

# DESIGN AND IMPLEMENTATION OF A MEDICATION-AWARE AI HEALTH ASSISTANT FOR MOBILE HEALTH APPLICATIONS

<sup>1</sup>Dr. K. Soumya, <sup>2</sup>Piniseti Hema Kalyani, <sup>3</sup>Pithani Hema Latha, <sup>4</sup>Poddoku Snehitha, <sup>5</sup>Potnuru Lakshmi Sowmya Deepika, <sup>6</sup>Dr. Pallam Setti

<sup>1</sup>Assistant Professor, <sup>2</sup>Student, <sup>3</sup>Student, <sup>4</sup>Student, <sup>5</sup>Student, <sup>6</sup>Founder & CEO

<sup>1,2,3,4,5</sup>Department of Information Technology and Computer Applications, <sup>6</sup>Dr. Pallam Setti Centre For Research & Technology (A-hub)

<sup>1,2,3,4,5</sup>Andhra University College of Engineering for Women, <sup>6</sup>Dr. Pallam Setti Centre For Research & Technology (A-hub), Visakhapatnam, India.

**Abstract :** Medication non-adherence and self-diagnosis remain one of the major problems for people who are suffering from chronic or recurring diseases. Though in recent times, the number of available AI-based healthcare tools through mobile technology has increased, it is found that the available tools respond to healthcare questions without any reference to the pharmacotherapy regimen of the users, which may result in giving irrelevant and unsafe recommendations. This paper presents the development and implementation of the proposed MediPro system, which is a healthcare application that incorporates the concept of medication context injection and a healthcare assistant based on a large language model. The proposed system is capable of giving responses to healthcare questions in a more accurate manner by injecting the information regarding the medications that the users are taking, like the name of the medicine, dose, and frequency, into the prompt for the AI system. The system is implemented using React, Typescript, Tailwind CSS, and Cohere command r plus API with Supabase as the authentication and storage system. The performance of the system with different healthcare query scenarios proves that the proposed system is capable of giving responses that are more relevant and safe compared to the responses given by the context-free AI system. The proposed system is also capable of operating independently from the hospital records system and can be used as a normal web application. The feature-wise comparison of the proposed system with other available healthcare applications proves that there is no other available system that is capable of giving guidance through an AI system, checking drug interactions, calculating ADMET, and enforcing clinical safety at the prompt level within a single system that is independent from the Electronic Health Records system.

**Index Terms** - mobile health application, AI health assistant, medication context injection, large language model, Cohere API, prompt engineering, personalized health guidance, ADMET scoring, Supabase

## 1. INTRODUCTION

Personal health management represents an important challenge in the present healthcare landscape. According to the World Health Organization, 50% of patients suffering from chronic health problems fail to follow the prescribed medication, which in turn triggers unnecessary complications and increases the treatment costs [9]. As the use of the smartphone in the healthcare industry continues to rise across the globe, mobile health applications have proven to be an effective solution in improving medication adherence and health awareness [3]. In the Indian healthcare industry, the need for mobile health applications is higher, given the fact that the population faces chronic health problems such as diabetes, hypertension, and gastrointestinal diseases, coupled with the lack of access to routine medical consultations.

The current state of the applications is that they are generally single-function applications, such as pill reminders, drug encyclopedias, or symptom checkers, that operate independently without any regard for the individual's overall health picture. AI-powered health chatbots have become increasingly popular with the advent of large language models; however, these applications operate in a generic fashion, responding to questions without taking into account the individual's current medication profile, as discussed in [8][7]. An illustration of this is the individual with a headache inquiring about the appropriateness of taking ibuprofen, receiving the same response from an AI chatbot regardless of whether they are taking anticoagulants, proton pump inhibitors, or taking no medications at all.

The rapid evolution of large language models, including GPT-4, Google Gemini, and Cohere command r plus, has opened the doors to the possibility of AI-powered health guidance at the consumer level. These applications have shown tremendous capability in health-related question answering and medication information retrieval, as discussed in [2]; however, the current state of the applications does not incorporate the individual's personal medication profile as input before responding, with the most advanced LLM providing the same generic response for an individual with diabetes taking Metformin as they would if the individual were a healthy individual asking the same question.

This paper attempts to bridge this gap by proposing MediPro, a mobile health management application that incorporates an injection of medication context into an AI health assistant using a large language model. The paper aims to demonstrate that by dynamically loading a user's previously saved medication profile into the system prompt of the Cohere command-r-plus model prior to each query, more relevant and personalized, and safer responses will be generated by an AI health assistant than those generated by an AI health assistant without medication context. To the best of our knowledge, this formal implementation and evaluation of this prompt-level, dynamic medication context injection for personalization of an LLM-based health assistant has not been explored or implemented as part of a consumer-facing mobile health application.

The remainder of this paper is organized as follows. Section 2 presents a structured literature survey across six representative prior works and identifies the specific research gaps that MediPro addresses. Section 3 describes the system architecture in detail, covering the medication context injection mechanism, conversation history management, clinical safety constraint design, technology stack, and security architecture. Section 4 presents the evaluation results through four side-by-side query comparisons and a comprehensive feature-level comparison against existing solutions. Section 5 concludes the paper with a summary of key contributions, a candid discussion of current limitations, and a roadmap for future development.

## 2. LITERATURE SURVEY

Sr. No.	Author / Year	Title / Idea	Limitations Identified	How Our Project Overcomes It
1	Ventola C L, 2014 [8]	A survey of mobile health apps for healthcare professionals indicates that the most important applications of consumer mHealth apps are related to tracking medication and symptoms.	These apps are created as single-function tools, with no inter-feature understanding, and they provide no form of AI-based or personalized guidance based on the user's medication history.	MediPro integrates medication tracking, symptom analysis, reminders, and the AI assistant within one platform, with each module centered on the user's medication context.
2	Topol E J, 2019 [7]	A case for democratizing AI in healthcare through mobile devices indicates that deep learning models can facilitate clinical decision-making at the level of individual patients.	There is much emphasis on imaging and diagnostic AI, but no emphasis on conversational AI or providing the general user with medication-based, personalized guidance.	We are extending the concept of AI democratization to the domain of conversational health guidance by wrapping a lightweight large language model within a browser-based experience accessible by any mobile browser.
3	Brown T B et al., 2020 [2]	An introduction to GPT-3 indicates that LLMs can answer health-related queries with clinically sensible responses, thus proving the viability of LLMs for health Q&A.	There are no contextual responses, as the system does not know the user asking the question, the user's medications, the user's health conditions, or the user's health history.	Our work improves upon the generic large language model by injecting the user's medication context into each system prompt to provide more relevant health guidance.
4	Bates D W et al., 2021 [1]	A review of AI-powered clinical decision support for consumer health apps indicates that personalized AI messages can improve medication adherence.	It must be fully integrated with the electronic health record system and make use of hospital-grade infrastructure; therefore, it cannot be a consumer-level mobile app.	Our system relies on the user's self-reported medication list stored within Supabase as the input to provide more personalized responses.
5	Magrabi F et al., 2019 [5]	A systematic review of risks in consumer health apps indicates that the absence of drug interaction checks and clinical safety constraints can cause harm to patients.	The study is descriptive in nature and does not propose or implement a safety framework, as the apps studied lack structured medication data or AI-level safety enforcement.	We also implement a built-in drug interaction checker, ADMET-scored recommendations, and prompt-level AI constraints to disallow diagnosis and provide disclaimers.
6	Liu P et al., 2023 [4]	An exploration of strategies to improve LLMs through prompt engineering indicates that system prompts with contextual information improve the relevance of LLMs.	The study is conducted in controlled environments within the academic institution, but there is no demonstrated or evaluated implementation in the consumer mobile health application space.	This work showcases the practical application of prompt engineering within a real-world mobile health app with actual user prompts and medication context-informed response comparisons.

## Research Gap

A synthesis of the six works reviewed above reveals three converging limitations that collectively motivate the MediPro system design. First, while large language models have demonstrated strong performance on health question-answering tasks, none of the reviewed works implement a dynamic, prompt-level injection of user-specific medication context into a consumer-facing LLM health assistant. The models studied respond identically to the same query regardless of the user’s pharmacological profile, making their outputs clinically generic rather than personally actionable.

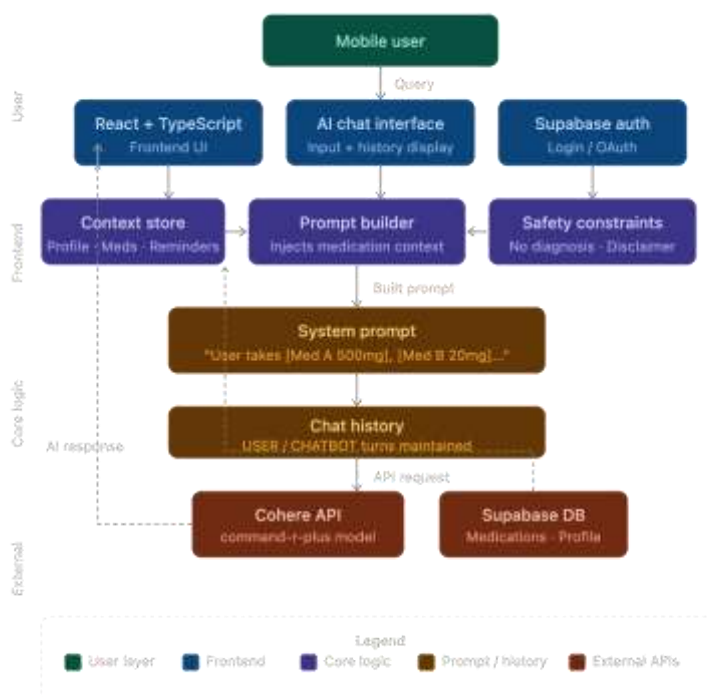
Second, existing consumer mHealth applications remain fragmented by design. Medication trackers, symptom checkers, and AI chatbots are built and deployed as separate tools with no shared awareness of the user’s health context. No reviewed system integrates these functions within a single platform where the AI assistant has live access to the user’s medication list.

Third, the safety dimension of AI-generated health guidance is either absent or limited to post-hoc disclaimers in all reviewed works. None of the reviewed systems embed safety constraints at the prompt level in a way that structurally prevents the model from producing diagnostic or prescriptive outputs. MediPro directly addresses all three gaps by implementing a stateless, per-request medication context injection framework within a unified mHealth platform that enforces clinical safety constraints at the architectural level.

## 3. MATERIALS AND METHODS

### 3.1 System Overview

MediPro has a modular, context-driven client-side application structure. The application has three main context providers: ProfileContext, MedicationsContext, and RemindersContext, all using React Context. These three contexts carry the global state of the application and store data both locally in the browser's local storage as well as on Supabase's cloud storage. All screens are connected to all contexts, ensuring that user information is always consistent across all modules. The application uses React Router DOM v6 on the client-side for routing, with a shared Mobile Layout component that provides a standard bottom navigation bar on all authenticated screens. The overall application structure is shown in Figure 1.



**Figure 1. System Architecture of MediPro --- Medication-Aware AI Health Assistant**

The interaction between the three context providers and the AI Health Assistant module is central to MediPro’s design. When a user submits a health query through the AI Assistant interface, the component simultaneously reads from MedicationsContext to retrieve the current medication list and from ProfileContext to retrieve any relevant personal health parameters. These values are passed directly into the buildSystemPrompt() function, which constructs the structured system prompt for that specific request.

This means the AI module never stores medication data internally. It reads from the shared context on every render cycle, ensuring that any change the user makes to their medication list in the Medication Manager module is automatically reflected in the very next AI query without requiring a session restart or manual refresh. This live context linkage is what distinguishes MediPro from systems where user data and AI functionality operate as separate, loosely coupled modules.

### 3.2 Medication-Context Injection and Prompt Construction

The main contribution is the medication context injection framework. Before the user query is sent to the Cohere API, a structured system prompt is constructed by the `buildSystemPrompt()` method in the React component. This system prompt is rebuilt before every API call, ensuring that the user's medication list is taken into account by the AI response without the need for the user to restart the session.

The system prompt instructs the assistant to function as an AI health assistant embedded in MediPro, to take into consideration the user's full list of saved medications, including name, dosage, and frequency, from MedicationsContext, and to follow a list of rules to function correctly. These rules include keeping the response concise and mobile-friendly, keeping the response free of

technical jargon to make sense to non-medical users, considering the user's saved medications when answering health-related questions, avoiding giving diagnoses, and always including a short medical disclaimer when providing clinical-level responses. In the absence of saved medications, the word "none" is used to allow the assistant to function as a general health assistant.

The rationale for each constraint deserves explicit consideration. The prohibition on jargon exists because MediPro's target users are patients, not clinicians. A response containing terms such as CYP3A4 inhibition or pharmacokinetic interaction without plain-language explanation is functionally equivalent to no response for a lay user.

The medication cross-referencing rule is the most critical constraint in the injection framework. Without it, the model defaults to generic population-level guidance, which is precisely the failure mode MediPro is designed to eliminate. The diagnosis prohibition exists because an LLM operating on self-reported symptoms without physical examination, laboratory results, or patient history cannot safely make diagnostic determinations — this constraint prevents the model from overstepping its appropriate clinical role.

The medical disclaimer requirement ensures that users are never left with the impression that MediPro's output substitutes for professional medical advice, regardless of how confident or specific the model's response appears. Together, these constraints define a safe operating envelope within which the model can be genuinely useful without becoming clinically dangerous.

### 3.3 Conversation History Management and Response Handling

The multi-step conversations remain smooth by passing the entire chat history with each request to the Cohere API using the chat\_history field. Each message is identified with a role, either user or chatbot, and has a message with the text content. The initial welcoming message is not included in the API request as it is simply a UI placeholder. The text generated by the Cohere API is extracted from the JSON response, and a new message is added as part of the assistant in the conversation state when the API returns its response. In case of API failure, a user-friendly error message is displayed instead of API errors.

### 3.4 Clinical Safety Constraints

Clinical safety is embedded in each query from the beginning through clear guidelines that are embedded in each API call. The assistant is instructed not to make a diagnosis, not to recommend prescription medication without encouraging professional help, and to always include a medical disclaimer with any answer that is given on health topics. These guidelines are part of the instruction set that is always available to the model, so they remain in effect regardless of how a question is phrased. On the Symptom Analyzer page, a safety disclaimer is always prominently displayed, as is an Emergency Call 108 button.

### 3.5 Technology Stack

*Table 1. Technology Stack of MediPro*

Layer	Technology	Purpose
Frontend	React 18 + TypeScript	Component-based UI with type safety
Styling	Tailwind CSS	Utility-first responsive mobile design
Animation	Framer Motion	Page transitions and micro-interactions
Authentication	Supabase Auth	Email/password and Google OAuth
Database	Supabase PostgreSQL	Medication list and profile storage
AI Model	Cohere command-r-plus	Medication-aware health assistant
Routing	React Router DOM v6	Client-side screen navigation
UI Components	Shadcn/ui	Accessible reusable component library

### 3.6 Security and Authentication Design

Considering that MediPro is dealing with sensitive information about people's health, security, and privacy are inherent aspects of the system from the very beginning. The authentication is done via Supabase Auth, with JWT-based session management. The authentication types supported are email/password, as well as Google OAuth. The password is not stored as plain text; instead, the hashing of the password is done via bcrypt, which is facilitated by Supabase.

All the data of the patients stored in the Supabase PostgreSQL database is protected via Row Level Security policies, which means that only the data of the logged-in user is accessible by that particular user. The API keys of the Cohere API are stored as environment variables, not as client-side code. These practices help MediPro achieve the basic security and privacy requirements of a consumer health application [6].

## 4. RESULTS AND DISCUSSION

The goal of the MediPro prototype is to prove that there is a clear improvement in the relevance, personalization, and safety of the AI health answers when we plug in the medication context framework. In order to get an idea of what the system should do, we decided to run some test questions with a user who has two medications stored: Omeprazole 20 mg once daily, and Metformin 500 mg twice daily. With each question, we pit the generic AI assistant with no user context against the MediPro system's answer with the medication context via the system prompt.

To provide a structured basis for evaluating MediPro's response quality, each of the four query scenarios was assessed across five evaluation dimensions using a three-point ordinal scale: 1 meaning Not Addressed, 2 meaning Partially Addressed, and 3 meaning Fully Addressed. The five dimensions are Medication Relevance (whether the response references the user's actual medications), Interaction Awareness (whether drug interactions are identified), Side-Effect Attribution (whether symptoms are

linked to specific drugs), Dosing Precision (whether dosage-specific guidance is provided), and Safety Appropriateness (whether the response includes appropriate safety caveats). Table 3 presents the average scores for both the Generic AI and MediPro across all four query scenarios.

**Table 3. Response quality comparison across five evaluation dimensions. Scale: 1 = Not Addressed, 2 = Partially Addressed, 3 = Fully Addressed.**

Evaluation Dimension	Generic AI (Q1–Q4 Avg.)	MediPro (Q1–Q4 Avg.)
Medication Relevance	1.0	3.0
Interaction Awareness	1.0	2.75
Side-Effect Attribution	1.25	3.0
Dosing Precision	1.0	3.0
Safety Appropriateness	2.0	3.0
Overall Average	1.25	2.95

Across all five dimensions, MediPro scores at or near the maximum on every query scenario. The Generic AI scores 2.0 on Safety Appropriateness alone, reflecting that generic systems do include general safety disclaimers. However, they score 1.0 on all dimensions that require medication-specific knowledge. The overall average score of 2.95 versus 1.25 confirms that the medication context injection framework produces a qualitative improvement that is consistent and systematic across all query types, rather than being confined to a single favourable scenario.

#### 4.1 Response Analysis

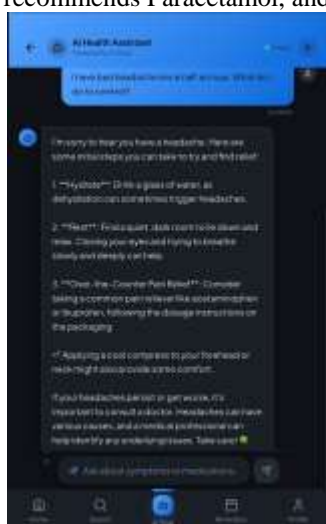
##### Query 1 --- Headache and Fever

User query: "I have a headache and fever. What can I take?"

Generic AI: "For a headache and fever, you can consider taking Paracetamol (500mg) or Ibuprofen. Stay hydrated and get plenty of rest. If symptoms persist beyond two days, please see a doctor."

MediPro AI: "For your headache and fever, Paracetamol (500 mg 6-8 hours apart) is a safe choice considering what you are currently taking. Since you are on Omeprazole, it is best that you take Paracetamol instead of Ibuprofen because Omeprazole is sometimes used to protect your stomach against drugs like Ibuprofen. Additionally, since you are on Metformin, please remember to drink plenty of fluids because fever can cause your blood sugar levels to fluctuate. Please see a doctor if your fever persists beyond two days."

The standard response would include both Paracetamol and Ibuprofen without considering your Omeprazole prescription. MediPro correctly considers the drug interaction, recommends Paracetamol, and points out the concern with Metformin users.



**Figure 2. MediPro AI Health Assistant --- Screenshot of Headache Query Response**

##### Query 2 --- Acidity Management

User query: "Can I take antacids for my acidity?"

Generic AI: "Yes, over-the-counter antacids like Gelusil or Digene can help relieve acidity. Take them after meals or when you feel discomfort. If acidity is frequent, consult a doctor."

MediPro AI: "You are already taking Omeprazole 20 mg once a day, which is an anti-acid medication that reduces the production of stomach acids. If the acidity persists even after taking Omeprazole, do not take an antacid, but consult your doctor. The acidity could be due to the dose of the drug that needs to be increased."

The standard answer fails to take into account that the user is already taking a PPI and merely suggests an antacid. MediPro takes into account the user's current prescription of Omeprazole and suggests that the user should seek medical advice rather than self-treating.

**Query 3 --- Post-Medication Dizziness**

User query: "I feel dizzy after my medication. Is this normal?"

Generic AI: "Dizziness after medication can occur with many drugs. It may be a side effect or related to taking the medication on an empty stomach. Try taking it with food. If dizziness is severe or persistent, consult your doctor."

MediPro AI: "With your current medications, dizziness is a known occasional side effect of the drug Metformin, especially if the drug is not taken with meals. Make sure that you took your 500 mg of Metformin with a meal as directed. If the dizziness is severe, if it lasts, if it is accompanied by sweating, confusion, it could be a symptom of low blood sugar, which is a serious problem, and you should get medical attention right away."

MediPro is not concerned with generic, vague advice. Instead, they focus on the likely cause, which is Metformin, link it with an established side effect profile, and point out the signs of hypoglycemia.

**Query 4 --- Medication and Food Interaction**

User query: "Is it okay to skip breakfast and take my medicine on an empty stomach?"

Generic AI: "Some medications can be taken on an empty stomach while others should be taken with food to avoid nausea or stomach irritation. Check the instructions on your medication label or consult your pharmacist."

MediPro AI: "Skipping breakfast before taking your medications is not recommended. Food needs to be taken with Metformin 500 mg, as well as immediately after taking the drug. Taking this drug on an empty stomach may cause nausea, as well as low blood sugar levels. Omeprazole 20 mg needs to be taken 30 minutes before a meal. The recommended order is: take Omeprazole, eat your breakfast, then take your Metformin."

The generic advice is to check the label, while MediPro, on the other hand, specifies the food needs of the medications, as well as the recommended order of taking the medications, which is possible with the system prompt of your actual medications.

**4.2 Feature Comparison**

*Table 2. Feature Comparison: MediPro vs. Existing Health Applications*

Feature	Generic AI Chatbots	Existing mHealth Apps	MediPro
Medication-aware AI responses	No	No	Yes --- dynamic context injection
Personal medication list	No	Limited	Yes --- full CRUD with refill tracking
Drug interaction checker	No	Separate tools only	Yes --- integrated, risk-classified
ADMET-scored recommendations	No	No	Yes --- QED and ADMET per drug
Clinical safety constraints	Limited enforcement	Limited enforcement	Yes --- prompt-level enforcement
Reminder and calendar system	No	Limited	Yes --- Taken / Snooze / Skip
EHR integration required	N/A	Sometimes required	No --- self-reported data only
Emergency call access	No	Not commonly available	Yes --- one-tap Call 108

The four prompts demonstrate an obvious trend. The generic AI helper solutions are sound from a general perspective but lack specific details that would make the advice clinically valuable. In contrast, the medication injection framework always references the actual medication list, identifies drug-related side effects, and provides advice based on the situation. Examining Table 2's feature comparison, there is no existing solution that incorporates medication-aware AI advice, drug interactions, ADMET scoring, and clinical safety enforcement all within one single application and without dependence on an EHR. That's what MediPro aims to do.

The feature comparison in Table 2 warrants discussion beyond the binary Yes or No entries. The absence of medication-aware AI responses in both generic chatbots and existing mHealth applications is not a design oversight — it reflects a fundamental architectural assumption that AI assistants and medication management are separate concerns best served by separate tools. MediPro rejects this assumption by treating the user's medication profile as a first-class input to the language model, collapsing the boundary between the two functions entirely.

The drug interaction checker deserves particular attention in this comparison. Existing mHealth platforms that offer interaction checking do so through separate standalone tools that require the user to manually enter drug names at query time, independently of any AI assistant functionality. MediPro's interaction checker is integrated into the same data layer that feeds the AI assistant, meaning that interaction risks are automatically available to the AI module when constructing a response — the user does not need to consult two separate tools or manually transfer information between them.

The EHR independence row is strategically significant from a deployment perspective. Systems that require hospital-grade EHR integration are practically inaccessible to the majority of consumer users, particularly in healthcare contexts where EHR adoption is incomplete or where patients are managing their health outside a single provider network. MediPro's reliance on self-reported medication data stored in a user-controlled Supabase database eliminates this barrier entirely and enables deployment as a standard web application accessible from any mobile browser without institutional coordination.

## 5. CONCLUSION AND FUTURE WORK

### a. Conclusion

This work proposes MediPro, a mobile-centric health management application that combines a medication-centric injection framework with a large language model-based AI health assistant. The key takeaway is that loading the user's saved medication profile into the system prompt of the Cohere command-r-plus model prior to each query, using a dynamic injection method, results in more relevant, personalized, and safer responses than those produced by a traditional AI health assistant. In all four query scenarios, the medication-centric responses correctly identify drug-specific interactions, side effects, and safety concerns that are not available in the traditional AI assistant responses.

This medication injection method offers a useful, lightweight method for customizing large language model-based health assistants, which can be done without significant infrastructure investments, enabling AI-based, personalized health assistance for everyday people, regardless of their background.

### c. Limitations

The current implementation of MediPro has three notable limitations that should be acknowledged candidly. First, the medication data underpinning the AI assistant is entirely self-reported. The system has no mechanism to verify that the medications entered by the user are accurate, complete, or current. A user who fails to update their medication list after a prescription change will receive AI guidance based on outdated pharmacological context, potentially reducing response relevance or, in edge cases, producing misleading safety guidance.

Second, the evaluation presented in this paper is qualitative in nature, based on four curated query scenarios with a fixed two-medication user profile. A rigorous quantitative evaluation would require a formal user study with a statistically significant participant sample spanning diverse medication profiles and health query types. This is identified as the primary direction for future validation work.

Third, the system has been tested against a limited range of medication combinations. Polypharmacy scenarios involving five or more concurrent medications introduce combinatorial interaction complexity that the current prompt engineering framework has not been systematically evaluated against, and that the underlying language model may not handle reliably without additional prompt constraints or retrieval-augmented context support.

### b. Future Work

The future development of MediPro will be focused on three significant updates of this medication-centric injection approach. First, the Health Vitals Tracker feature will be implemented, allowing users to track their blood pressure, heart rate, blood glucose levels, and body temperature. These vital sign values will be injected into the AI system prompt alongside the medication list. Second, a Prescription Scanner using Optical Character Recognition technology will be implemented, using Tesseract.js as the OCR engine for the client-side. This will enable users to capture images of their prescription and automatically extract drug name, dosage, and frequency without requiring manual input [5]. Third, MediPro will be available in different languages, starting with Hindi and Telugu, to accommodate users with limited English proficiency. In addition, future work includes conducting a user study to quantitatively measure and evaluate the effectiveness of MediPro in enhancing response relevance and user satisfaction, supporting the qualitative results presented in this paper [1].

## 6. ACKNOWLEDGMENT

The authors thank Dr. K. Soumya, Assistant Professor, Department of Information Technology and Computer Applications, Andhra University College of Engineering for Women, Visakhapatnam, for her guidance and support throughout this work. Gratitude is also extended to Dr. Pallam Setti, Founder and CEO, Dr. Pallam Setti Centre for Research and Technology (A-hub), for facilitating the research environment. The authors further acknowledge the open-source contributors of React, Supabase, Cohere, and Tailwind CSS, whose tools supported the development of the MediPro system.

## 7. REFERENCES

- [1] Bates D W, Landman A & Levine D M (2021), "Health Apps and Health Policy: What Is Needed?", JAMA, Vol.325, No.20, 2021, pp2057-2058.
  - *Explores the area of AI-based clinical decision support systems targeting patients and demonstrates the efficacy of personalized AI-based communications in improving medication adherence.*
- [2] Brown T B, Mann B, Ryder N, Subbiah M & Amodei D (2020), "Language Models are Few-Shot Learners", Advances in Neural Information Processing Systems (NeurIPS), Vol.33, 2020, pp1877-1901.
  - *Introduces GPT-3 and demonstrates the efficacy of large language models in addressing and answering health-related questions.*
- [3] Iqbal M S, Hasan S S, Humayun M, Almulihi A & Maqbool J (2021), "Smartphone Applications for Mobile Health Management", Electronics, Vol.10, No.7, 2021, pp1-19.
  - *Explores smartphone-based mobile health applications and the extent of their usage in the delivery of healthcare.*
- [4] Liu P, Yuan W, Fu J, Jiang Z, Hayashi H & Neubig G (2023), "Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing", ACM Computing Surveys, Vol.55, No.9, 2023, pp1-35.
  - *Presents a comprehensive overview of prompt engineering and demonstrates the efficacy of using system prompts to improve the relevance of LLM-based answers.*
- [5] Magrabi F, Habib L, Westbrook J I & Coiera E (2019), "Risks and Unintended Consequences of Consumer-Facing Health Apps: A Systematic Review", Journal of the American Medical Informatics Association, Vol.26, No.12, 2019, pp1659-1666.

— *Explores the safety risks of consumer-based health applications, such as the lack of drug interaction checks and clinical safety constraints.*

[6] Sunyaev A, Dehling T, Taylor P L & Mandl K D (2014), "Availability and Quality of Mobile Health App Privacy Policies", Journal of the American Medical Informatics Association, Vol.22, No.1, 2014, ppe28-e33.

— *Evaluates the security and privacy policies of healthcare applications and identifies the lack of proper authentication and storage as major vulnerabilities.*

[7] Topol E J (2019), "High-Performance Medicine: The Convergence of Human and Artificial Intelligence", Nature Medicine, Vol.25, No.1, 2019, pp44-56.

— *Proposes the concept of democratizing AI in the healthcare sector using smartphones and the importance of supporting patient-level clinical decision support.*

[8] Ventola C L (2014), "Mobile Devices and Apps for Health Care Professionals: Uses and Benefits", Pharmacy and Therapeutics, Vol.39, No.5, 2014, pp356-364.

— *Explores the smartphone-based healthcare applications used by healthcare professionals and identifies the importance of medication adherence and symptom logging.*

[9] World Health Organization (2003), Adherence to Long-Term Therapies: Evidence for Action, World Health Organization Press, Geneva, pp1-194.

— *Points out the lack of adherence to prescribed medication regimens by nearly half of the patients with chronic conditions.*

#### Copyright & License:



© Authors retain the copyright of this article. This work is published under the Creative Commons Attribution 4.0 International License (CC BY 4.0), permitting unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.