Design of an Efficient Hospital Management Database System for Enhanced Data Management

Wenjie Mo[a]
International School, Jinan University, Jinan, China

Keywords: Hospital Management, Database System, Entity-Relationship, MySQL.

Abstract: The rapid advancement of digital technologies has significantly impacted the informatization of the medical industry, particularly in data management. This study aims to design and implement an efficient, secure, and reliable hospital management database system to address the limitations of traditional manual data processing methods. Utilizing relational database management system principles, the study employs Entity-Relationship (ER) diagrams to model the complex relationships among hospital entities. The ER model is then converted into a relational schema and implemented using My Structured Query Language (MySQL). The system is designed to manage key hospital data, including medical records, doctor files, and prescriptions, optimizing data storage through normalization and indexing techniques. Data for this study was generated using the Mockaroo data generator to simulate a real hospital environment. Experimental results demonstrate that the proposed system significantly enhances data retrieval efficiency and system reliability, particularly under high data loads. The system effectively reduces data redundancy, improves query performance, and supports large-scale medical data management. Overall, this study provides a scalable solution that contributes to the broader goal of digital transformation in healthcare.

1 INTRODUCTION

In recent years, with the rapid development of digital technology and the requirements of informatization, the medical industry is gradually transforming to information technology, and is facing more and more severe data management challenges (Kraus and et.al, 2021). Hospitals have the responsibility of processing and managing large amounts of data, such as hospital records, patient records, physical examination results, etc (Dash and et.al, 2019), while facing the challenge of how to effectively integrate and use data making traditional manual management methods both inefficient and error-prone (Wang and et.al, 2019). Therefore, it is inevitable to design an efficient hospital management database system to improve diagnosis and treatment efficiency, reduce human error and optimize patient experience.

With the increasing needs of medical institutions, a single function can no longer meet the increasingly complex data management requirements. In order to solve this problem, scholars have proposed a variety of methods to design management systems, and the database design method based on Entity-Relationship (ER) diagram and relational model is one of the most widely used (Storey, 1991). In this approach, ER diagrams visually show entities, relationships between entities, and multiplicity, thus helping developers to better model, understand, and design complex database structures (Li and Chen, 2009). On this basis, further standardizing data storage and management methods can better ensure the consistency and integrity of data (Connolly and Begg, 2005). For the past few years, My Structured Query Language (MySQL) has been widely used in the implementation of various management systems because of its stability and ease of use as an open source relational database management system (Rawat and Purnama, 2021). A large number of studies have shown that the relational database design method combining ER diagram and relational model can significantly improve the efficiency and reliability of the system (Kashyap and et.al, 2016), which provides strong technical support for the development of hospital management system.

[a] https://orcid.org/0009-0008-8111-8663

Mo and W.
Design of an Efficient Hospital Management Database System for Enhanced Data Management.
DOI: 10.5220/0013516300004619
In *Proceedings of the 2nd International Conference on Data Analysis and Machine Learning (DAML 2024)*, pages 311-316
ISBN: 978-989-758-754-2

The primary objective of this study is to design and implement an efficient and reliable hospital management database system to address the shortcomings of traditional methods and reduce redundancy (Racheal and Divine) (Liu and et.al, 2018). The study employs an ER diagram to model complex relationships among hospital entities such as patients, doctors, and prescriptions. This approach ensures clear and structured data representation, reducing redundancy and enhancing data consistency. The ER diagram is converted into a relational schema using the relational model and implemented with MySQL, an open-source relational database management system. MySQL was selected for its stability, scalability, and ease of use, making it well-suited for handling extensive hospital data. This method addresses issues of data integrity and retrieval efficiency, ensuring the system performs well under heavy loads. Experimental results indicate that combining the ER diagram with the relational model significantly enhances the system's data management capabilities, leading to more accurate and efficient hospital operations. The system reduces manual data entry errors and improves overall operational efficiency by providing real-time data access through query functionality. This study demonstrates the system's practical significance in simplifying hospital operations, reducing data management burdens, and ultimately improving patient care by facilitating timely medical interactions. The scalable design of the system allows for adaptation to various healthcare settings, contributing to the broader goal of digital transformation in healthcare.

2 METHODOLOGY

2.1 Dataset Description

The dataset for this study is a collection of various entities and their attributes in a practical scenario of hospital management, including those from patient records, doctor profiles, prescription details, and treatment histories. Various attributes are included such as patient demographics, medical diagnoses, prescribed medications, and consultation dates, among others. The Data set of this experiment was generated by mockaroo Data Generator (Mockaroo, 2024) and partially modified, such as patient contact information, address and other data. This data set is used to insert into the database implemented by MySQL so as to test the insertion, query, update and other functions.

2.2 Proposed Approach

The aim of this study is to design and develop a robust hospital management database system to enhance data management efficiency in the medical industry and address the limitations of traditional methods. Initially, the study employs an ER diagram to model the relationships between various hospital entities, such as patients, doctors, and prescriptions. This approach clarifies and visualizes the complex interrelationships among these entities. The ER model is then converted into a relational model to standardize data storage, ensuring consistency and reducing redundancy. The relational model employs normalization techniques to address issues such as storage space wastage and data inconsistency. MySQL is used to implement the relational model due to its stability and scalability. After implementation, the system undergoes rigorous testing to assess data retrieval efficiency and reliability under various load conditions. Evaluation results demonstrate that the system significantly improves hospital operational efficiency. The methodology pipeline for this model is illustrated in Figure 1.

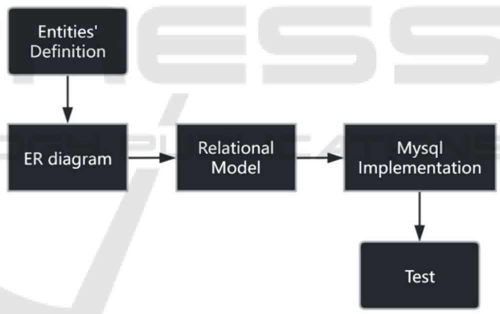

Figure 1: The pipeline of the model (Picture credit: Original).

2.2.1 ER Diagram

ER Diagram is a common technique used to design database systems. It visually describes the data structure in the system by using a graphical way, and clearly represents the entities and attributes, and the relationships between these entities. The characteristic of ER diagram lies in its intuitiveness and simplicity, which can help developers better understand and construct complex database structures in the design phase. ER diagrams have two main characteristics, one is clarity: ER diagrams present data and relationships between entities in a clear and understandable way, making them particularly useful for capturing the structure of complex systems. The other is a single view: unlike functional models, ER diagrams usually provide a single view showing all entities and their relationships without further decomposition, which makes the whole system easier to understand.

There are two reasons for using ER diagram in this study. One is that ER diagram serves as the basis of data modeling and can accurately define entities and their relationships in the database, thus providing a clear blueprint for subsequent database design and implementation. The second is that ER diagrams provide an application-independent view that can be verified by the user and translated to the design of any type of database management system. The structure of ER diagram is composed of entities, attributes and relationships. An entity is represented by a table, an attribute is the column of the table of its corresponding entity, a relationship is a diamond connecting two entities, and the multiplicity of an entity is shown in the thin line between the entity and the relationship.

This study firstly collects the business description of general hospitals to obtain 11 entities and their attributes that may be included in the ER diagram, and then identifies the primary key and candidate key in the attribute, and then identifies the relationship between each entity and determines the multiplicity. After these steps, the ER diagram is complete. The ER diagram with reduced attributes is shown in Figure 2.

2.2.2 RM

The RM is a theoretical framework for structuring and managing data in a database system, originally proposed by E.F. Codd (Codd, 1970). This model organizes data into relations, which are essentially tables consisting of rows and columns. Each table, or relation, represents an entity, with rows as records and columns as attributes. The RM ensures data consistency, supports data integrity, and facilitates efficient data retrieval by allowing operations such as selection, projection, and join to be performed on these tables.

One feature that RM has is normalization: RM uses normalization techniques to organize data to reduce redundancy and maintain data integrity. This process consists of dividing the large table into small tables and defining the relationships between them. Another distinguishing feature is data independence: One of the main advantages of RM is its ability to provide data independence, which means that changes to the physical storage of data do not affect

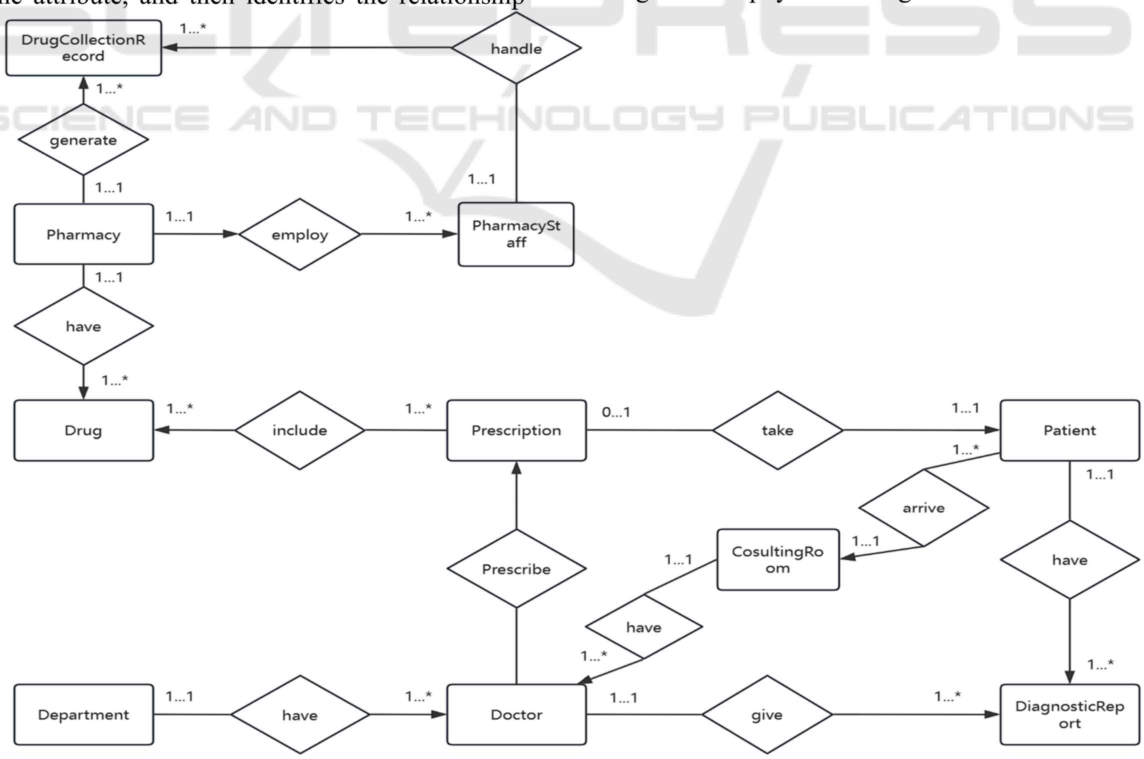

Figure 2: ER diagram with reduced attributes (Picture credit: Original).

the logical structure of the database. This allows applications to be unaffected by changes in data representation or storage methods. RM is used in this study for two reasons. The first is that RM uses primary and foreign keys to ensure data integrity and consistency across tables, thus accurately maintaining relationships between data. The second is that efficient query processing can be implemented in RM due to its structured format and allows for fast and reliable execution of complex queries.

For the structure of the RM, the table is the basic unit of data in the RM, and each table represents a set of entities. Each table has a Primary key (PK) that is used as a unique identifier for each record in the table, ensuring that there are no duplicates. The Foreign key (FK) joins the tables by referring to the PK of the other tables, thus implementing the relationship between each table.

To convert the ER diagram to RM in this experiment, each entity is first transformed into a relational schema, and the attributes of the entity are the attributes of the relational schema. It then classifies the type of relationship between every two entities, which can be divided into three types: one-to-one, one-to-many, and many-to-many. For a one-to-one relationship, the PK of an entity at either end is added to the table of the entity at the other end as FK. For one-to-many relationships, the PK of one end entity is taken as FK of many end entities. For a many-to-many relationship, an intermediate table is created to hold the primary keys of the two entities and the necessary attributes. After these steps are completed, the preliminary RM is obtained.

Next, the normalization operation is entered, the database redundancy will be reduced with the increase of normal form (NF), but too high NF will lead to increased query complexity and increase the overhead of update operation, etc. Therefore, 3NF is the most balanced scheme in this study. 1NF first ensures atomicity of the table, every column of the table cannot be decomposed. 2NF eliminates the partial dependence of non-primary attributes based on 1NF. The transitive dependency is subsequently eliminated thus completing the 3NF. After these steps, the complete RM is obtained.

2.2.3 MySQL Implementation

MySQL is a popular open-source SQL database management system developed by Oracle Corporation. MySQL has the ability to manage structured data collections and to add, access, and process data stored in the database (Christudas, 2019). MySQL was chosen for this study precisely because of its stability, scalability, and ease of use, its strong support for relational databases, and its ability to process large amounts of data efficiently.

To implement this system using MySQL, first convert RM into a SQL statement compatible with MySQL environment, define the table, set the primary key foreign key, and then create the table using MySQL's Data Definition Language (DDL) and specify the appropriate data type for each column. In this study, the appropriate data type is used to ensure data integrity and accuracy, and improve storage efficiency and query performance. In the third step, PK and FK indexes are added to improve the query performance. After a series of table creation, the fourth step uses MySQL's Data Manipulation Language (DML) to insert data to simulate the actual hospital management scenario. The fifth step is to write Data Query Language (DQL) and execute it. These queries are used for common retrieval and update operations that may appear in hospital management, which is a big means to improve the efficiency of hospital operation. Through such a series of complete steps, the hospital database management system is effectively completed in the MySQL environment.

3 RESULT AND DISCUSSION

In the result section, two key results of the hospital management database system will be presented: the structure of partial tables, the results of data query. These results show in detail how the system can deal with complex operations in hospital management safely and efficiently under the guarantee of data integrity through appropriate design.

3.1 The Structure of Partial Tables

The “Patient” table stores information data for “Patient” entities. The primary key is Patient ID, Since multiple patients may visit is the same Consulting Room, resulting in a one-to-many relationship, the Consulting Room ID is referenced as the foreign key of the table to be associated with the “Consulting Room”,and the structure design result is shown in Table 1.

Table 1: The structure design result of “Patient” table.

Field Name	Type	Byte	Field Name	Type	Byte
Patient ID	varchar	20	Contact Information	varchar	100
Patient Name	varchar	50	Medical History	varchar	1000
Gender	char	1	Address	varchar	200
Age	tinyint	1	Consulting Room ID	varchar	10

“Doctor” table stores information data for “Doctor” entities. The primary key is Doctor ID. Since there are two one-to-many relationships in the “Doctor" table, a Consulting room can be visited by different doctors at different time periods, and there are multiple doctors in a department. So in the table reference Consulting Room ID as a foreign key is associated with “Consulting Room” and reference Department ID as a foreign key is associated with “Department”, and the structure design result is shown in Table 2.

Table 2: The structure design result of “Doctor” table.

Field Name	Type	Byte	Field Name	Type	Byte
Doctor ID	varchar	10	Contact Information	varchar	100
Doctor Name	varchar	50	Consult Room ID	varchar	8
Title	varchar	100	Department ID	varchar	4

Multiple drugs may be included in a prescription, and a drug may also appear in different prescriptions, thus forming a many-to-many relationship. To efficiently store information about this relationship and optimize query performance, an intermediate table “PrescriptionDrug” is created. This table references Drug Name and Prescription ID as a compound primary key and foreign key to simplify the table structure and associate the table with “Prescription” and “Drug”,and the structure design result is shown in Table 3.

Table 3: The structure design result of “PrescriptionDrug” table.

Field Name	Type	Byte	Field Name	Type	Byte
Prescription ID	varchar	20	Drug Name	varchar	200
Drug Quantity	tinyint	1			
Field Name	Type	Byte	Field Name	Type	Byte

3.2 Data Query Results

In this study, DQL is written to simplify and optimize query performance by querying specific data, such as all doctors and consultation rooms in a given department, and searching for drugs for a specific diagnosis.

3.2.1 Query All the Doctors in the Designated Department and Their Consultation Rooms

This query checks the names of all doctors who work in a particular department, as well as the location of their consulting rooms. Taking the data corresponding to the query department ID= "01" as the experiment, the experimental results shown in Table 4. show that the system can efficiently connect the data in the "doctor" table and the "consultation room" table based on the department ID.

Table 4: Query results for all doctors and their Consulting Rooms in the Department with Department ID “01”.

Department ID	Doctor Name	Consulting Room Location
01	Dr.Alice	Building 3, Room 101
01	Dr.Charlie	Building 3, Room 103
01	Dr.Jackson	Building 3, Room 107

3.2.2 Query the Drug Prescribed for a Given Diagnosis

The query focuses on showing the corresponding drugs prescribed by a given Diagnosis, thus helping the doctor to better analyze the disease. The experiment uses “Eczema” as the specified Diagnosis, shown in Table 5. The experimental results show that the system can effectively and accurately search the data in “Prescription” and “Diagnostic Report” table according to Diagnosis.

Table 5: Query result of querying the drugs prescribed for diagnosis “Eczema”.

Diagnosis	Drug Name
Eczema	Loratadine syrup
Eczema	Cetirizine hydrochloride capsules
Eczema	Cetirizine hydrochloride capsules
Eczema	amoxicillin capsule
Eczema	Cetirizine hydrochloride capsules

4 CONCLUSIONS

This study presents an efficient, secure, and reliable hospital management database system designed to overcome the limitations of traditional data management methods. By using ER diagrams to visually model complex relationships among hospital entities, and applying relational models to standardize data storage, the system effectively reduces data redundancy. Implemented with MySQL, the database ensures data integrity and optimizes query performance. Experimental results demonstrate that the system significantly enhances data retrieval and processing speed. It effectively minimizes data redundancy, provides real-time access to hospital data, and substantially improves operational efficiency. Future research will focus on expanding the system’s scalability to accommodate larger datasets and more complex hospital operations. Additionally, integrating advanced data analysis tools will be explored to offer deeper insights into healthcare management and further advance the informatization of healthcare services.

REFERENCES

Christudas, B., 2019. MySQL. Apress, 877-884.

Codd, E.F., 1970. A relational model of data for large shared data banks. Communications of the ACM, 13(6), 377-387.

Connolly, T.M., Begg, C.E., 2005. Database systems: a practical approach to design, implementation, and management. Pearson Education.

Dash, S., Shakyawar, S.K., Sharma, M., et al. 2019. Big data in healthcare: management, analysis and future prospects. Journal of big data, 6(1), 1-25.

Kashyap, N.K., Pandey, B.K., Mandoria, H.L., et al. 2016. A review of leading databases: Relational & non-relational database. i-Manager's Journal on Information Technology, 5(2), 34.

Kraus, S., Schiavone, F., Pluzhnikova, A., et al. 2021. Digital transformation in healthcare: Analyzing the current state-of-research. Journal of Business Research, 123, 557-567.

Li, Q., Chen, Y.L., 2009. Entity-relationship diagram. Modeling and analysis of enterprise and information systems. Springer, Berlin, Heidelberg, 125-139.

Liu, Y., Zeng, X., Zhang, K., et al. 2018. Transforming entity-relationship diagrams to relational schemas using a graph grammar formalism. IEEE International Conference on Progress in Informatics and Computing, 327-331.

Mockaroo., 2024. Mockaroo: Random Data Generator. Retrieved on 2024, Retrieved from: https://www.mockaroo.com/.

Racheal, A.S., Divine, O.O. The Development of a Relational Database Management System For Uniongate Diagnostic Center Akure Ondo State Nigeria.

Rawat, B., Purnama, S., 2021. MySQL Database Management System (DBMS) On FTP Site LAPAN Bandung. International Journal of Cyber and IT Service Management, 1(2), 173-179.

Storey, V.C., 1991. Relational database design based on the Entity-Relationship model. Data & knowledge engineering, 7(1), 47-83.

Wang, X., Williams, C., Liu, Z.H., et al. 2019. Big data management challenges in health research—a literature review. Briefings in bioinformatics, 20(1), 156-167.