Brainwave: A Multidisciplinary Journal, vol. 7, no. 2, pp. 1416-1437, June 2026
ISSN(online): 2582-659X

A design for an enhanced data storage in PPH 4.0 involving a time-based graph database

by Sweety Chatterjee[1*], Mst Hanifah Khatun[1], Ayantika Saha[1], Moupriya Sarkar[1]

[1]Dept. of Information Technology, Techno International New Town, Kolkata-700156, India
**Corresponding Author. Email: sweety.chatterjee.it.2022@tint.edu.in*

Received on: April 20, 2026 | Accepted on: June 19, 2026 | Published on: June 30, 2026

Abstract

The design of Privacy-Preserving Health 4.0 (PPH 4.0) inherently addresses the challenges associated with efficient big data storage and management. To improve data management capabilities in the Healthcare 4.0/5.0 scenario, we aim to present a novel architecture and deployment strategy involving a time-based graph database, integrating the strengths and functionalities of both the time-series database model and the graph database model. In our presented approach, InfluxDB©, ®, representing a time-series database model, and Neo4j©, ®, demonstrating a graph database model, were conjointly utilized to establish a hybrid and effective data management framework in PPH 4.0. With reference to big data scenario, the proposed time-based graph database exhibits optimized approach for relationship-intensive queries, achieving superior performance as compared to traditional databases. Additionally, the proposed time-based graph database in PPH 4.0 offers enhanced flexibility via a dynamic schema that is capable to adapt the evolving data models, high-performance ingestion, and querying with time-stamped data, and efficient storage. Furthermore, the built-in temporal analysis functionalities may enable comprehensive time-based insights, and thereby may facilitate enhanced analytics, and quick data-driven decision-making process. Experimental results from the successful deployment of our presented model demonstrate its potential, and effectiveness for real-world applications in eHealth scenarios.

Keywords: *eHealth; PPH 4.0 (Privacy Preserving Health 4.0); time-series database; graph database; time-based graph database.*

1. Introduction

The major developments in healthcare services and eHealth are primarily due to the integration of several advanced technologies that enhance service accessibility, efficiency, and quality. At present, eHealth facilitates different digital services, e.g., remote monitoring, virtual consultations, and electronic health record management, especially for patients residing in remote areas. The primary goal in eHealth is to enhance the healthcare outcomes, empower both patients and healthcare professionals, and extend services beyond traditional clinical settings at a reduced cost. Thus, eHealth has been a focus among researchers for possible cost effective, secured and sustainable improvements [1-3]. On the other hand, the shift toward eHealth 5.0 is closely linked with the broader industrial transformation influenced by Industry 5.0. While Industry 4.0 introduced automation and data exchange through technologies like Artificial Intelligence (AI), Internet of Things (IoT), Cloud

Computing, and robotics, Industry 5.0 highlights a more human-centered approach. Thus, it transforms the existing healthcare system from hospital-centric models to personalized, intelligent systems that further enhance patient engagement, operational sustainability, and responsiveness to evolving challenges [1-4]. A typical diagram representing the broad relationship between Healthcare 4.0 and Healthcare 5.0 is presented in the following Figure 1.

From Figure 1, it might be observed that, among the core enablers of eHealth, cloud computing plays a vital role by offering scalable and cost-efficient data storage and data processing. It facilitates on-demand access to medical records and optimizes IT resources. However, its deployment in eHealth also introduces possible critical challenges such as a lack of data privacy, possible mismatch to different regulatory compliances, and the security of sensitive medical information. To support enhanced data security and privacy, among several others, we particularly find an interesting design in [2]. In [2], we observed the design of a Privacy-Preserving Health 4.0 (PPH 4.0) that supported cost-effectiveness towards enhanced data privacy and security by involving an integration of Machine Learning (ML) and cellular automata (CAs)-based modelling. In the PPH 4.0 framework, the ML algorithm offered effective data-cleaning, and CAs offered inherent parallelism and a very large-scale integration facility at a low cost with respect to its physical implementation and low power consumption. The PPH 4.0 framework contained five major modules: (i) ML-based approaches for data cleaning; (ii) ML-based approaches for data dimension reduction; (iii) CAs-based MapReduce framework to effectively process high-volume user data, (iv) CAs-based ABE to effectively provide enhanced data security, and (v) CAs-based lightweight authentication towards authenticated access of stored secure data. Further in [3], signal encryption at MapReduce with ECAs generated a pseudo-random noise signal, and collaborative attribute-based access with ECAs was discussed for enhanced data security and privacy in PPH 4.0. Unfortunately, in [2, 3], no focus on secured data storage in a big data scenario was observed; big data is characterized by five attributes— volume, velocity, variety, veracity, and value. Volume measures data size; velocity captures the speed of data generation; variety encompasses data diversity; veracity assesses quality and reliability; and value reflects the meaningful insights derived from the data [4].

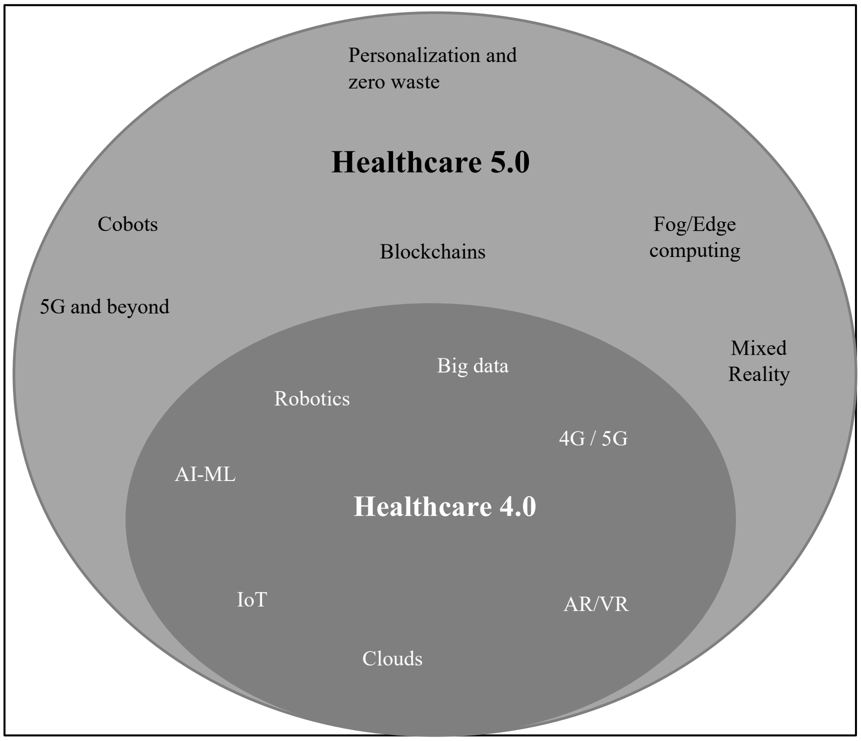


Figure 1 A diagram to represent the broad relationship between Healthcare 4.0 and Healthcare 5.0 (partially inspired by [1, 3])

In another attempt to ensure enhanced data security and privacy in such scenarios, we observed that several researchers have suggested that the successful implementation of enhanced data security and privacy requires robust encryption, secure access control mechanisms, and dependable service-level agreements (SLAs) [5]. To deal with the high volume of time-sensitive data, Time series databases (TSDBs) [6] were designed to store, manage, and analyze such time-stamped data efficiently. We learned that TSDBs may allow to resolve fast queries and aggregation of continuous data streams [6], and further may support high ingestion rates and execution of complex queries with low latency [7]. As our presented research is greatly inspired by the uses of TSDBs, a brief description of TSDBs is presented next.

A time series database typically facilitates optimal storage and management for a sequence of data points, which typically records data at discrete time intervals in chronological order. Each data point is assigned with a timestamp value, allowing for an information retrieval and real-time data analysis. However, working with such data is often very crucial, as those data are often highly correlated, incomplete, and often collected from multiple sources [7-12]. A typical diagram representing a time series database is presented in the following Figure 2.

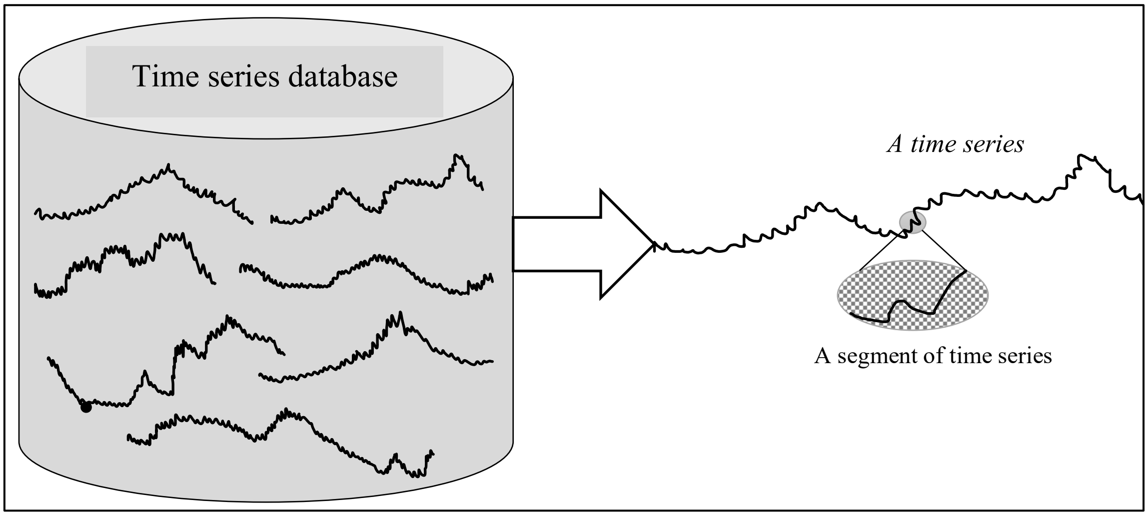


Figure 2 Time series database (partially inspired by [11])

On the other hand, to handle the complex and interconnected healthcare data, graph databases (GDBs) (refer to Figure 2) offer an alternative approach. GDB represents the data as collection of nodes and edges, which directly model all relationships between all entities. Although, performance benchmarks show that graph databases like Neo4j©, ® may significantly outperform traditional systems in pattern-matching and traversal tasks [12-14]. However, graph databases also have limitations: e.g., less efficiency for aggregation or sorting tasks, requirement for specialized query languages, and need for a high storage volume due to metadata overhead [14]. A typical diagram representing a graph database is presented in the following Figure 3.

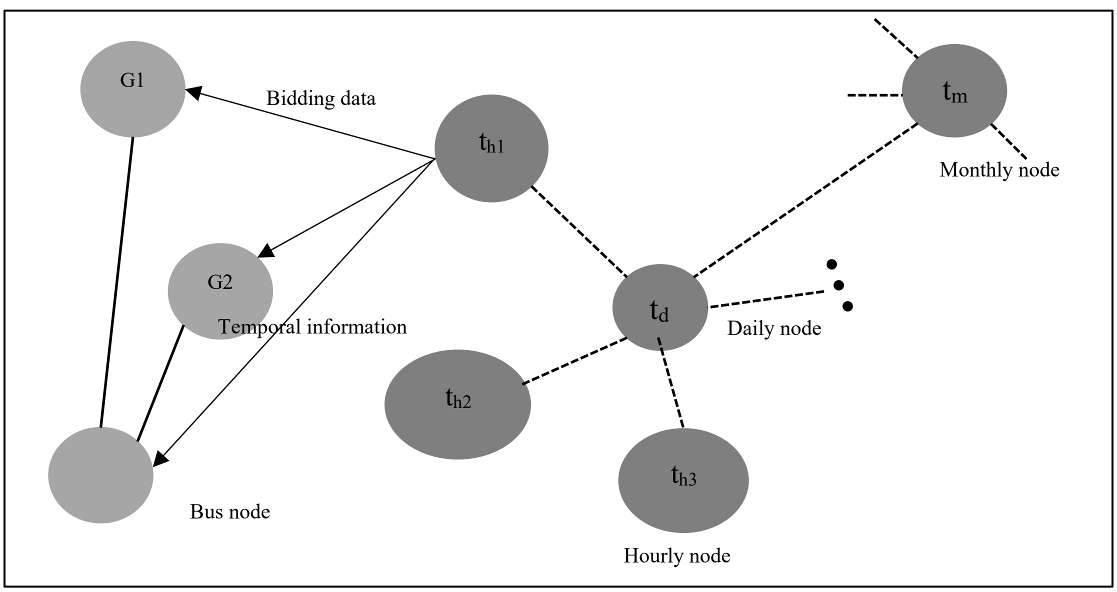


Figure 3 Time-based graph database (partially inspired by [15])

Background and Motivation

It might be noted that we are deeply influenced by the PPH 4.0 design of [2, 3]. From our studies of [2, 3], we learned that in modern healthcare, real-time action is critical. Time-sensitive data like oxygen levels or heart rate patterns must be processed instantly to support timely intervention. At the same time, health data is deeply relational: patients interact with multiple doctors, medications, and conditions in complex ways. Although we observed several novel data security and privacy approaches in Healthcare 4.0/ Healthcare 5.0 were presented in [2, 3]. Unfortunately, the issue of designing an efficient database in such a scenario was unanswered. We observed that traditional databases in such scenarios may struggle to handle both dimensions, time and relationships, simultaneously. As digital healthcare evolves, especially with the rise of eHealth 5.0, neither time-series are databases or graph databases alone sufficient to meet their full demands. We believe that a hybrid approach involving both time series and graph databases may offer a more robust solution by combining temporal tracking with relational mapping. We further believe that such a hybrid model will support real-time, context-aware decision-making by maintaining both time-based data streams and the complex connections between healthcare entities. For this reason, we find there exists a scope for further research that might enhance the design of PPH 4.0 by providing an effective and secure design of database/data storage in Healthcare scenarios.

The major contributions of the presented research paper are as follows.

a) It presents a design and implementation of a secure data storage in PPH 4.0.
b) It presents an advancement in eHealth in view of a secure data storage design involving the hybrid characteristics of both time-based database (i.e., InfluxDB©, ®) and graph database (i.e., Neo4j©, ®).
c) It presents an effective Cloud integration in PPH 4.0.

The rest of the manuscript's architecture is as follows. Related works are presented in Section 2; the proposed design is presented in Section 3; experimental results and analysis are presented in Section 4; and conclusion, limitation, and future research direction are presented in Section 5.

2. Related works

The rise of digital technologies in the healthcare sector has led to significant research into enhancing patient care, data security that ensures effective healthcare services are delivered efficiently. At present, eHealth provides several opportunities, e.g., personalized medicines, advanced diagnostics, and patient-centric care, by making the use of cutting-edge technologies like artificial intelligence, blockchain, big data analytics, and robotics. Healthcare 5.0 is typically built on several technological advancements of the previous stages (i.e., Healthcare 4.0, Healthcare 3.0, Healthcare 2.0, and Healthcare 1.0). The evolution of modern healthcare began with traditional personal care, and direct interaction with physicians in Healthcare 1.0, followed by Healthcare 2.0's digital data management using Electronic Health Records (EHRs) and public health promotion and preventive care in Healthcare 3.0. To enhance diagnosis, monitoring, and treatment, Healthcare 4.0 introduced various technologies, e.g., AI (Artificial Intelligence), IoTs (Internet of Things), and big data analysis. Healthcare 5.0 is built upon the changes caused by emphasizing patient-centric care, and individualized care that typically depends on social, environmental, and behavioural aspects. It facilitates human-centric principles of Industry 5.0, which emphasize empathy, sustainability, and human-machine collaboration to improve healthcare outcomes and delivery [3, 4].

On the other hand, big data refers to an extremely high volume of data that typically may contain complex data that cannot be managed easily by using traditional data processing tools. By nature, data might be structured, semi-structured, unstructured, or mixed. Big data is defined by five V's: volume, velocity, variety, veracity, and value [16, 17].

With the integration of healthcare with Cloud computing, it became easier to store and manage such big data easily, overcoming several issues, e.g., synchronization problems, security concerns, and storage limitations. The Cloud-based healthcare storage enhances accessibility, cost-effectiveness, reliability, scalability, and improves the quality of medical services. However, ensuring data availability and security is still challenging [18]. To effectively handle big data, Graft, is often used, which transforms big data into a graph structure to mine co-occurring patterns efficiently. Unlike traditional methods, Graft uses symbolic representation and Piecewise Aggregate Approximation (PAA) to reduce data dimensionality while preserving all the temporal dependencies [19, 20].

In a distinct effort, a ttraditional time series mining was applied to medical domains, including electroencephalography (EEG) for event identification [21]. In [21], the time series databases were transformed into graph structures where series became nodes with similarity-based edges. In distinctive other efforts, researchers tried to enhance the services of eHealth. Among several others, in [22-26], inclusion of several technological advancements in eHealth were suggested that might led to a long-lasting positive impact in eHealth design, along with its comprehensive evaluation. Further, in [27-29], we observed how the growing emphasis on intelligent, connected, and sustainable healthcare has paved the way for Healthcare 5.0, a paradigm that integrates advanced technologies to create smarter, more patient-centric designs for healthcare systems. In healthcare, IoT devices such as wearable sensors and monitoring systems generate vast real-time data streams, necessitating scalable, cloud-based infrastructure for efficient storage and analysis [30-32]. In [33], a novel architecture was presented that reduced network congestion and latency, supporting timely diagnosis, responsive systems, and customized healthcare delivery [34].

In recent days, we observed several studies [31-35] that explores advanced data representation and analysis techniques across databases, radar systems, and healthcare. Bohlen et al. present temporal coalescing methods to efficiently manage time-based database records. Chen et al. introduce a graph-based radar target detection framework using Spatial-Temporal Graph Fourier Transform for improved accuracy. Elkoumikhi et al. review Personal Health Knowledge Graphs for personalized healthcare and decision support. Tan et al. propose FKG-MM, a multimodal fuzzy knowledge graph that integrates diverse healthcare data and handles uncertainty for enhanced clinical reasoning.

In different other researches, uses of traditional time series mining methodology were suggested to analyse big data at real time [39-41] to achieve real time analysis which was not available in graph databases of [19, 20].

From our studies, we observed that, graph database and time-series database both have important characteristics to handle and store big data, but we found that none of them are complete and efficient enough to handle the real-time big data analysis and easy modelling of relationships between entities, simultaneously (refer to the brief description of Graph database and time series-based database of Section 1). Despite several individual advantages for Graph database and time series-based database, the integration of time series analysis and graph-based modelling in eHealth 5.0 remains underexplored that might facilitate both easy modelling of relationships between entities, and real-time data analysis in view of big data scenario, simultaneously. For this reason, we believe that a hybrid approach involving such an integration of both time series analysis and graph-based representations might be more suitable for designing an effective data storage for healthcare applications. For this reason, we find a scope for our research.

3. Proposed design

As already mentioned in the previous section, we plan to incorporate both the benefits of a time sample-based database and a graph database. For said reasons, for a time sampled-based database we used the InfluxDB©, ® (available at https://www.influxdata.com/, accessed on October 30, 2025), and for a graph database we used the Neo4j©, ® (available at https://neo4j.com/, accessed on October 30, 2025). From studies, we explored that InfluxDB©, ® is optimized for storing and querying high-volume metrics, ideal for IoT and monitoring, while Neo4j is built for managing and querying highly connected data, such as social networks or knowledge graphs.

InfluxDB©, ®excels at fast ingest and analysis of timestamped data using an SQL (structured query language)-like query language, whereas Neo4j©, ®'s strength is in efficiently traversing relationships using its cypher query language [41].

A typical flowchart for our proposed approach is presented in the following Figure 4.

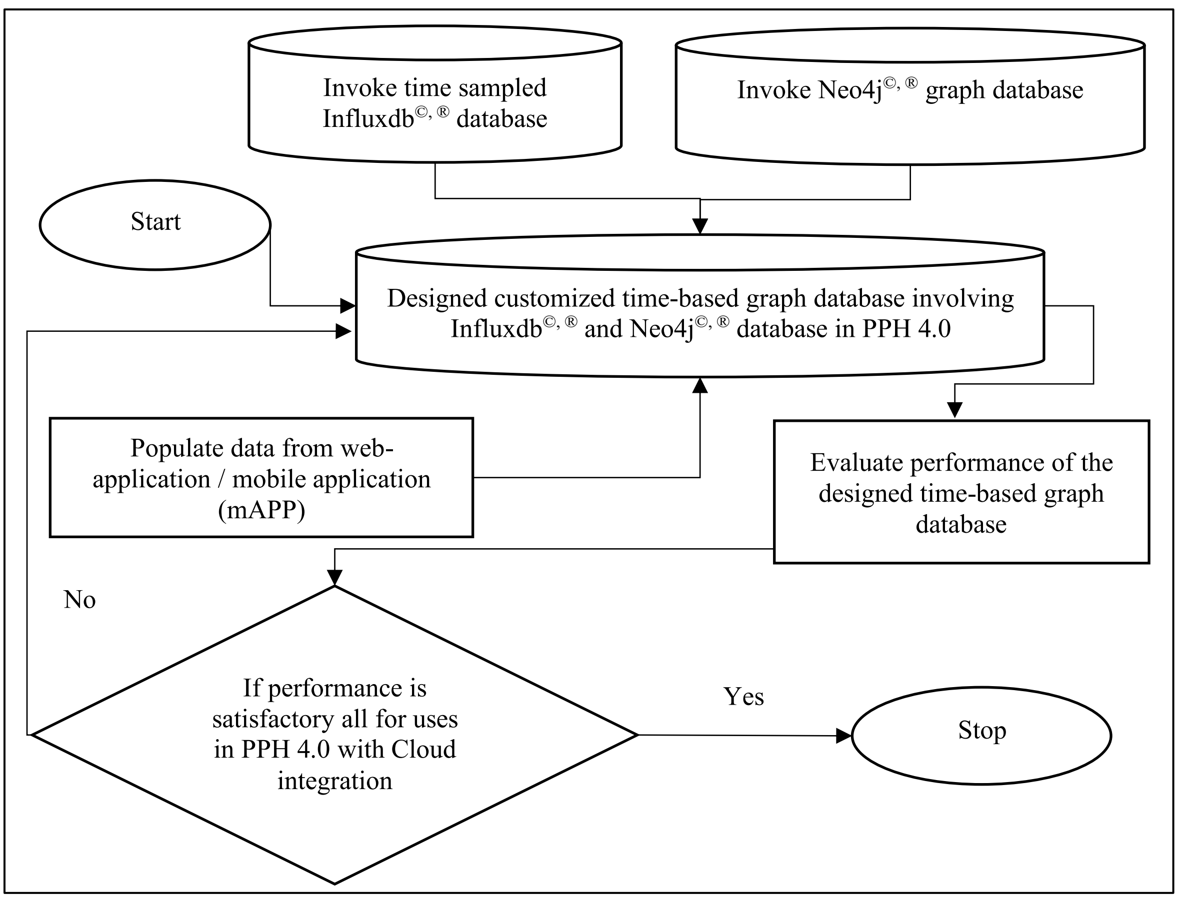


Figure 4 A typical flowchart for our proposed approach

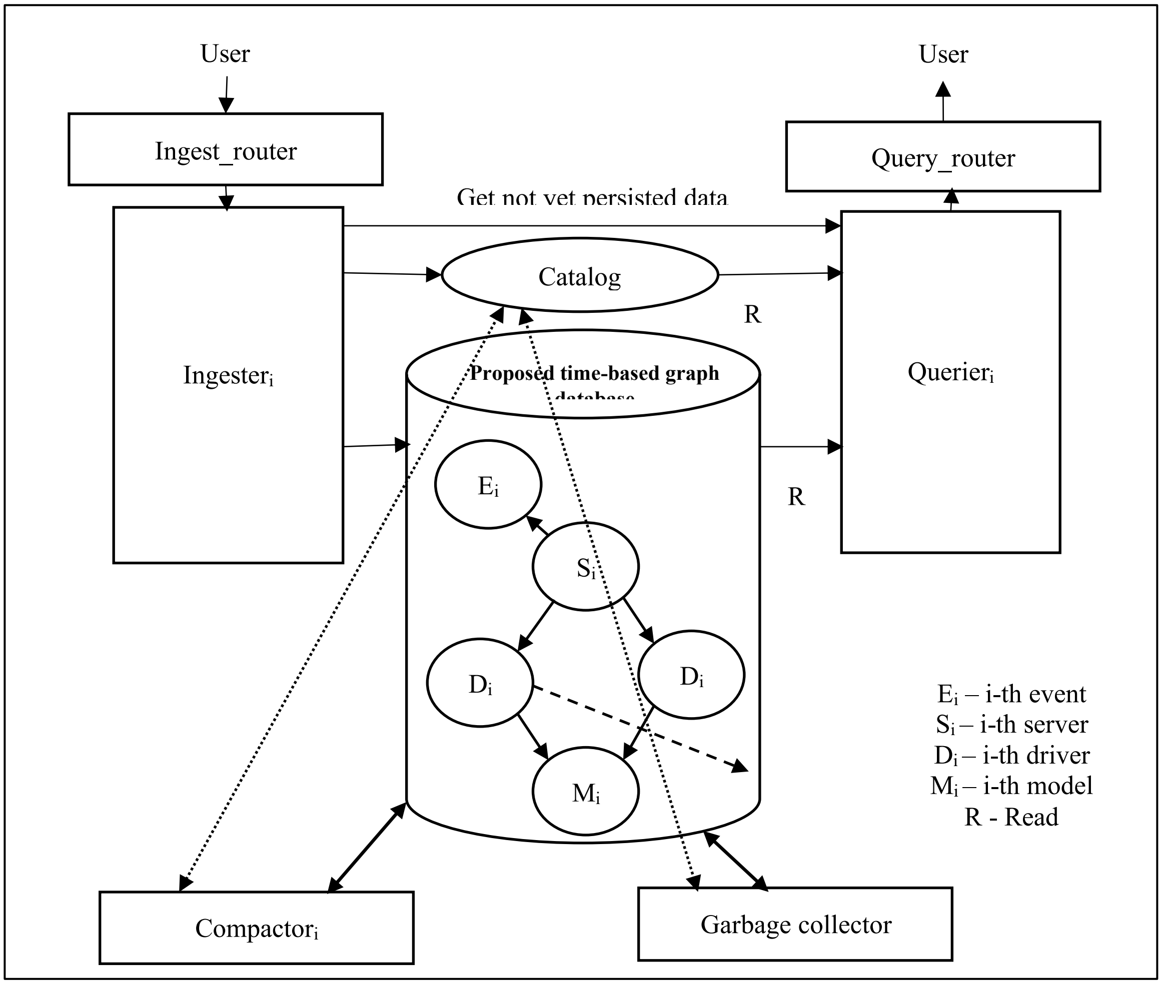


Figure 5 Proposed architecture for a time-based graph database in PPH 4.0 (partially inspired by combining [42,43]

Figure 4 presents the flowchart of our proposed approach. The process begins with invoking both the InfluxDB©, ® time-sampled database and the Neo4j©, ® graph database. These are combined to design a customized time-based graph database under the PPH 4.0 framework. Data from web or mobile applications (mAPP) is then populated into this system. After that, the performance of the designed database is evaluated. If the performance meets the required level for all use cases in PPH 4.0, which could be integration, the process stops; otherwise, the design is revised until satisfactory results are obtained.

Figure 5 represents the proposed architecture of the time-based graph database in PPH 4.0. It includes several components for data ingestion, query handling, and storage. The *Ingest_router* and *Query_router* manage data input and user queries, while the Ingester module stores data in the proposed database that connects different entities such as events (E_i), servers (S_i), drivers (D_i), and models (M_i). The Catalog component handles metadata and temporary data. The Compactor and Garbage Collector ensure the database remains efficient by compressing data and removing outdated entities. This design supports time-based data management and improves overall system performance.

4. Experimental results and analysis

To design an enhanced and secure time-based graph database schema in PPH 4.0, we presented a detailed graph database schema in 3NF (Third Normal Form), which is based on the eHealth ER (Entity Relationship) diagram (refer to Figure A1 of Annex) involving a "database forward engineering" (i.e., the process of converting an ER model to a relational schema) [44]. It might be noted that, unlike the well-defined principles of designing a RDBMS (Relational Database Management Systems), the design rationale for a graph database is often found to be less structured and ad hoc [44]. Converting a relational model into a graph model is conceptually similar to transforming it into an ER model. The presence of keys in a 1NF (First NF) relational model naturally corresponds to the existence of nodes in a graph database. Relationships in the graph model can be viewed as counterparts to those in the relational model, while node properties are determined according to 3NF requirements. The essence of 3NF can be summarized as: "every non-key attribute must depend on the key, the whole key, and nothing but the key" [45]. That is "each non-key attribute describes the key (i.e., the node in the graph) exclusively and completely, establishing a unique connection to that node" [44]. Mathematical representation for the same is as follows.

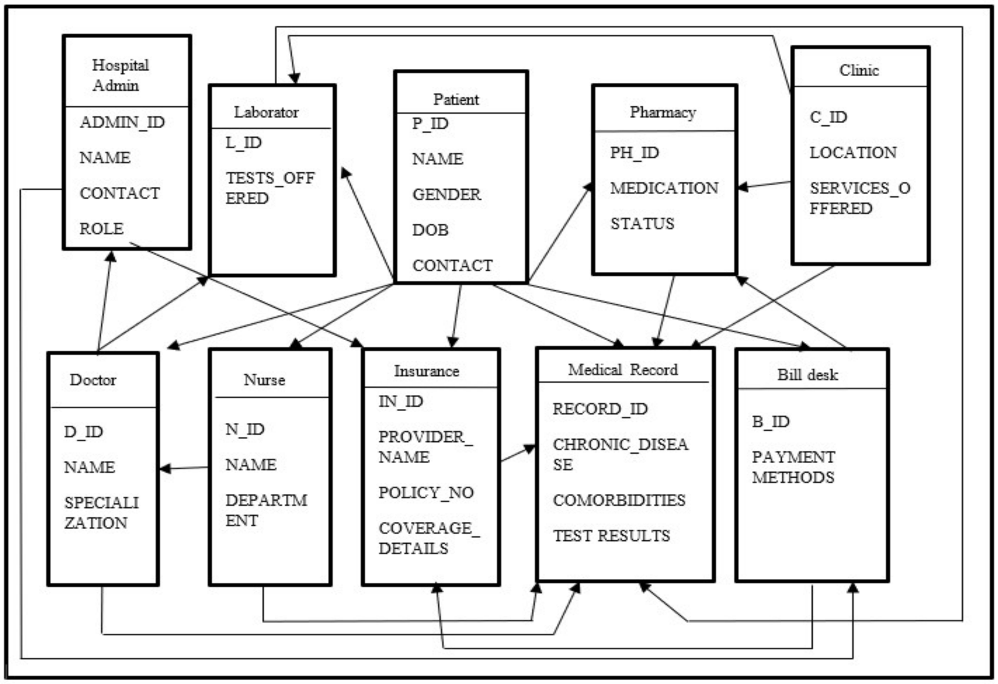
Let a relational table be defined as $T = \{(x, y)\}$ and a graph as $G = \{(n, r) \mid n \in N, r \in R\}$, where N and R represent the sets of nodes and relationships, respectively. Assume that xxx is the primary key of T. The 3NF Equivalent Graph (3EG) transformation is carried out by applying the following four rules:

a) **Tuple-to-node mapping:** Each tuple $t \in T$ is transformed into a node $n \in N$. Each node n is uniquely identified by its table name and primary key: $id(n) = \{name\ (T), x\}$.
b) **Foreign key relationships:** If y is a foreign key, a relationship $r \in R$ is created between nodes n and m. Here, $id(m) = \{name(T_y), y\}$, where T_y denotes the table for which y is the primary key.
c) **Non-key attributes:** If y is a non-key attribute, it becomes a property of the node n such that $id(n) = \{name(T), x\}$.
d) **Composite keys:** For any set of keys within a tuple t, each pairwise relationship among the keys is represented as an edge between the corresponding nodes.

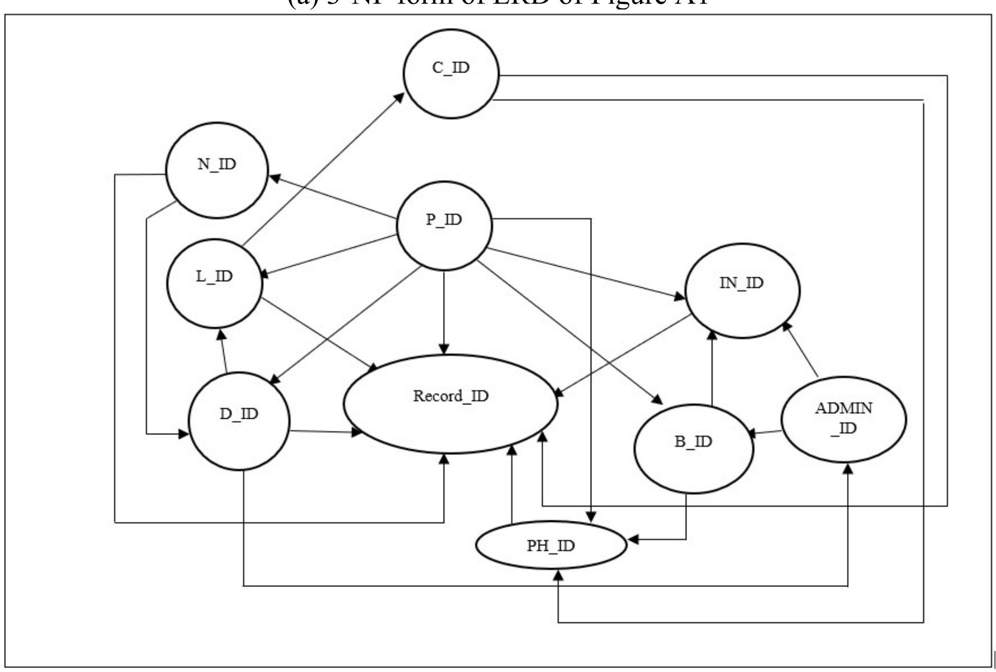
The rationale for these rules is as follows:

- Rule 1 corresponds to the principles of 1NF.
- Rule 3 reflects the 3NF condition that each non-key attribute depends on "the key, the whole key, and nothing but the key" [45].
- Rules 2 and 4 stem from the foundational definition of RDBMS.

Involving the stated rules, achieved structural foundation of the PPH 4.0 healthcare graph database is presented in following Figure 6.



(a) 3-NF form of ERD of Figure A1



(b) 3-NF equivalent graph design of Figure A1

Figure 6 Structural foundation of the PPH 4.0 healthcare database

Figure 6 (partially inspired by [44]) illustrates the structural foundation of the proposed PPH 4.0. Figure 6.(a) presents the major entities: e.g., Patient, Doctor, Nurse, Laboratory, Pharmacy, Clinic, Insurance, Bill Desk, and Hospital Administration, and maps their interactions, capturing how medical, billing, and insurance data flow within an integrated eHealth system. Figure 6.(b) represents the transformation of relational structure (refer to Figure 6.(a)) into a graph model (Figure 6.(b)), where entities are represented as nodes and their connections as edges, thereby enabling more rapid, flexible, and real-time querying of complex healthcare relationships.

Further, the designed 3NF schema for PPH 4.0 includes the design of the time-based database as suggested in [45-46].

The diagram in Figure 7 demonstrates the workflow for the proposed hybrid time-based graph database model in PPH 4.0. It depicts how the system processes input data (representing features over time) through a central computational *Black Box model*, which could encapsulate machine learning or graph-based operations. Simultaneously, metadata or auxiliary inputs M are integrated via a node n, combining both temporal and relational information streams. The output from the model includes the predicted values Y and the model-derived estimations Y_M, which are then compared through a loss function to assess performance and guide optimization. Overall, the figure represents the interaction between temporal data processing, metadata fusion, and performance evaluation for enhanced cloud-integrated data management in the PPH 4.0 environment.

A prototype for PPH 4.0 was designed involving the proposed database architecture of Figure 5 and Figure 6. For said design, React.JS, NodeJS (Express.js), HTML/CSS/JS were used, involving InfluxDB©, ® and Neo4j©, ® to support a time-based graph database in a big data scenario. Initially, the prototype was designed in a standalone system. A few (out of 18) User Interfaces of PPH 4.0 are presented in the following Figure 8. For such execution, an 12th Gen Intel® Core™ i3-1215U (1.20 GHz) processing facility with 8 GB RAM was availed.

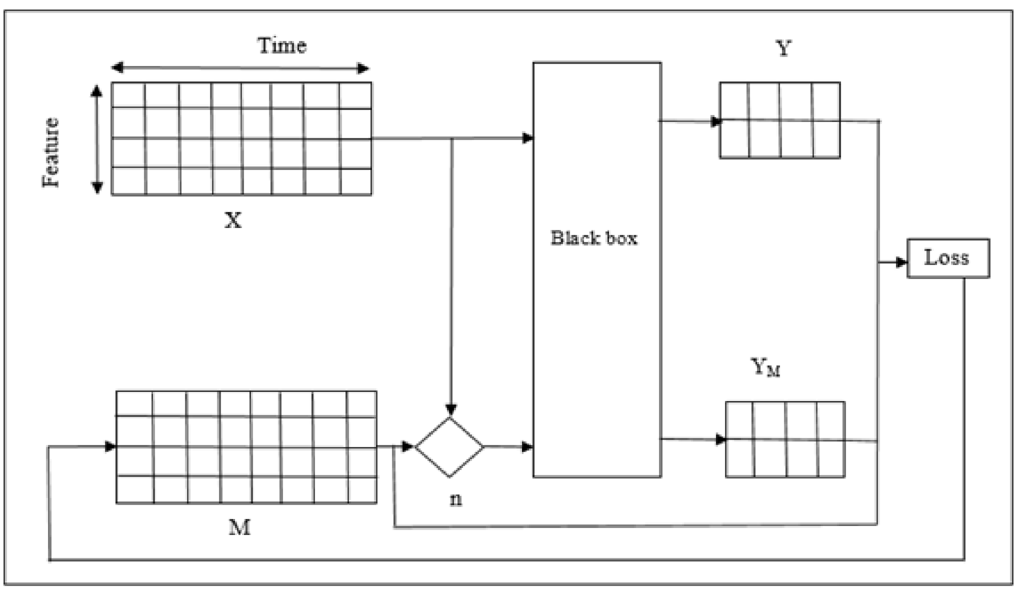
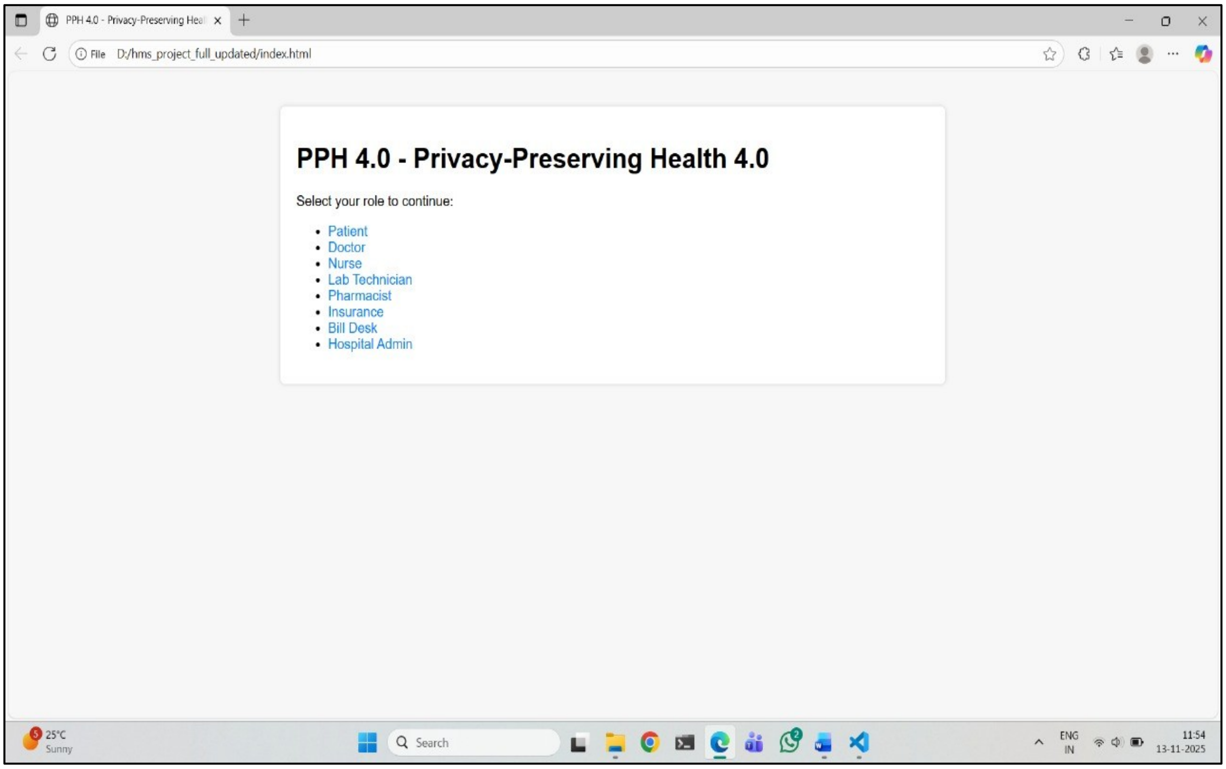
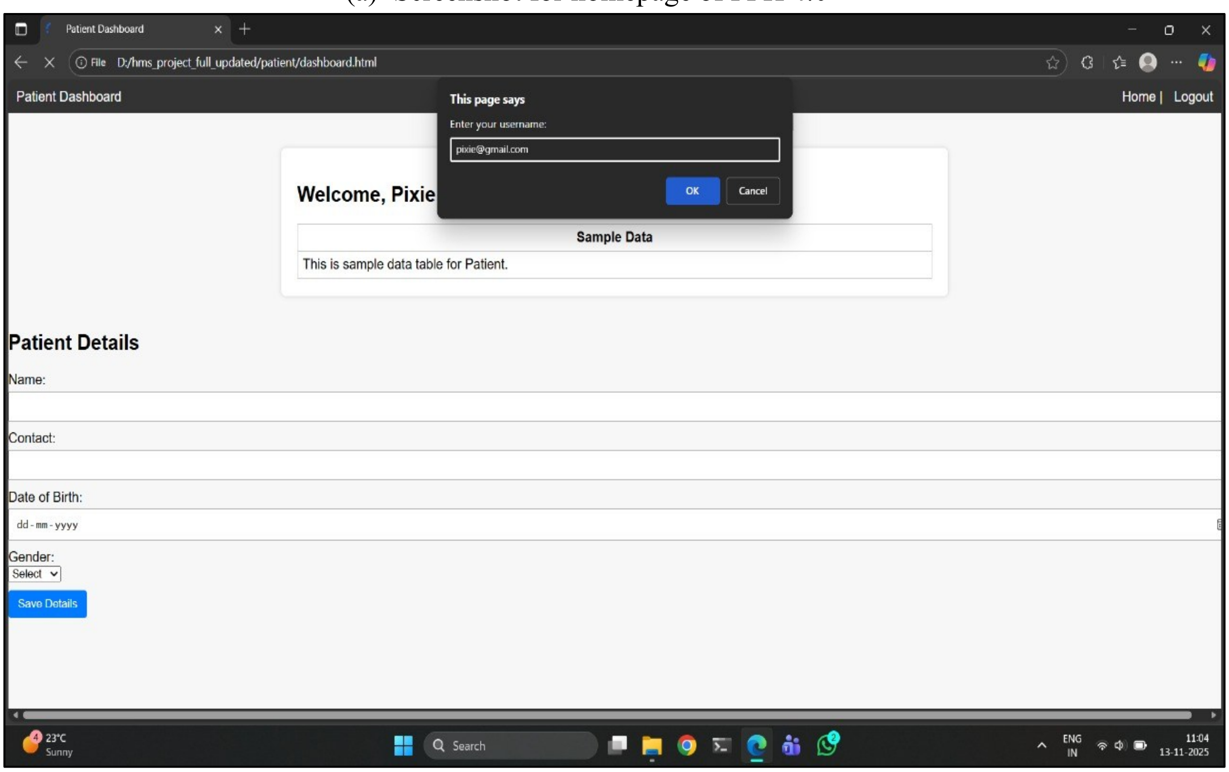


Figure 7 Conceptual workflow of a hybrid time-based graph model (partially inspired by [45])



(a) Screenshot for homepage of PPH 4.0



(b) Screenshot for the user login page of PPH 4.0

Figure 8 Screenshot for the designed user interfaces of PPH 4.0

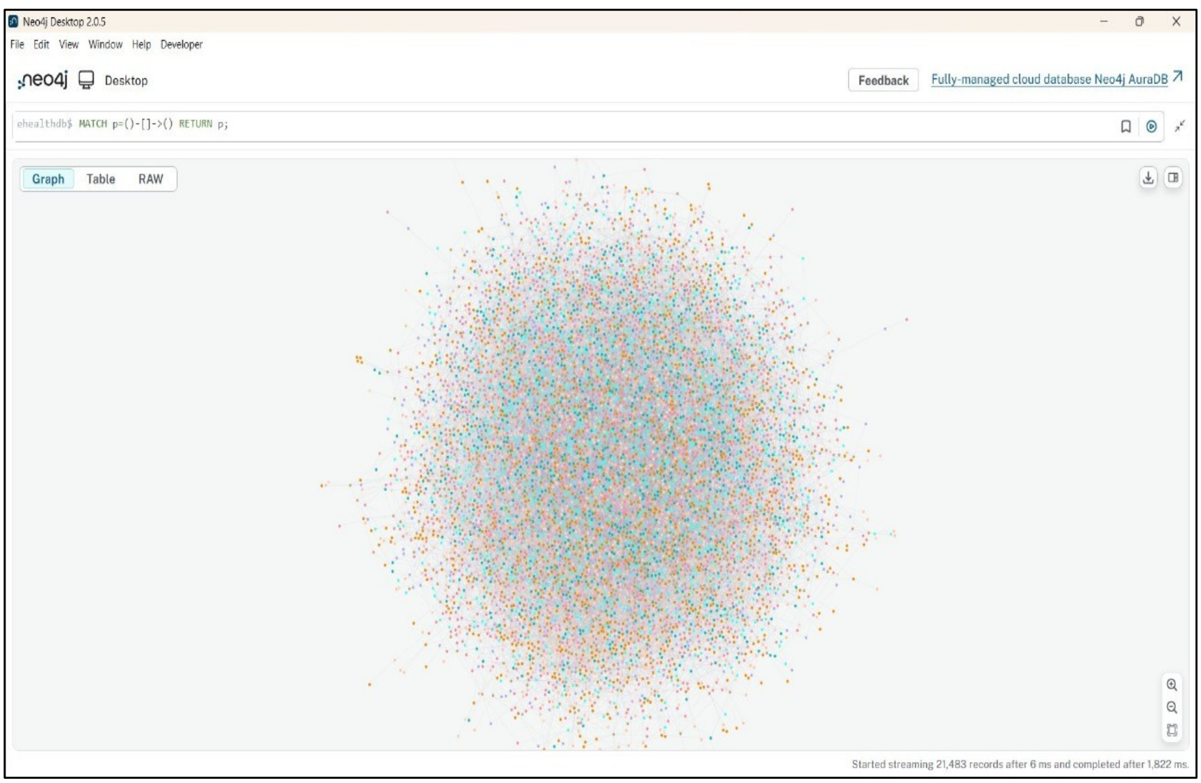
Table 1 Time and space complexities for the User Interfaces in PPH 4.0

Module	Time Complexity	Space Complexity
User Interface	Time complexity is effectively constant (i.e., $O(1)$) for individual operations.	Space complexity is effectively constant ($O(1)$) with overall resource usage depending on the size of static assets and network conditions.

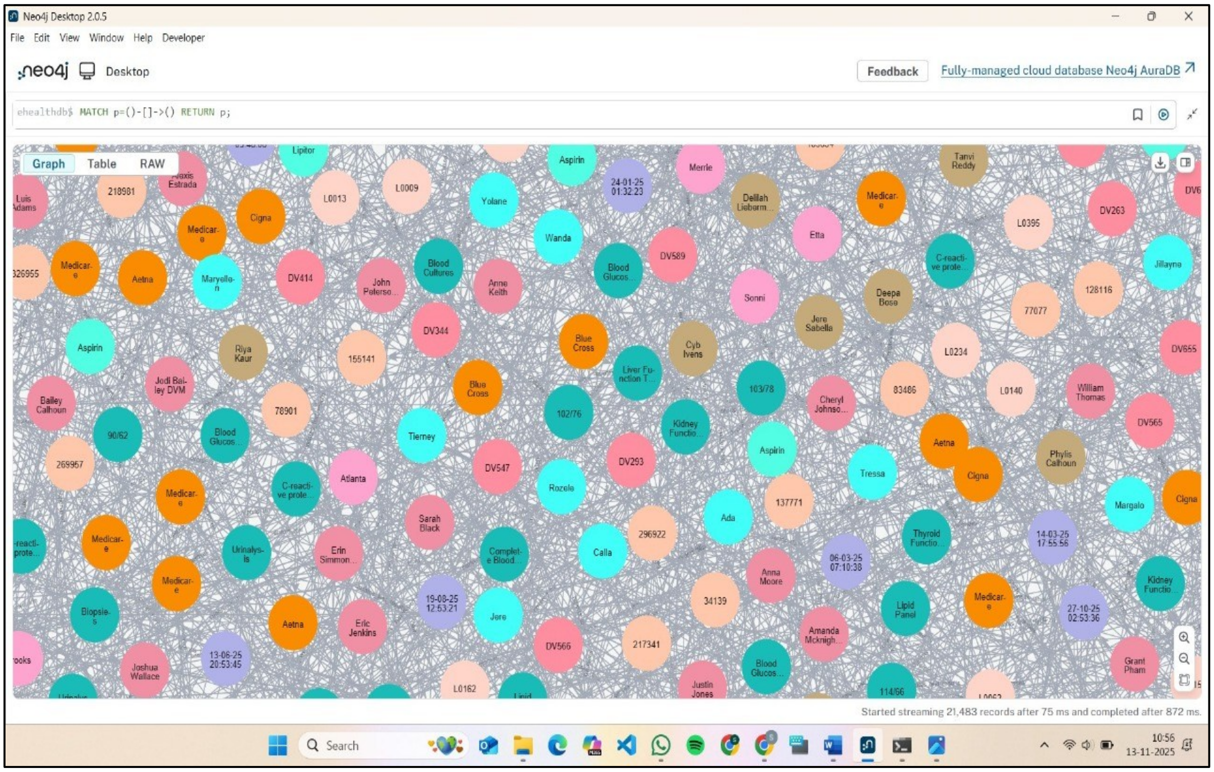
The time and space complexity measured in [47] for the code used to design the user interfaces are reported in the following Table 1.

From Table 1, it is observed that since the scenario of the code is static frontend without dynamic data processing, both time and space complexities are effectively constant ($O(1)$).

Using the user interfaces, we collected a sample data set to examine the performance of PPH 4.0. During the preparation of sample data set, we studied the data set freely available at "https://www.kaggle.com/datasets/anouskaabhisikta/healthcare-management-system", "https://www.kaggle.com/datasets/kanakbaghel/hospital-management-dataset", and "https://www.kaggle.com/datasets/prasad22/healthcare-dataset" accessed on November 08, 2025 to understand electronic healthcare records (EHRs). The sample data is further processed in Neo4j©, ® to build the target time-based graph database of PPH 4.0. Achieved graph database presented in the following Figure 9. For such execution, an AMD Ryzen 5 5500U processing facility with Radeon Graphics (2.10GHz) with 8 GB RAM was availed. The time-based graph database for PPH 4.0 is presented in the following Figure 9.



(a) Achieved full graph database of PPH 4.0 in Neo4j©, ®

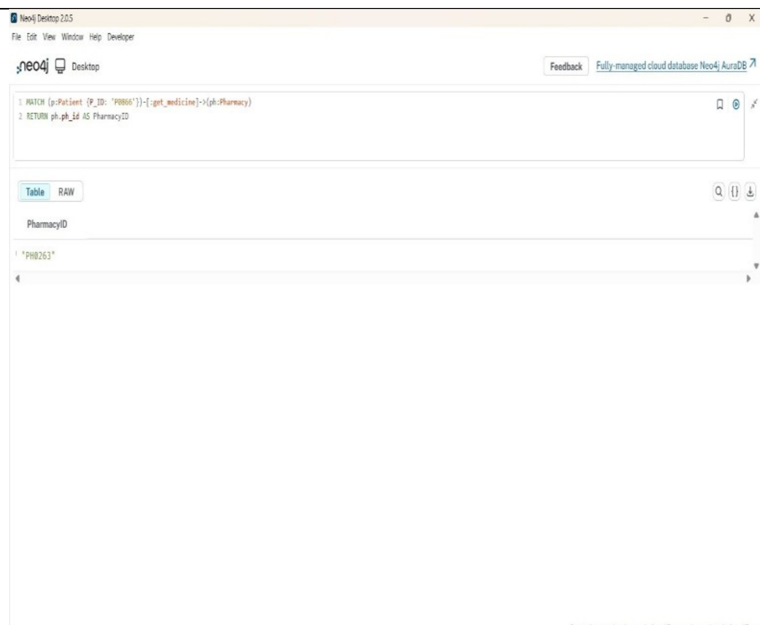


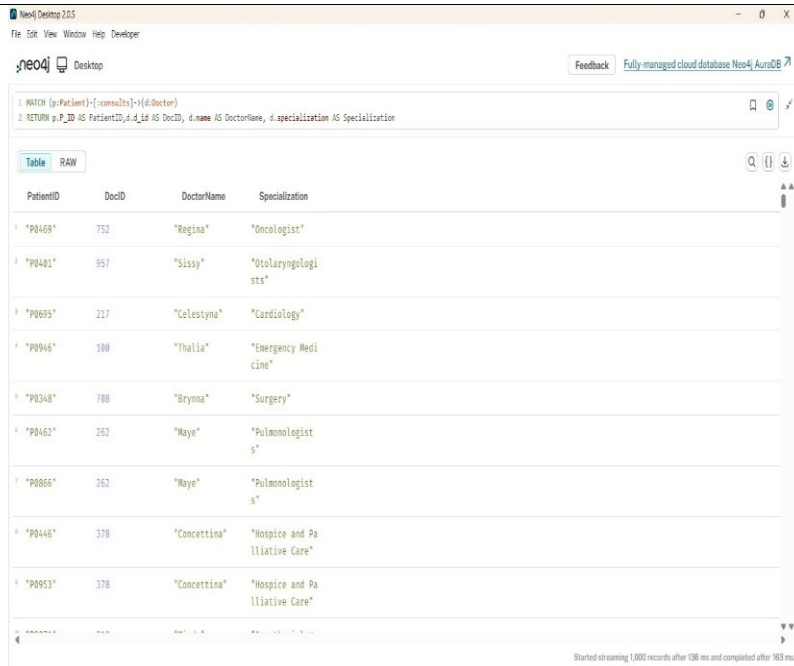
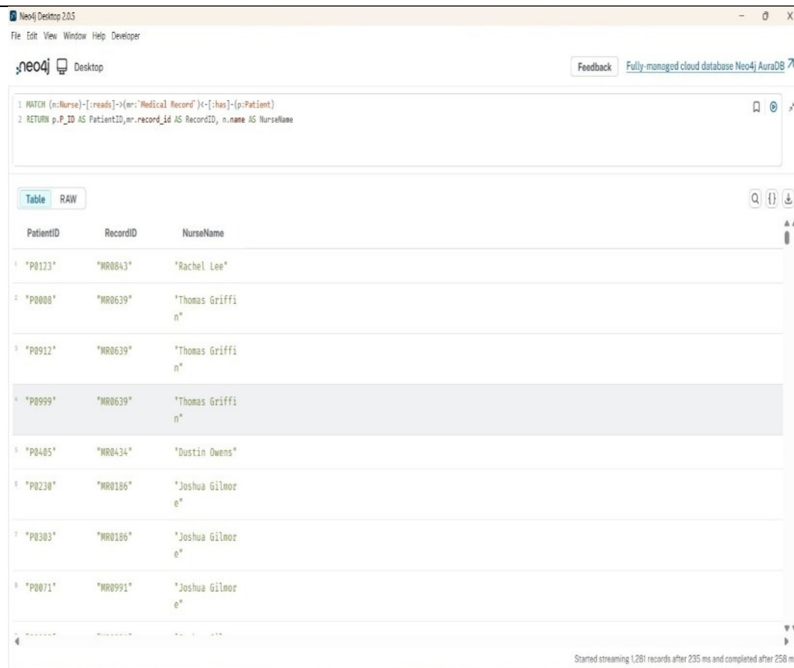
(b) An enlarged view of Figure 9.(a) (partial view)

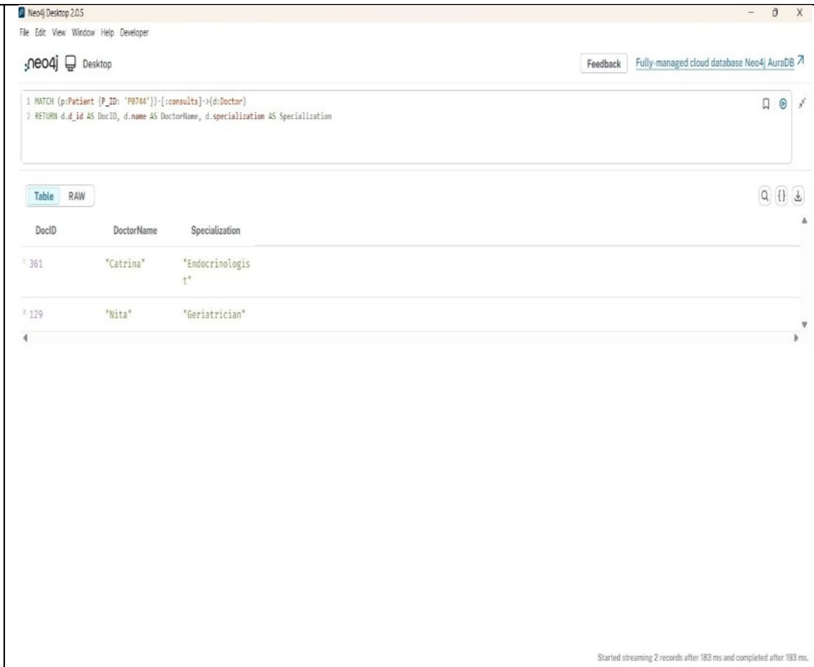
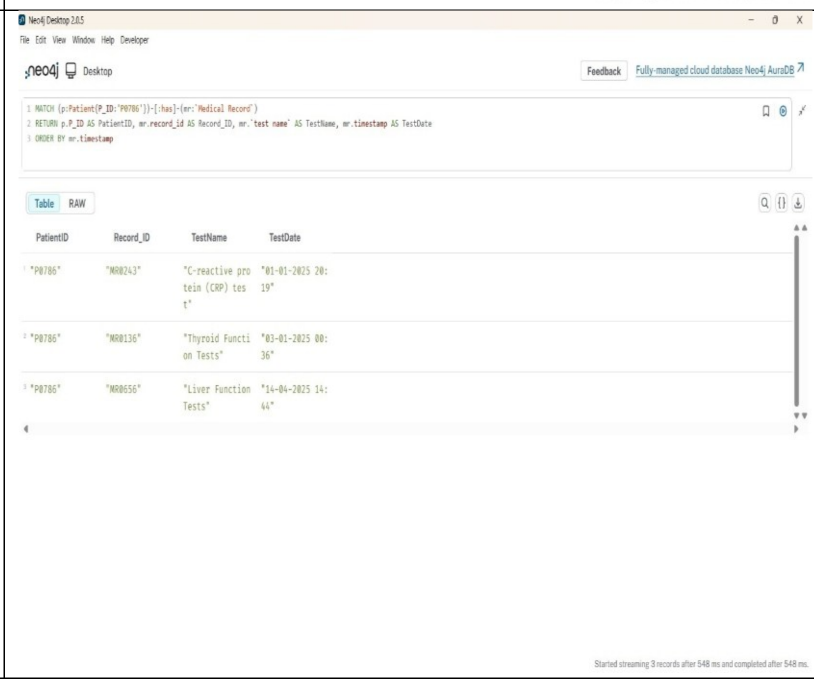
Figure 9 Screenshot for achieved graph database of PPH 4.0

Query results for such time-based graph database is presented in following Table 2.

Table 2 Query results for time-based graph database

<table>
<tr><th>Query</th><th>Query processing result</th><th>Execution time
(in mili second (ms))</th></tr>
<tr><td>All pharmacies those were visited by the patients to get medicine.</td><td>
</td><td>3</td></tr>
</table>

<table>
<tr>
<td>All the doctors consulted by all patients.</td>
<td>
</td>
<td>28</td>
</tr>
<tr>
<td>All medical records maintained by a nurse for all patients.</td>
<td>
</td>
<td>24</td>
</tr>
</table>

<table>
<tr>
<td>All doctors whom a patient has consulted, along with their specialization.</td>
<td>
</td>
<td>11</td>
</tr>
<tr>
<td>The medical records of a given patient along with the timestamp.</td>
<td>
</td>
<td>1</td>
</tr>
</table>

From Table 2, efficiency with reference to query processing time might be observed and it is found that all queries are consuming very little time (a few ms) to generate the result, though the data size is typically not small. As no other existing model in eHealth was found that involved incorporation of time-based graph database, further comparison could not be performed.

5. Conclusion, limitation and future research direction

In the present study, InfluxDB©, ®, and Neo4j©, ®, representing graph database and time series-based database respectively, were integrated to establish an efficient framework for big data storage and management within the PPH 4.0

environment. Within the broader context of big data analytics, the proposed time-oriented graph database architecture demonstrates substantial advantages by optimizing the execution of relationship-intensive queries and achieving superior performance relative to conventional database models through the direct storage of relationships. Moreover, the system exhibits improved adaptability involving a dynamic schema that accommodates continuous evolvement of data model. The proposed integration further ensures high-throughput ingestion and querying of time-stamped data, efficient storage enabled by compression mechanisms and specialized algorithms, and the inclusion of native functionalities for temporal analysis. Collectively, these features contribute to improved analytical capabilities, reduced latency in data processing, and accelerated, data-driven decision-making.

Please note that it has already been mentioned in Section 4 that the prototype was designed in a standalone system. In the near future, we plan to launch our prototype with Cloud integration and collect real-time healthcare big data, and examine the performance of PPH 4.0. Further, we plan to compare the performance of PPH 4.0 with other existing systems, if possible.

References

1) Raisa, Jasiya Fairiz, et al. "Transition towards Healthcare 5.0: Impact of COVID-19 in the healthcare industry." *ICT Express* (2025).
2) A. Mitra and A. Pal, “PPH 4.0: a privacy-preserving health 4.0 framework with machine learning and cellular automata,” Data Science in the Medical Field, pp. 1–14, Jan. 2024, doi: 10.1016/B978-0-443-24028-7.00002-7.
3) A. Mitra and A. Pal, “On signal encryption at MapReduce and collaborative attribute-based access with ECAs for a preprocessed data set with ML in a privacy-preserving health 4.0,” e-Prime - Advances in Electrical Engineering, Electronics and Energy, vol. 12, Jun. 2025, doi: 10.1016/j.prime.2025.100983.
4) L. Gomathi, A. K. Mishra, and A. K. Tyagi, “Industry 5.0 for Healthcare 5.0: Opportunities, Challenges and Future Research Possibilities,” 7th International Conference on Trends in Electronics and Informatics, ICOEI 2023 - Proceedings, pp. 204–213, 2023, doi: 10.1109/ICOEI56765.2023.10125660.
5) R. Sivan and Z. A. Zukarnain, “Security and Privacy in Cloud-Based E-Health System,” Symmetry 2021, Vol. 13, Page 742, vol. 13, no. 5, p. 742, Apr. 2021, doi: 10.3390/SYM13050742.
6) S. Timón-Reina, M. Rincón, and R. Martínez-Tomás, “An overview of graph databases and their applications in the biomedical domain,” 2021, Oxford University Press. doi: 10.1093/database/baab026.
7) R. Kang and S. Song, “Optimizing Time Series Queries with Versions,” Proceedings of the ACM on Management of Data, vol. 2, no. 3, pp. 1–27, May 2024, doi: 10.1145/3654962.
8) Y. Tong et al., “Technology investigation on time series classification and prediction,” PeerJ Comput Sci, vol. 8, p. e982, May 2022, doi: 10.7717/PEERJ-CS.982.
9) F. Di Martino and F. Delmastro, “Explainable AI for clinical and remote health applications: a survey on tabular and time series data,” Artif Intell Rev, vol. 56, no. 6, pp. 5261–5315, Jun. 2023, doi: 10.1007/s10462-022-10304-3.
10) J. Kim, H. Kim, H. G. Kim, D. Lee, and S. Yoon, “A comprehensive survey of deep learning for time series forecasting: architectural diversity and open challenges,” Artif Intell Rev, vol. 58, no. 7, Jul. 2025, doi: 10.1007/s10462-025-11223-9.
11) Lovric, Miodrag, Marina Milanovic, and Milan Stamenkovic. "Time series data mining: similarity search and its application to the stock indices in the region." *Technics Technologies Education Management* (2012).
12) S. Haben, M. Voss, and W. Holderbaum, “Time Series Forecasting: Core Concepts and Definitions,” Core Concepts and Methods in Load Forecasting, pp. 55–66, 2023, doi: 10.1007/978-3-031-27852-5_5.

13) J. Monteiro, F. Sá, and J. Bernardino, “Experimental Evaluation of Graph Databases: JanusGraph, Nebula Graph, Neo4j, and TigerGraph,” Applied Sciences 2023, Vol. 13, Page 5770, vol. 13, no. 9, p. 5770, May 2023, doi: 10.3390/APP13095770.

14) E. Bollen, R. Hendrix, B. Kuijpers, V. Soliani, and A. Vaisman, “Analysing River Systems with Time Series Data Using Path Queries in Graph Databases,” ISPRS Int J Geoinf, vol. 12, no. 3, Mar. 2023, doi: 10.3390/ijgi12030094.

15) Chen, Tao, et al. "Graph based platform for electricity market study, education and training." 2018 IEEE Power & Energy Society General Meeting (PESGM). IEEE, 2018.

16) T. Alsahfi, A. Badshah, O. I. Aboulola, and A. Daud, “Optimizing healthcare big data performance through regional computing,” Sci Rep, vol. 15, no. 1, Dec. 2025, doi: 10.1038/s41598-025-87515-5.

17) A. Badshah et al., “Use of regional computing to minimize the social big data effects,” Comput Ind Eng, vol. 171, Sep. 2022, doi: 10.1016/j.cie.2022.108433.

18) A. Tahir et al., “A systematic review on cloud storage mechanisms concerning e-healthcare systems,” Sep. 02, 2020, MDPI AG. doi: 10.3390/s20185392.

19) K. Mishra, S. Basu, and U. Maulik, “Graft: A graph based time series data mining framework,” Eng Appl Artif Intell, vol. 110, p. 104695, Apr. 2022, doi: 10.1016/J.ENGAPPAI.2022.104695.

20) Z. A. Khan, T. Hussain, F. U. M. Ullah, S. K. Gupta, M. Y. Lee, and S. W. Baik, “Randomly Initialized CNN with Densely Connected Stacked Autoencoder for Efficient Fire Detection,” Eng Appl Artif Intell, vol. 116, p. 105403, Nov. 2022, doi: 10.1016/J.ENGAPPAI.2022.105403.

21) C. Ji, Y. Hu, K. Wang, P. Zhan, X. Li, and X. Zheng, “Identifiable Temporal Feature Selection via Horizontal Visibility Graph Towards Smart Medical Applications,” Interdiscip Sci, vol. 13, no. 4, pp. 717–730, Dec. 2021, doi: 10.1007/S12539-021-00460-5/METRICS.

22) T. R. Alawiye and T. R. Alawiye, “The Impact of Digital Technology on Healthcare Delivery and Patient Outcomes,” Ehealth Telecommun Syst Netw, vol. 13, no. 2, pp. 13–22, Jun. 2024, doi: 10.4236/ETSN.2024.132002.

23) J. Ashaba and J. Nabukenya, “Beyond monitoring functionality to results evaluation of eHealth interventions: Development and validation of an eHealth evaluation framework,” Health Informatics J, vol. 28, no. 4, Oct. 2022, doi: 10.1177/14604582221141834.

24) E. Mbunge, B. Muchemwa, S. Jiyane, and J. Batani, “Sensors and healthcare 5.0: transformative shift in virtual care through emerging digital health technologies,” Global Health Journal, vol. 5, no. 4, pp. 169–177, Dec. 2021, doi: 10.1016/J.GLOHJ.2021.11.008.

25) D. Giansanti, “The Future of Healthcare Is Digital: Unlocking the Potential of Mobile Health and E-Health Solutions,” Healthcare, vol. 13, no. 7, p. 802, Apr. 2025, doi: 10.3390/HEALTHCARE13070802.

26) D. Saraswat et al., “Explainable AI for Healthcare 5.0: Opportunities and Challenges,” IEEE Access, vol. 10, pp. 84486–84517, 2022, doi: 10.1109/ACCESS.2022.3197671.

27) A. Rehman, S. Abbas, M. A. Khan, T. M. Ghazal, K. M. Adnan, and A. Mosavi, “A secure healthcare 5.0 system based on blockchain technology entangled with federated learning technique,” Comput Biol Med, vol. 150, p. 106019, Nov. 2022, doi: 10.1016/J.COMPBIOMED.2022.106019.

28) C. Lin, C. W. Yu, and N. Wang, “Introduction to the special section on enabling technologies for Healthcare 5.0,” Technology and Health Care, vol. 32, no. 6, p. 4759, Nov. 2024, doi: 10.3233/THC-248052.

29) J. F. Raisa, M. S. Rahman, I. Mahmud, M. S. Kaiser, and D. S. Han, “Transition toward Healthcare 5.0: Impact of COVID-19 in the healthcare industry,” ICT Express, vol. 11, no. 3, pp. 371–389, Jun. 2025, doi: 10.1016/J.ICTE.2025.04.002.

30) L. Yao et al., "Global trends of big data analytics in health research: a bibliometric study," Front Med (Lausanne), vol. 12, p. 1456286, 2025, doi: 10.3389/FMED.2025.1456286/FULL.

31) R. Raja, I. Mukherjee, B. K. Sarkar, and S. Ali, "A Systematic Review of Healthcare Big Data," Sci Program, vol. 2020, no. 1, p. 5471849, Jan. 2020, doi: 10.1155/2020/5471849.

32) I. J. B. Do Nascimento et al., "Impact of big data analytics on people's health: Overview of systematic reviews and recommendations for future studies," J Med Internet Res, vol. 23, no. 4, p. e27275, Apr. 2021, doi: 10.2196/27275.

33) W. N. Hussein, H. N. Hussain, and I. M. Humod, "A proposed framework for healthcare based on cloud computing and IoT applications," Mater Today Proc, vol. 60, pp. 1835–1839, Jan. 2022, doi: 10.1016/J.MATPR.2021.12.505.

34) F. Xie et al., "Deep learning for temporal data representation in electronic health records: A systematic review of challenges and methodologies," J Biomed Inform, vol. 126, p. 103980, Feb. 2022, doi: 10.1016/J.JBI.2021.103980.

35) Böhlen, M. H., Snodgrass, R. T., & Soo, M. D. (2026). Coalescing in temporal databases. In *Proceedings of the 22nd International Conference on Very Large Data Bases, Mumbai (Bombay), India (February 1996) pp. 180Y191*.

36) Chen, X., Su, N., Wang, H., Wang, X., Guan, J., Zhang, Y., & Liu, W. (2026). Digital Array Ubiquitous Radar Target Detection via Graph Representation and Spatial-Temporal Graph Fourier Transform. *IEEE Journal on Miniaturization for Air and Space Systems*.

37) Elkoumikhi, H., Madani, A., & El Ouadghiri, M. D. (2026). Personal Health Knowledge Graphs: A Comprehensive Review of Applications and Future Directions. *Smart Technologies for a Sustainable Environment*, 86-93.

38) Tan, N. H., Tuan, T. M., Chuan, P. M., Hoang, N. D., Thanh, L. Q., & Son, L. H. (2026). FKG-MM: A multi-modal fuzzy knowledge graph with data integration in healthcare. *PloS one*, *21*(1), e0339864.

39) C. Yin et al., "SepsisCalc: Integrating Clinical Calculators into Early Sepsis Prediction via Dynamic Temporal Graph Construction," KDD, vol. 2025, no. v1, p. 2779, Jul. 2025, doi: 10.1145/3690624.3709402.

40) G. M. Nagamani and C. K. Kumar, "Design of an improved graph-based model for real-time anomaly detection in healthcare using hybrid CNN-LSTM and federated learning," Heliyon, vol. 10, no. 24, p. e41071, Dec. 2024, doi: 10.1016/J.HELIYON.2024.E41071.

41) TrustRadius (2025). InfluxDB vs. Neo4j. Available at https://www.trustradius.com/compare-products/influxdb-vs-neo4j#pricing, accessed on October 30, 2025.

42) InfluxDB 3: System Architecture. Available at https://images.ctfassets.net/o7xu9whrs0u9/6sNYojmny8XlpPSSyPoMTS/ac10da2c92ecede8ccc6bd3c93ae6a72/InfluxDB-3-0--System-Architecture---OG.png, accessed on accessed on October 30, 2025.

43) Time Traveling with Graph Databases. Available at https://arangodb.com/wp-content/uploads/2019/07/ArangoDB_data-model-for-the-simple-application.png, accessed on October 30, 2025.

44) Park, Y., Shankar, M., Park, B. H., & Ghosh, J. (2014, March). Graph databases for large-scale healthcare systems: A framework for efficient data management and data services. In *2014 IEEE 30th International Conference on Data Engineering Workshops* (pp. 12-19). IEEE.

45) Kent, W. (1983). A simple guide to five normal forms in relational database theory. Communications of the ACM, 26(2), 120-125.

46) Crabbé, J., & Van Der Schaar, M. (2021, July). Explaining time series predictions with dynamic masks. In *International Conference on Machine Learning* (pp. 2166-2177). PMLR.

47) Time and Space Complexity Calculator. Available at https://www.bigocalc.com/, accessed on November 10, 2025

Annex

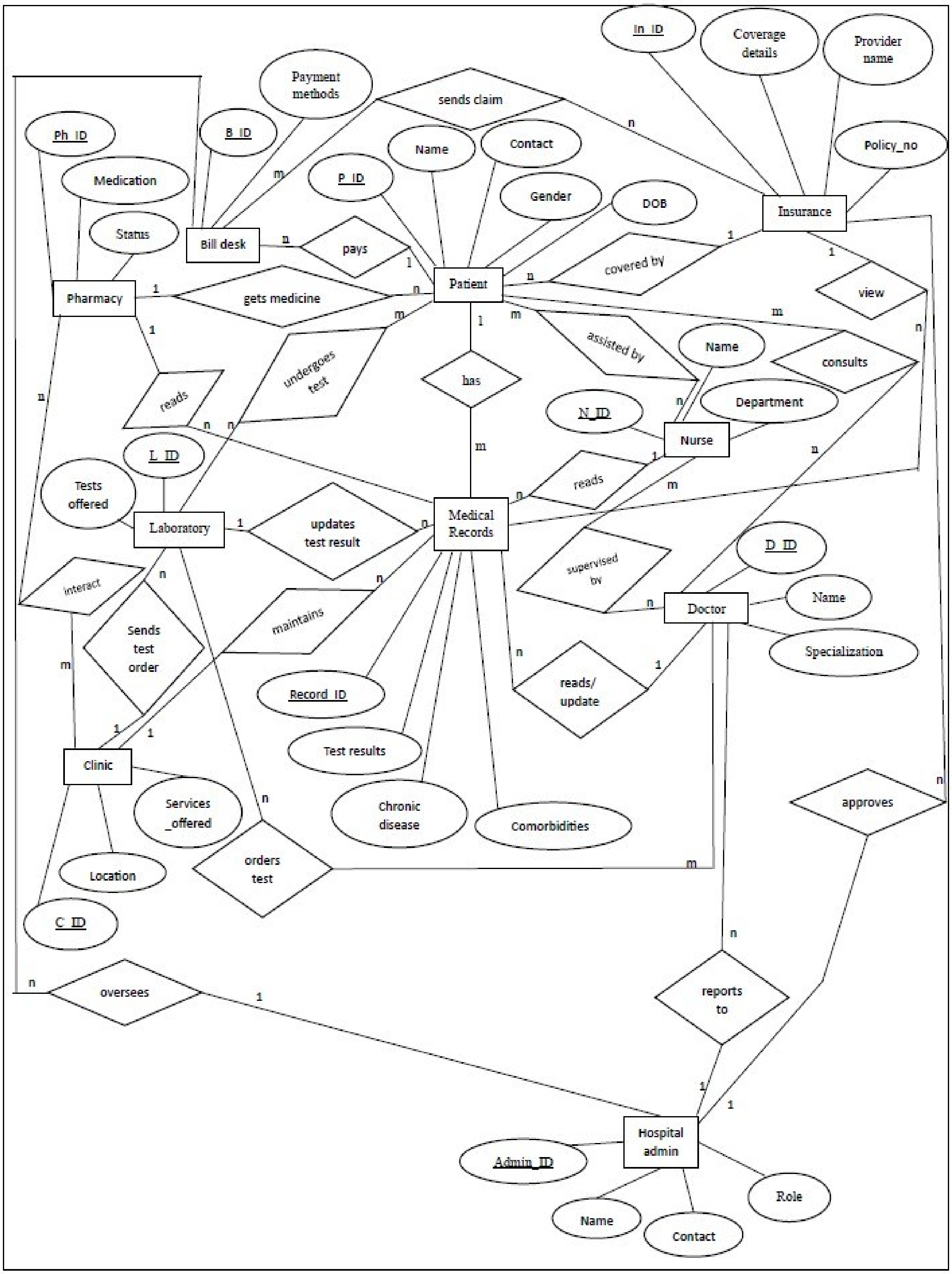


Figure A1 An entity-relationship diagram for the proposed database in PPH 4.0

From Figure A1, we can infer that the ER diagram models the relationships among the core healthcare entities: Patient, Doctor, Nurse, Laboratory, Insurance, Pharmacy, Bill Desk, Clinic, and Hospital admin. The diagram illustrates patient medical records management. Each Patient entity includes primary attributes such as PatientID, Name, and Contact Information.
The Doctor and Nurse entities possess attributes like StaffID, Name, and Department, with both entities able to access and update patient medical records. The Clinic entity manages patient appointments and sends test orders to the Laboratory, which processes these tests and updates results linked to patients' medical histories. A many-to-many relationship exists between Patients and Tests, facilitating multiple diagnostic procedures per patient. The Bill Desk entity handles billing transactions, payments, and interacts with Insurance for claims processing. The Pharmacy dispenses medicines based on prescriptions, connecting both to Patients and the billing system. The Hospital Admin entity supervises and approves operations across entities, maintaining system-wide governance. Cardinalities such as one-to-many and many-to-many accurately reflect real-world relationships—for example, a doctor may attend multiple patients, and a patient may receive care from multiple doctors or nurses.

Acknowledgement:

All authors sincerely acknowledge the guidance and helps received from Dr. Arnab Mitra and Dr. Kamalesh Karmakar, Dept. of Information Technology, Techno International New Town, Kolkata-700156, India, for completion of the project and manuscript.

Declaration of Conflicting Interests

All authors declare that no funding was obtained to carry out the research and write the manuscript. Hence, all authors declare no conflicts.

Dataset availability

Code and data used for this research are publicly available at the DOI: 10.17632/w5y33hjy79.1.

Contributors

Sweety Chatterjee
Dept. of Information Technology, Techno International New Town, Kolkata-700156, India.

Email: sweety.chatterjee.it.2022@tint.edu.in

.

Mst Hanifah Khatun
Dept. of Information Technology, Techno International New Town, Kolkata-700156, India.

Email: mst.hanifah.khatun.it.2021@tint.edu.in

Ayantika Saha
Dept. of Information Technology, Techno International New Town, Kolkata-700156, India.

Email: ayantika.saha.it.2022@tint.edu.in

Moupriya Sarkar
Dept. of Information Technology, Techno International New Town, Kolkata-700156, India.

Email: moupriya.sarkar.it.2022@tint.edu.in

Authors' contribution

Sweety Chatterjee: implemented simulation for time-based graph database, obtained results, and wrote the initial draft of the manuscript.

Hanifah Khatun: designed code for user interfaces, executed simulation, obtained results, and collected data from the results.

Ayantika Saha: designed figures 1, 2, 3, 4 and wrote a part for the initial draft of the manuscript.

Moupriya Sarkar: designed figures 5, 6, 7 and wrote a part for the initial draft of the manuscript.