

Author Response to Reviewer Comments

We sincerely thank the reviewer for their valuable and constructive feedback. We have revised the manuscript accordingly and provide detailed responses to each comment below:

1. The abstract could be more concise, mention key contributions and outcomes.

Response:

Thank you for this suggestion. We have revised the abstract to be more concise while explicitly highlighting the key contributions and outcomes of our work. Specifically, we now emphasize:

- The development of a hierarchical query system for indigenous medicine knowledge,
 - The integration of hybrid SQL/NoSQL data structures,
 - A client-server model with load-balanced deployment, and
 - Our alignment with Sustainable Development Goal 3.
- This revised abstract succinctly communicates the novelty and societal impact of HKIMed.
-

2. Explicitly state how HKIMed differs from existing systems. Highlight unique features like hierarchical queries or hybrid SQL/NoSQL databases.

Response:

We appreciate the reviewer's request for clearer differentiation. A new section has been added to explicitly compare HKIMed with existing indigenous knowledge systems. HKIMed is distinct in that it:

- Supports **hierarchical querying**, allowing users to traverse multi-level knowledge structures (e.g., disease → symptom → treatment → herbal composition),
 - Implements a **hybrid SQL/NoSQL architecture** to balance structured medical metadata with unstructured traditional narratives, and
 - Utilizes a **cluster-based load-balanced server model** for improved accessibility and scalability.
-

3. Clarify the hybrid SQL/NoSQL implementation. How are relational and non-relational data integrated?

Response:

We have clarified the hybrid architecture in the implementation section. Relational data (e.g., disease taxonomy, medicinal properties) is stored in a PostgreSQL database, while non-relational data (e.g., ethnographic records, user annotations, community inputs) is managed using MongoDB.

Integration is achieved through a unified API layer and an indexing strategy that maps SQL primary keys to NoSQL document references, enabling seamless cross-database querying and rendering within the MVC framework.

4. Specify the load-balancing algorithm and justify its choice.

Response:

We have updated the system architecture section to specify that HKIMed uses the **Least Connections** load-balancing algorithm, implemented via **NGINX with HAProxy**. This choice was made because HKIMed supports a highly variable user base with differing session lengths, and the Least Connections algorithm is better suited than Round Robin for handling uneven traffic distribution, ensuring lower response times and improved user experience in real-time queries.

5. Compare HKIMed's performance with existing systems to highlight improvements in query speed or usability.

Response:

We have added a comparative evaluation subsection that benchmarks HKIMed against two existing platforms.

6. Include more recent studies (2021–2024) on indigenous knowledge systems or load-balancing techniques.

Response:

Thank you for pointing this out. We have significantly updated our literature review with recent studies from 2021–2024.

We hope these revisions meet the expectations of the reviewer and improve the clarity, technical strength, and relevance of the manuscript. We welcome any further suggestions for enhancement.

Sincerely,
Purab Das