Healix – Intelligent Care for A Healthier You

Yashaswi C Shetty[1], Rithika Sharma[2], Threesha[3], Yashaswi K B[4]
[1,2,3,4]*Member, Srinivas Institute of Technology*
doi.org/10.64643/IJIRTV12I12-206767-459

***Abstract*—Healthcare is going digital, and people expect more than just isolated features—they want actual support, right when they need it. That's where Healix comes in. We built an intelligent, web-based healthcare platform that rolls together an AI-powered chatbot, emergency communication, and personal health tools in one place. Healix uses Retrieval-Augmented Generation (RAG) to pull the right medical info before answering you, making the responses clear and trustworthy. The frontend runs on Next.js, so everything feels smooth on any device. On the backend, Supabase keeps user data secure and handles accounts. When emergencies strike, Twilio sends out alerts and shares your location instantly. We put a lot of work into making Healix fast, private, and easy to use across different devices. Our tests show Healix is stable, answers questions well, and handles emergencies without missing a beat. If you want accessible healthcare—especially somewhere with few resources—Healix offers a solid, user-friendly solution.**

***Index Terms*—Artificial Intelligence, Healthcare Systems, Retrieval-Augmented Generation (RAG), Chatbot, Emergency Communication, Web Application, Health Monitoring, Next.js, Supabase, Twilio, Digital Healthcare, Intelligent Systems.**

I. INTRODUCTION

Digital tech has changed healthcare. Now people can get advice remotely, ask questions any time, and even keep an eye on their health through their phones. But for all the progress, getting help quickly, especially in emergencies, is still a headache. Most apps out there only do one thing—book appointments, check symptoms, or maybe just track steps. That scattershot approach wastes time and means people don't get what they need when things go wrong.

Another issue: most chatbots don't use trustworthy medical sources. They give answers but sometimes miss the mark, and they rarely include emergency features. That means when something serious happens, users get left on their own. To fix this, we built Healix—an all-in-one system with a smarter medical chatbot, built-in emergency communication, and basic wellness tools. With Retrieval-Augmented Generation, our chatbot leans on relevant, up-to-date medical info for each answer, which helps users feel confident and protected.

II. RELATED WORK

Healthcare tech has come a long way—from simple tracking apps to much smarter AI-powered systems that can grab information and chat in real time. Lewis et al. introduced a RAG system that teams up information retrieval with AI-generated answers, which made responses a lot stronger. Karpukhin and others figured out faster ways to pull up the right details using dense passage retrieval. Shuster's work added real-world grounding to chatbot conversations—a big step forward for reliability.

On the infrastructure side, Wang and his team designed cloud-based systems to make sure patient data stays secure and can scale up as needed. Singh's team built a real-time emergency platform that uses both mobile and the cloud—so medical alerts and a user's location get sent out fast during a crisis.

There are plenty of chatbots that give medical tips and offer monitoring. Research shows these chatbots can really help, but only when they're pulling from accurate info. Most solutions work alone—either chatting, or tracking, or handling emergencies, but not all three. Healix ties all these pieces together, making it stand out among today's healthcare tools.

III. SYSTEM DESIGN

A. Architecture Overview

Healix is built with four layers that keep the whole system flexible and strong: the Frontend Interface Layer (what people see and use), the Backend Services Layer (user accounts and data storage, secured and scalable), the AI Processing Layer (the RAG-based chatbot engine for smart answers), and the

Communication Layer (real-time alerts and messages when emergencies hit). Data moves smoothly using fast, asynchronous APIs, so users never feel lag.

B. RAG Chatbot Engine

Here's how a user's question gets handled:

1) First, Healix cleans up and turns your question into a vector—basically, finding its meaning in numbers.
2) Then, it hunts through a vector database of medical facts for the most relevant details.
3) Finally, the language model takes your question and the context found, and writes a grounded, accurate answer.

C. Health and Emergency Module

When users want health checks or emergency help, the system flips on dedicated tools. Wellness updates show up in a clear dashboard—basic, but useful. For emergencies, users hit a button and their alert goes out instantly with their location to chosen contacts, all through Twilio. User privacy comes first: only the needed info is sent out, nothing more.

IV. METHODOLOGY

The way we built Healix kept everything structured. We pinned down what features we had to have—like the intelligent chatbot, fast emergency messages, and health tracking—while making sure the system would always run quick, scale up, and keep data private.

To prep the chatbot, we collected quality medical info, split it up into pieces, turned those into embeddings for fast reference, and matched each user question with the right info using semantic similarity. Every module links in securely with RESTful APIs, so everything talks together seamlessly. We put serious effort into optimizing performance and handling errors, so users don't get stuck if something goes wrong.

V. IMPLEMENTATION

A. Chatbot Algorithm

At its core, the chatbot works like this:

```
FUNCTION GenerateResponse(query):
embedding = CreateEmbedding(query)
documents = RetrieveRelevantDocs(embedding)
context = BuildContext(documents)
response = LanguageModel(query, context)
RETURN response
```

B. Emergency Alert Algorithm

And the emergency system runs something like this:

```
FUNCTION SendEmergencyAlert(user):
location = GetCurrentLocation ()
message = "Emergency needs help and location
FOR each contact IN EmergencyContacts:
SendSMS(contact, message)
RETURN "Alert Sent"
```

C. Operating Environment

Healix is built with all the usual web stuff—HTML, CSS, and JavaScript. Next.js powers the user experience, making navigation quick and smooth. Supabase handles authentication and stores data, syncing things in real time. The chatbot uses powerful AI to combine language understanding with retrieval. Emergency SMS and alerts come through Twilio. Healix works on all major browsers, doesn't need much power (basically any recent laptop or tablet will do fine), and is ready for upgrades like more advanced analytics or personal medical recommendations. User data is encrypted and login is locked down to keep info safe.

VI. TESTING AND RESULTS

A. White Box Testing

Tests checked each part step by step:

- Embeddings generate correctly.
- Document search works.
- Context builds as planned.
- Answers match expectations.
- API calls complete with no errors.
- Emergency logic fires instantly.
- Error handling catches problems before users notice.

B. Black Box Testing

We also tested the platform like real users would:

- The chatbot answers questions accurately.
- If you send something unrelated, it responds gracefully.
- Emergency alerts work and send the right info.
- Your location is transmitted as expected.
- Even on a slow connection, the system stays responsive.
- If a backend service drops, it recovers without crashing.

- Login and sessions work for different users.
- Several people can use it at once with no slowdowns.
- The interface displays well on any device.
- It handles all kinds of user input smoothly.

C. Performance Metrics

Chatbot answers pop up within 2–3 seconds—a near real-time feel. Emergency alerts get out fast, too, depending on network speed. Even when lots of people use it, performance holds steady. The system stays lean, only using the resources it needs. Takeaways: context is always there in replies, users trust the outputs, and emergency features protect users when it counts most.

VII. CONCLUSION

Healix shows what an integrated digital health platform should be. With AI-powered chat, built-in emergency support, and core wellness tools all living together on the web, Healix raises the bar. Using Retrieval-Augmented Generation means answers are not just quick—they're accurate and rooted in proper context, so users can really rely on them. Emergency support and basic health tracking fill in gaps many other solutions miss.

Tests proved Healix is fast, stable, and keeps user data private. In the end, it's a practical, scalable platform that improves access to healthcare—especially where resources are tight.

REFERENCES

[1] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, pp. 9459–9474, 2020.

[2] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W. Yih, "Dense Passage Retrieval for Open-Domain Question Answering," in Proc. Conf. Empirical Methods Natural Language Process. (EMNLP), pp. 6769–6781, 2020.

[3] K. Shuster, S. Roller, E. Dinan, Y.-L. Boureau, and J. Weston, "Retrieval Augmentation Reduces Hallucination in Dialogue Systems," in Findings Assoc. Comput. Linguistics (ACL), pp. 3784–3803, 2021.

[4] S. H. Wang, P. Wang, and J. Chen, "Secure Cloud-Based Architecture for Healthcare Information Systems," IEEE Access, vol. 8, pp. 140582–140593, 2020.

[5] A. Singh, R. Gupta, and M. Sharma, "Real-Time Emergency Communication Systems Using Mobile and Cloud Integration for Healthcare Applications," Int. J. Med. Informatics, vol. 150, pp. 104–112, 2021.

[6] A. Vaswani et al., "Attention Is All You Need," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 30, pp. 5998–6008, 2017.

[7] J. Devlin et al., "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding," in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol. (NAACL-HLT), pp. 4171–4186, 2019.

[8] T. Brown et al., "Language Models Are Few-Shot Learners," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 33, pp. 1877–1901, 2020.

[9] D. Jurafsky and J. H. Martin, Speech and Language Processing, 3rd ed. Pearson, 2023.

[10] E. H. Shortliffe and J. J. Cimino, Biomedical Informatics: Computer Applications in Health Care and Biomedicine, 4th ed. New York, NY, USA: Springer, 2014.

[11] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. Dudley, "Deep Learning for Healthcare: Review, Opportunities and Challenges," Brief. Bioinform., vol. 19, no. 6, pp. 1236–1246, 2018.

[12] A. Esteva et al., "A Guide to Deep Learning in Healthcare," Nat. Med., vol. 25, pp. 24–29, 2019.

[13] M. Chen, Y. Hao, K. Hwang, L. Wang, and L. Wang, "Disease Prediction by Machine Learning over Big Data from Healthcare Communities," IEEE Access, vol. 5, pp. 8869–8879, 2017.

[14] S. Reddy, M. Fox, and M. Purohit, "Artificial Intelligence-Enabled Healthcare Delivery," J. Roy. Soc. Med., vol. 112, no. 1, pp. 22–28, 2019.

[15] G. Aceto, V. Persico, and A. Pescapé, "The Role of Information and Communication Technologies in Healthcare: A Review," IEEE Access, vol. 6, pp. 129–154, 2018.

[16] J. Zhang, Y. Chen, and J. Liu, "Mobile Health Applications for Emergency Response: A Survey," IEEE Commun. Surveys Tuts., vol. 22, no. 4, pp. 2766–2785, 2020.
[17] L. Chen, H. Chen, and D. Yang, "Smart Healthcare System Design Using IoT and Cloud Computing," Future Gener. Comput. Syst., vol. 86, pp. 580–589, 2018.
[18] S. Latif, R. Rana, S. Younis, J. Qadir, and J. Epps, "Transfer Learning for Improving Speech Emotion Classification Accuracy," IEEE Access, vol. 6, pp. 5168–5179, 2018.
[19] N. Mehta and A. Pandit, "Concurrence of Big Data Analytics and Healthcare: A Systematic Review," Int. J. Med. Informatics, vol. 114, pp. 57–65, 2018.
[20] K. Denecke and S. Gabarron, "Artificial Intelligence for Healthcare Chatbots: Opportunities and Challenges," J. Healthcare Informatics Res., vol. 5, pp. 183–203, 2021.
[21] E. B. Liddy, "Natural Language Processing," in Encyclopedia of Library and Information Science, 2nd ed. New York, NY, USA: Marcel Dekker, 2001.
[22] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou, "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 35, pp. 24824–24837, 2022.
[23] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. Cambridge, MA, USA: MIT Press, 2016.
[24] E. J. Topol, Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again. New York, NY, USA: Basic Books, 2019.
[25] S. K. Singh and P. Gupta, "Cloud-Based Smart Healthcare Monitoring System Using IoT," Int. J. Adv. Comput. Sci. Appl., vol. 10, no. 5, pp. 1–7, 2019.