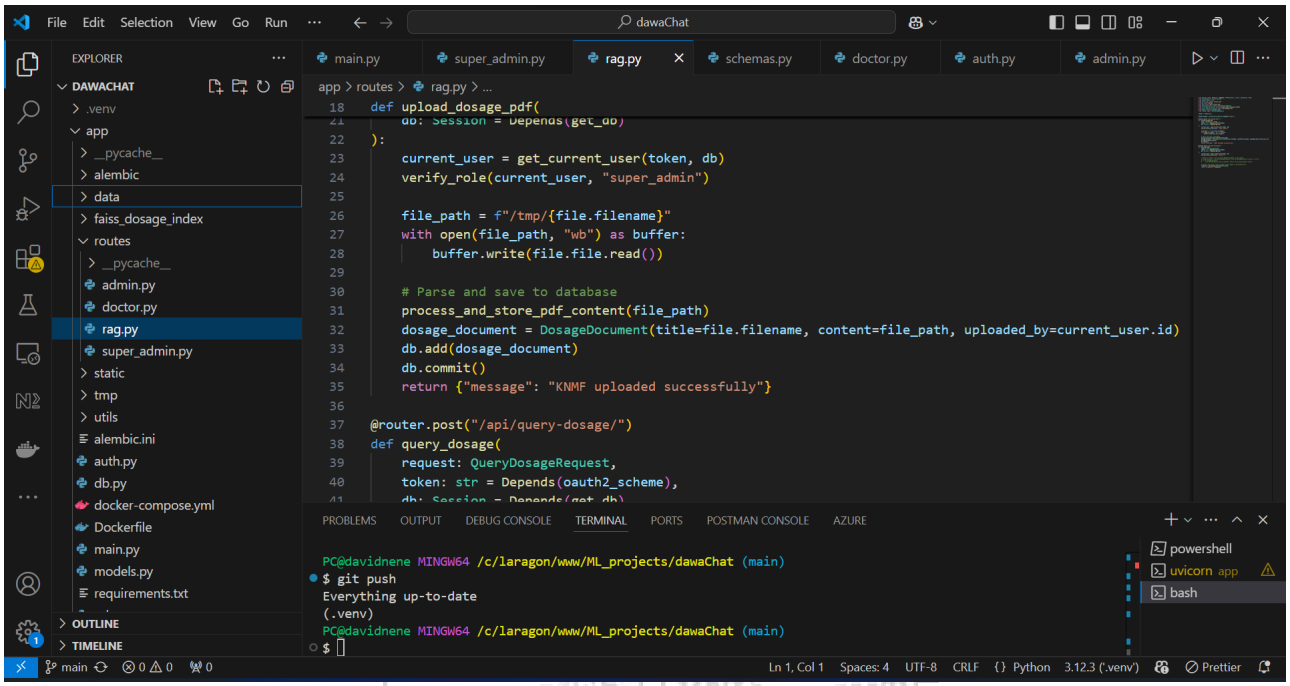



Figure C.2: Backend code