

**Database Solution for Lakeridge Community Hospital**

**LRCH ADVANCEMENT PROPOSAL**

**Group 1**

Addan Zahra: 100883421

Neilavan Vijayakanthan: 100787121

Himanshu: 100898751

Manu Sugunakumar: 100748877

**Table of Contents**

**Title Page ---------------------------------------------------------------------------------Pg 0**

**Table of Contents -----------------------------------------------------------------------Pg 1**

**Introduction ------------------------------------------------------------------------------Pg 2**

**Mission Statement ----------------------------------------------------------------------Pg 2**

**Dataflow Diagrams ---------------------------------------------------------------------Pg 3**

**Level 0 Diagram – Pg 3**

**Level 1 Diagram – Pg 4**

**Functional Requirements -------------------------------------------------------------Pg 5**

**Non-Functional Requirements -------------------------------------------------------Pg 8**

**3NF Relational Schema ---------------------------------------------------------------Pg 11**

**ERD Model -----------------------------------------------------------------------------Pg 13**

**Data Dictionary ------------------------------------------------------------------------Pg 14**

**Queries ----------------------------------------------------------------------------------Pg 17**

**Database and Tables Creation Query – Pg 17**

**Data Insert Query – Pg 21**

**Room Utilization Query – Pg 28**

**Physician-Patient Details – Pg 29**

**Query Used in Demo – Pg 32**

**Conclusion ------------------------------------------------------------------------------Pg 33**

**Introduction**

Lake Ridge Community Hospital (LRCH) is facing several challenges with existing information systems. The hospital wants to increase its capacity from 100 to 200 beds in order to better serve its expanding community of almost 400,000 people. The hospital is divided into two groups, physicians, and administrators. The current batch-oriented information systems are unable to support medical staff, treatment recommendations, fast system-wide access, cost tracking, or flexibility to adapt to changing needs. Based on this business problem, the key requirements that we understand include patient management, physician management, physician-patient reporting, revenue management, room utilization, patient billing, database design and development, SQL queries, formal report and presentations, and collaboration and peer evaluation.

The systematic approach we took to understand the business problem was interviewing stakeholders, including the professor acting as a client or stakeholder. The strategy calls for many deliverables at various project phases and a final presentation to LRCH to persuade them of the acceptability of the suggested solution. This strategy makes sure that we as a team fully comprehend the business issue, compiles crucial information, and offers a thorough database solution that takes LRCH's requirements into account while maintaining openness and transparency in communication.

**Mission Statement**

“*Our goal in building the database for Lake Ridge Community Hospital is to provide a strong and effective information system that supports the hospital's dedication to excellence, empowers medical staff, and improves patient care. Our mission is to provide a database system that effortlessly combines information from different medical departments while maintaining accuracy, usability, and security*.”

*BREIF EXPLANATION*

Our main goal is to deliver a comprehensive and user-friendly database that satisfies the changing requirements of the hospital's medical personnel, administrators, and patients. We are committed to enhancing decision-making through data-driven insights, process efficiency, and improved data management.

Our goal is to provide a trustworthy, scalable, and sustainable solution that is in line with Lake Ridge Community Hospital's vision of offering the community high-quality healthcare services. We are firmly committed to transparency, collaboration, and adherence to best practices in database development.

We hope to contribute to the hospital's success via innovation and ongoing development by facilitating effective operations, fostering research projects, and eventually enhancing patient outcomes. Our goal is to build a database that not only takes care of the hospital's current needs but also sets it up for expansion and excellence in healthcare delivery in the future.

**Data Flow Diagrams**

**Level 0**



**Level 1**



**Functional Requirements**

1**. Patient Information Management:**

***Data Input and Modification*:** Employees in the Admissions Office should be able to enter and update detailed patient data quickly and easily with the system's user-friendly interface. Name, address, phone number, insurance information, and medical history of the patient should all be included.

***Data Amendments and History*:** A patient's record must be automatically updated with any modifications in the event of their readmission. This makes sure that the patient's medical history is accurately and completely recorded in the database.

**2. Handling of Physician Information:**

***Physician Database*:** The system ought to keep an extensive database of recommending doctors that includes specifics like their complete names, phone numbers, and email addresses, in addition to their medical specializations.

***Patient-Physician Association***: Doctors should be able to check patient information, appointment schedules, treatments received, and notes particular to each patient by accessing patient lists linked to their recommendations.

3. **Physician-Patient Report:**

***Customized Daily Reports*:** Making customized daily reports for every referring physician should be made possible by the system. Each patient who is currently admitted to LRCH and was referred by that doctor should be listed in detail in these reports.

***Patient Specifics*:** A thorough summary of the patient's medical history, including names, dates of admission, appointments, and treatment records, should be included in every report. Additionally, any pertinent notes regarding the patient's health should be included.

**Maintaining Laboratory Results: -**

***Reporting and Recording***: The system should allow medical staff to record and report the results of laboratory tests and procedures and also link them with the patient’s information using the unique patient ID.

**Maintaining Prescription: -**

***Prescription Records***: - Each prescription written for the patient should be stored and linked with patient data, allowing future medical care to look at the patient record.

4**. Daily Revenue Report:**

***Transaction Records*:** Every single revenue-generating transaction that took place on a given day must be included in the daily report that the system is mandated to automatically generate. Information about the patient, the expenses incurred, and the associated cost centers must all be included in these transactions.

***Revenue Monitoring*:** The day-to-day financial performance of the hospital should be closely monitored with the help of the daily revenue report. Trends and possible trouble spots can be identified with its help.

**5. Room Utilization Report:**

***Occupancy Overview*:** A daily report with a detailed summary of hospital room occupancy should be automatically generated by the system. It ought to specify the kind of room—private or semi-private, for example—as well as the patients who are presently using it.

***Bed and Room Status*:** The report should also include important details about the bed discharges that are planned for the day and the number of empty rooms broken down by type of room.

6. **Revenue Analysis:**

***Weekly Financial Report*:** The system needs to be able to generate a weekly financial report that is thorough and provides a comprehensive overview of the total revenues generated by different cost centers. Revenue should also be broken down into payment categories such as insurance companies, self-pay, and other sources.

**7. Patient Billing:**

***Billing Statement Generation*:** For every patient, the system ought to automatically produce comprehensive billing statements. In addition to providing an itemised record of services performed throughout the patient's hospitalisation, these statements should rigorously explain all charges linked with particular cost centres.

***Balance Calculation*:** Prior to applying insurance coverage or before the patient makes a payment, billing statements must show the outstanding balance. The patient's debt to the hospital is represented by this sum.

**8. Data Archiving:**

***Record Retention*:** For a least of three years after the patient's final discharge, the system should keep the patient records in the active database. By doing this, it is certain that past patient data will always be accessible through the database.

***Archiving Policy*:** Patient records that have not been readmitted in the last three years ought to be archived and deleted from the live database. Optimizing database performance and storage is the goal of this archiving process.

**9. Security and Access Control:**

***User Authentication and Authorization*:** To confirm users' identities when they access the database, the system needs to have a strong user authentication procedure in place. Additionally, it must offer various degrees of authorization and access to various user roles and duties.

10. **Treatment Management:**

***Treatment Recording*:** All medical treatments and procedures that patients receive during their hospital stay should be comprehensively recorded and managed by the system. The kind of treatment, the day it was administered, and any related medical staff, are all included in this.

***Cost Center Linkage*:** To accurately track healthcare costs and ensure financial accountability, clerks and medical personnel should be able to link each treatment with the appropriate cost center

**Non-Functional Requirements**

**1. *Performance***: Performance is the capacity of the system to carry out operations and produce outcomes in a reasonable amount of time. It indicates that operations like getting patient records, producing reports, or handling transactions ought to happen quickly in the context of the LRCH database. For example, in order to facilitate fast decision-making and patient care, the system has to react quickly when a healthcare practitioner searches up a patient's medical history.

**2. *Security***: Security includes all of the precautions used to safeguard private patient information kept in the database. This covers audit logging, access control, and data encryption. Preventing data breaches and illegal access guarantees that patient information may only be accessed and modified by authorized people.

**3. *Reliability***: The capacity of the system to operate without interruption and its consistent availability are the main aspects of reliability. Due to the ongoing nature of hospital activities, the LRCH database must be always accessible with little downtime. Reliable data backup and recovery procedures should also be in place to guard against data loss in the case of unplanned system failures.

**4. *Usability***: Usability is the measure of a system's ease of use and intuitiveness for healthcare staff. In this case, taking into account different degrees of technical skill, the database's user interface should be created with the unique requirements of healthcare professionals in mind. It is imperative to furnish personnel with adequate training and assistance to enable them to traverse the system with confidence.

**5. *Scalability***: Scalability is the ability of the system to grow in terms of functionality, user base, and data volume. The database has to grow without interruption to accommodate the additional data and performance requirements as LRCH prepares to grow from 100 to 200 beds.

**6. *Compatibility***: The system's compatibility refers to how well it integrates with other medical software and hardware. The LRCH database ought to function well with other programs that are already in place, including billing software, medical equipment, and other databases.

**7. *Regulatory Compliance***: Adherence to healthcare data rules and standards, including HIPAA and local healthcare legislation, is ensured by regulatory compliance with the database system. Patient confidentiality and privacy are protected by this provision, which ensures that patient data is handled in compliance with ethical and legal standards.

**8. *Reporting***: The capacity of the system to produce different reports precisely and on time is known as reporting. Financial analysis, compliance, and hospital management all depend on these reports. To enable the creation of reports with accurate data and promote well-informed decision-making, the database must support it.

**9. *Data Archiving and Retention***: The procedures for keeping and preserving patient records are outlined in data archiving and retention. Patient records that are not readmitted within three years should be archived in the LRCH database to facilitate orderly and effective data retrieval. This stipulation guarantees the preservation and accessibility of past patient data when required.

**10. *Resource Utilization***: Optimizing the usage of physical resources, such as servers and storage, is known as resource utilization. Resource allocation by the database should be done effectively to satisfy system needs, particularly during spikes in activity. It guarantees timeliness and cost-effective performance.

**11. *Change Management***: The flexibility of the system to changing management demands and external reporting requirements is addressed by change management. Changes are common in healthcare settings, and the database needs to adapt to new requirements without interfering with operations or compromising data integrity.

**12. *Data Integrity***: Ensuring the accuracy, consistency, and reliability of data in the system is known as data integrity. To avoid mistakes and corruption, the data entered by users should be validated by the system. To preserve data quality, it imposes restrictions and standards on data.

**13. *Concurrency Control***: For multi-user support, concurrency control is essential. Multiple users must have simultaneous access to the system in a healthcare environment. This criterion makes sure that when numerous users access, alter, or retrieve information at the same time, the data stays consistent.

**14. *Data Recovery Time***: The amount of time needed to fully return the system to operation following a breakdown is known as the data recovery time. This is essential in a hospital database to reduce interruptions and guarantee ongoing patient care. For a speedy recovery, the system must have a well-defined Recovery Time Objective (RTO).

**15. *Database Backup Frequency***: The frequency of database backups determines how frequently they are carried out. To prevent data loss in the case of hardware failures, data corruption, or calamities, regular, planned backups are vital.

**16. *User Permissions***: Database access is controlled by user permissions. Who may access what data should be granularly controlled by the system. To protect data security, administrators should be able to provide distinct user roles varied access privileges.

**17. *Data Accessibility***: The system's ability to support users with impairments is ensured via data accessibility. Making the database accessible to all users, including those with impairments, is crucial in the healthcare industry in order to comply with regulatory requirements for accessibility compliance.

**18. *User Feedback and Satisfaction***: In order to assess users' satisfaction with the system, feedback from users must be gathered. In order to make sure that the database satisfies the requirements and expectations of healthcare personnel, periodic surveys and feedback systems aid in identifying areas for development.

**19. *Healthcare Data Privacy***: The privacy of healthcare data is crucial. The strictest degree of patient data privacy and confidentiality is required under this criterion. It guarantees adherence to regulations on healthcare data privacy and implements security measures to preserve patient information.

**20. *Real-Time Updates***: The term "real-time updates" describes the system's capacity to deliver instantaneous data updates for vital operations such as patient registration and billing queries. To avoid delays and enable efficient patient care and billing procedures, real-time updates are crucial in the healthcare industry.

21. ***Data Retrieval*:** It should be possible for users—including administrators and healthcare professionals—to obtain patient data by providing essential details such as the patient's name, contact number, or unique ID.

22. ***Audit Trail*:** To document all user interactions with patient and financial data, the system ought to keep thorough audit trails. These audit trails are necessary to monitor modifications, guarantee data integrity, and spot possible security lapses.

**3NF Relational Schema**

USERS\_TYPE (**USER\_TYPE\_CODE**, USER\_TYPE, USER\_DESC)

CLERK (**CLERK\_ID**, CLERK\_FIRST\_NAME, CLERK\_LAST\_NAME, CLERK\_PHONE\_NUM, **FK** **USER\_TYPE\_CODE**)

SPECIALITY (SPECIALITY\_ID, SPECIALITY\_TYPE, SPECIALITY\_DESC)

PHYSICIAN (**PHYSICIAN\_ID,** PHYSICIAN\_FIRST\_NAME, PHYSICIAN\_LAST\_NAME, PHYSICIAN\_TELEPHONE, **FK SPECIALITY\_ID**, **FK USER\_TYPE\_CODE**)

BED (**PK** **ROOM\_NUM, PK BED\_LETTER**, BED\_TYPE, **FK** **PATIENT\_NO**)

INSURANCE\_PROVIDER (**INSURANCE\_PROVIDER\_ID**, PROVIDER\_NAME, PROVIDER\_ADDRESS, PROVIDER\_TELEPHONE)

PATIENT (**PATIENT\_NO**, PATIENT\_FIRST\_NAME, PATIENT\_LAST\_NAME, PATIENT\_ADDRESS, PATIENT\_SEX, PATIENT\_HCN, **FK ROOM\_NUM**,PATIENT\_EXTENSION, DATE\_ADMITTED, FINACIAL\_STATUS, DISCHARGE\_DATE, **FK CLERK\_ID**, **FK PHYSICIAN\_ID**, PATIENT\_NOTES)

PATIENT\_INSURANCE\_PROVIDER (**PK FK INSURANCE\_PROVIDER\_ID**, **PK FK PATIENT\_NO)**

APPOINTMENT (**APPOINTMENT\_ID**, **FK PATIENT\_NO**, **FK CLERK\_ID**, APPOINTMENT\_DATE, APPOINTMENT\_NOTES)

PRESCRIPTION (**PRESCRIPTION\_ID**, **FK** **PATIENT\_NO**, **FK CLERK\_ID**, PRESCRIPTION\_NOTE, PRESCRIPTION\_DATE**, FK APPOINTMENT\_ID**)

LAB (**LAB\_ROOM\_NUM**, LAB\_VACANCY)

LAB\_TEST (**LAB\_TEST\_ID**, **FK PATIENT\_NO, FK PHYSICIAN\_ID**, **FK** **LAB\_ROOM\_NUM**, LAB\_TEST\_NAME, LAB\_TEST\_DATE, LAB\_TEST\_NOTES)

COST\_CENTRE (CONST\_CENTRE\_ID, COST\_CENTRE\_NAME, COST\_CENTRE\_DESC)

ITEMS (**ITEM\_CODE**, **FK** **COST\_CENTRE\_ID**, ITEM\_NAME, ITEM\_DESC, ITEM\_COST)

TRANSACTION (**TRANSACTION\_ID**, **FK** **PATIENT\_NO**, **FK INSURANCE\_ID**, **FK** **ITEM\_CODE,** TRANSACTION\_DATE, TRANSACTION\_TOTAL\_COST)

**ERD Diagram**



A blue and white document with black text

Description automatically generated

A blue and white document with black text

Description automatically generated with medium confidence A blue and white document with black text

Description automatically generated A close-up of a document

Description automatically generated A table with numbers and letters

Description automatically generated

**Queries**

**Database and Table Creation Script**

-- Drop the table if it exists

DROP TABLE IF EXISTS patient\_insurance\_provider;

DROP TABLE IF EXISTS transactions;

DROP TABLE IF EXISTS lab\_tests;

DROP TABLE IF EXISTS prescription;

DROP TABLE IF EXISTS appointments;

DROP TABLE IF EXISTS patient;

DROP TABLE IF EXISTS bed;

DROP TABLE IF EXISTS physician;

DROP TABLE IF EXISTS items;

DROP TABLE IF EXISTS lab;

DROP TABLE IF EXISTS clerk;

DROP TABLE IF EXISTS insurance\_provider;

DROP TABLE IF EXISTS cost\_centers;

DROP TABLE IF EXISTS speciality;

DROP TABLE IF EXISTS user\_type;

-- Drop the Database if exists

DROP DATABASE IF EXISTS lakeridge;

-- Creating the Database

CREATE DATABASE lakeridge;

USE lakeridge; -- Using the database

-- Creating the table user\_type

CREATE TABLE user\_type

(

user\_type\_code VARCHAR(10) PRIMARY KEY,

user\_type VARCHAR(50) NOT NULL,

user\_type\_desc VARCHAR(120)

);

-- Creating the Speciality table

CREATE TABLE speciality

(

speciality\_id INT PRIMARY KEY IDENTITY(1,1),

speciality\_type VARCHAR(200),

speciality\_desc VARCHAR(1024)

);

-- Creating the cost\_centers table

CREATE TABLE cost\_centers

(

cost\_center\_id INT PRIMARY KEY IDENTITY(1,1),

cost\_center\_name VARCHAR(100) NOT NULL,

cost\_center\_desc VARCHAR(1024)

);

-- Creating the insurance\_provider table

CREATE TABLE insurance\_provider

(

insurance\_provider\_id INT PRIMARY KEY IDENTITY(1,1),

provider\_name VARCHAR(100) NOT NULL,

provider\_address VARCHAR(400) NOT NULL,

provider\_telephone VARCHAR(20)

);

-- Creating the clerk table

CREATE TABLE clerk

(

clerk\_id INT PRIMARY KEY IDENTITY(1,1),

clerk\_first\_name VARCHAR(100),

clerk\_last\_name VARCHAR(100),

clerk\_phone VARCHAR(20) NOT NULL,

user\_type\_code VARCHAR(10),

FOREIGN KEY (user\_type\_code) REFERENCES user\_type(user\_type\_code)

);

-- Creating the lab table

CREATE TABLE lab

(

lab\_room\_num INT PRIMARY KEY,

lab\_vacancy VARCHAR(350)

);

-- Creating the items table

CREATE TABLE items

(

item\_code INT PRIMARY KEY,

item\_name VARCHAR(500) NOT NULL,

cost\_center\_id INT,

item\_price FLOAT NOT NULL,

item\_desc VARCHAR(500) NOT NULL,

FOREIGN KEY (cost\_center\_id) REFERENCES cost\_centers(cost\_center\_id)

);

-- Creating the physician table

CREATE TABLE physician

(

physician\_id INT PRIMARY KEY IDENTITY(1,1),

physician\_first\_name VARCHAR(100) NOT NULL,

physician\_last\_name VARCHAR(100) NOT NULL,

physician\_telephone VARCHAR(20) NOT NULL,

speciality\_id INT,

user\_type\_code VARCHAR(10),

FOREIGN KEY (speciality\_id) REFERENCES speciality(speciality\_id),

FOREIGN KEY (user\_type\_code) REFERENCES user\_type(user\_type\_code)

);

-- Creating the bed table

CREATE TABLE bed

(

room\_num INT,

bed\_letter VARCHAR(10),

bed\_type VARCHAR(10),

bed\_vacancy BIT,

PRIMARY KEY (room\_num, bed\_letter)

);

-- Creating the patient table

CREATE TABLE patient

(

patient\_no INT PRIMARY KEY IDENTITY(1,1),

patient\_first\_name VARCHAR(100) NOT NULL,

patient\_last\_name VARCHAR(100) NOT NULL,

patient\_address VARCHAR(200) NOT NULL,

patient\_sex VARCHAR(20) NOT NULL,

patient\_hcn INT NOT NULL,

physician\_id INT,

patient\_extension INT NOT NULL,

date\_admitted DATETIME NOT NULL,

discharge DATETIME,

room\_num INT,

bed\_letter varchar(10),

patient\_notes VARCHAR(1024),

clerk\_id INT,

FOREIGN KEY (physician\_id) REFERENCES physician(physician\_id),

FOREIGN KEY (room\_num, bed\_letter) REFERENCES bed(room\_num, bed\_letter),

FOREIGN KEY (clerk\_id) REFERENCES clerk(clerk\_id),

);

-- Creating the appointments table

CREATE TABLE appointments

(

appointment\_id INT PRIMARY KEY IDENTITY(1,1),

clerk\_id INT,

appointment\_date DATETIME NOT NULL,

appointment\_notes VARCHAR(1024),

patient\_no INT,

FOREIGN KEY (clerk\_id) REFERENCES clerk(clerk\_id),

FOREIGN KEY (patient\_no) REFERENCES patient(patient\_no)

);

-- Creating the prescription table

CREATE TABLE prescription

(

prescription\_id INT PRIMARY KEY IDENTITY(1,1),

patient\_no INT,

physician\_id INT,

prescription\_type VARCHAR(500) NOT NULL,

prescription\_notes VARCHAR(1024) NOT NULL,

prescription\_date DATETIME NOT NULL,

appointment\_id INT,

FOREIGN KEY (physician\_id) REFERENCES physician(physician\_id),

FOREIGN KEY (appointment\_id) REFERENCES appointments(appointment\_id)

);

-- Creating the lab\_tests table

CREATE TABLE lab\_tests

(

lab\_test\_id INT PRIMARY KEY IDENTITY(1,1),

patient\_no INT,

physician\_id INT,

lab\_room\_num INT,

test\_date DATETIME NOT NULL,

lab\_test\_notes VARCHAR(1024),

result VARCHAR(200) NOT NULL,

FOREIGN KEY (patient\_no) REFERENCES patient(patient\_no),

FOREIGN KEY (physician\_id) REFERENCES physician(physician\_id),

FOREIGN KEY (lab\_room\_num) REFERENCES lab(lab\_room\_num)

);

-- Creating the patient\_insurance\_provider table

CREATE TABLE patient\_insurance\_provider

(

insurance\_provider\_id INT,

patient\_no INT,

PRIMARY KEY (insurance\_provider\_id, patient\_no),

FOREIGN KEY (insurance\_provider\_id) REFERENCES insurance\_provider(insurance\_provider\_id),

FOREIGN KEY (patient\_no) REFERENCES patient(patient\_no)

);

-- Creating the transactions table

CREATE TABLE transactions

(

transaction\_id INT PRIMARY KEY IDENTITY(1,1),

patient\_no INT,

item\_code INT,

transaction\_date DATETIME NOT NULL,

transaction\_total\_cost FLOAT NOT NULL,

insurance\_provider\_id INT,

FOREIGN KEY (patient\_no) REFERENCES patient(patient\_no),

FOREIGN KEY (item\_code) REFERENCES items(item\_code),

FOREIGN KEY (insurance\_provider\_id) REFERENCES insurance\_provider(insurance\_provider\_id)

);

**Data Insert Script**

-- To make sure the lakeridge database is being used

USE lakeridge;

-- Inserts for user\_type table

INSERT INTO user\_type (user\_type\_code, user\_type, user\_type\_desc) VALUES

('CL', 'Clerk', 'Responsible for administrative tasks'),

('PH', 'Physician', 'Medical professional providing healthcare services'),

('AD', 'Admin', 'Administrative user with special privileges');

-- Inserts for speciality table

INSERT INTO speciality (speciality\_type, speciality\_desc) VALUES

('Cardiology', 'Specializing in heart-related issues'),

('Orthopedics', 'Specializing in musculoskeletal issues'),

('Dermatology', 'Specializing in skin conditions'),

('Neurology', 'Specializing in disorders of the nervous system'),

('Ophthalmology', 'Specializing in eye care');

-- Inserts for cost\_centers table

INSERT INTO cost\_centers(cost\_center\_name, cost\_center\_desc)

VALUES

('Emergency Department', 'Handles urgent medical cases 24/7'),

('Cardiology Department', 'Specialized in heart-related conditions'),

('Surgery Department', 'Performs various surgical procedures'),

('Laboratory', 'Conducts diagnostic tests and experiments'),

('Radiology Department', 'Imaging and diagnostic services'),

('Pediatrics Department', 'Specialized care for children'),

('Orthopedics Department', 'Deals with musculoskeletal issues'),

('Obstetrics and Gynecology', 'Womens health and pregnancy care'),

('Neurology Department', 'Focuses on nervous system disorders'),

('Ophthalmology Department', 'Specialized in eye care'),

('Dental Clinic', 'Provides dental services'),

('Psychiatry Department', 'Mental health and behavioral disorders'),

('Physical Therapy', 'Rehabilitation and movement disorders'),

('Nutrition Services', 'Offers dietary and nutrition advice'),

('Pharmacy', 'Dispenses medications and pharmaceuticals');

-- Inserts for insurance\_provider table

INSERT INTO insurance\_provider (provider\_name, provider\_address, provider\_telephone) VALUES

('Maple Insurance', '123 Main Street, Toronto, Canada', '1234567890'),

('GreatNorth Assurance', '456 Oak Avenue, Toronto, Canada', '9876543210'),

('EhShield Healthcare', '789 Pine Road, Toronto, Canada', '5551234567'),

('TrueNorth Health Plans', '321 Cedar Lane, Toronto, Canada', '8889876543'),

('Polaris Coverage', '654 Elm Street, Toronto, Canada', '2224567890');

-- Inserts for clerk table

INSERT INTO clerk (clerk\_first\_name, clerk\_last\_name, clerk\_phone, user\_type\_code) VALUES

('John', 'Smith', '1234567890', 'CL'),

('Emily', 'Jones', '9876543210', 'CL'),

('Daniel', 'Wong', '5551234567', 'CL'),

('Jennifer', 'Chen', '8889876543', 'CL'),

('Michael', 'Taylor', '2224567890', 'CL');

-- Inserts for items table

INSERT INTO items

VALUES

(101, 'Stethoscope', 1, 50.00, 'Medical instrument for listening to internal sounds'),

(102, 'Blood Pressure Monitor', 2, 120.00, 'Device for measuring blood pressure'),

(103, 'Surgical Gloves', 3, 5.00, 'Disposable gloves for surgical procedures'),

(104, 'X-Ray Machine', 4, 150000.00, 'Diagnostic imaging equipment'),

(105, 'MRI Scanner', 5, 300000.00, 'Magnetic Resonance Imaging device'),

(106, 'Pediatric Ventilator', 6, 1500.00, 'Assists breathing in pediatric patients'),

(107, 'Orthopedic Implants', 7, 2000.00, 'Implants for orthopedic surgeries'),

(108, 'Prenatal Vitamins', 8, 15.00, 'Supplements for pregnant women'),

(109, 'Neurology Consultation', 9, 200.00, 'Consultation with a neurologist'),

(110, 'Eyeglasses', 10, 100.00, 'Prescription eyeglasses'),

(111, 'Dental Cleaning Kit', 11, 20.00, 'Kit for dental hygiene'),

(112, 'Antidepressant Medication', 12, 50.00, 'Medication for mental health'),

(113, 'Physical Therapy Session', 13, 75.00, 'Individual physical therapy session'),

(114, 'Dietary Supplements', 14, 25.00, 'Supplements for nutritional needs'),

(115, 'Prescription Medications', 15, 10.00, 'Various prescription medications');

-- Inserts for lab table

INSERT INTO lab (lab\_room\_num, lab\_vacancy) VALUES

(101, 'Vacant'),

(102, 'Occupied'),

(103, 'Vacant');

-- Inserts for physician table

INSERT INTO physician (physician\_first\_name, physician\_last\_name, physician\_telephone, speciality\_id, user\_type\_code) VALUES

('Dr. Manu', 'Sugunakumar', '1234567890', 1, 'PH'),

('Dr. Himanshu', 'Blank', '9876543210', 2, 'PH'),

('Dr. Neilavan', 'Vijayakanthan', '5551234567', 3, 'PH'),

('Dr. Addan', 'Zahra', '8889876543', 4, 'PH'),

('Dr. Jennifer', 'Short', '2224567890', 5, 'PH');

-- Inserts for bed table

INSERT INTO bed (room\_num, bed\_letter, bed\_type, bed\_vacancy)

VALUES

(251, 'A', 'W4', 1),

(251, 'B', 'W4', 1),

(251, 'C', 'W4', 1),

(251, 'D', 'W4', 1),

(252, 'A', 'W4', 1),

(252, 'B', 'W4', 0),

(252, 'C', 'W4', 0),

(252, 'D', 'W4', 0),

(253, 'A', 'W4', 1),

(253, 'B', 'W4', 1),

(253, 'C', 'W4', 1),

(253, 'D', 'W4', 0),

(254, 'A', 'W4', 0),

(254, 'B', 'W4', 0),

(254, 'C', 'W4', 1),

(254, 'D', 'W4', 0),

(255, 'A', 'SP', 0),

(255, 'B', 'SP', 0),

(256, 'A', 'SP', 1),

(256, 'B', 'SP', 1),

(257, 'A', 'SP', 1),

(257, 'B', 'SP', 1),

(258, 'A', 'SP', 1),

(258, 'B', 'SP', 1),

(259, 'A', 'SP', 1),

(259, 'B', 'SP', 1),

(261, 'A', 'W3', 0),

(261, 'B', 'W3', 0),

(261, 'C', 'W3', 0),

(262, 'A', 'W3', 0),

(262, 'B', 'W3', 1),

(262, 'C', 'W3', 0),

(263, 'A', 'W3', 0),

(263, 'B', 'W3', 1),

(263, 'C', 'W3', 1),

(264, 'A', 'P', 0),

(265, 'A', 'P', 0),

(266, 'A', 'P', 0),

(267, 'A', 'P', 0),

(268, 'A', 'P', 0),

(269, 'A', 'P', 0),

(270, 'A', 'P', 0),

(271, 'A', 'IC', 1),

(271, 'B', 'IC', 1),

(271, 'C', 'IC', 1),

(271, 'D', 'IC', 1),

(272, 'A', 'IC', 1),

(272, 'B', 'IC', 1),

(272, 'C', 'IC', 0),

(272, 'D', 'IC', 0),

(273, 'A', 'IC', 0),

(273, 'B', 'IC', 0),

(273, 'C', 'IC', 0),

(273, 'D', 'IC', 0);

-- Inserts for patient table

INSERT INTO patient (patient\_first\_name, patient\_last\_name, patient\_address, patient\_sex, patient\_hcn, physician\_id, patient\_extension, date\_admitted, discharge, room\_num, bed\_letter, patient\_notes, clerk\_id)

VALUES

('John', 'Doe', '123 Main St', 'Male', 123456789, 1, 101, '2023-01-01', '2023-01-05', 251, 'A', 'Broken thumb.', 1),

('Jane', 'Smith', '456 Oak St', 'Female', 987654321, 2, 102, '2023-02-05', '2023-02-10', 251, 'B', 'Cough, body overheating.', 2),

('Michael', 'Johnson', '789 Elm St', 'Male', 555123456, 3, 103, '2023-03-10', '2023-03-15', 251, 'C', 'Broken leg.', 3),

('Emily', 'Williams', '101 Pine St', 'Female', 444567890, 4, 104, '2023-04-15', '2023-04-20', 251, 'D', 'Stitches Removal', 4),

('Daniel', 'Brown', '202 Cedar St', 'Male', 333678901, 5, 105, '2023-05-20', '2023-05-25', 252, 'A', 'Tongue stuck on pole.', 5),

('Alice', 'Johnson', '111 Pine St', 'Female', 111223344, 1, 111, '2023-06-01', NULL, 252, 'B', 'Recovering from car accident.', 1),

('Bob', 'Smith', '222 Oak St', 'Male', 222334455, 2, 222, '2023-06-02', NULL, 252, 'C', 'Recovering from car accident.', 2),

('Charlie', 'Williams', '333 Elm St', 'Male', 333445566, 3, 333, '2023-06-03', NULL, 252, 'D', 'Left foot sprained.', 3),

('David', 'Brown', '444 Cedar St', 'Male', 444556677, 4, 444, '2023-06-04', '2023-06-05', 253, 'A', 'Butt hurts.', 4),

('Eva', 'Taylor', '555 Maple St', 'Female', 555667788, 5, 555, '2023-06-05', '2023-06-05', 253, 'B', 'Head Hurts', 5),

('Frank', 'Miller', '666 Pine St', 'Male', 666778899, 1, 666, '2023-06-06', '2023-06-06', 253, 'C', 'Stuffy nose.', 1),

('Grace', 'Davis', '777 Oak St', 'Female', 777889900, 2, 777, '2023-06-07', NULL, 253, 'D', 'Broke his middle finger.', 2),

('Henry', 'White', '888 Elm St', 'Male', 888990011, 3, 888, '2023-06-08', NULL, 254, 'A', 'Broke his neck.', 3),

('Ivy', 'Harris', '999 Cedar St', 'Female', 999001122, 4, 999, '2023-06-09', NULL, 254, 'B', 'Cat scratched his face.', 4),

('Jack', 'Jones', '101 Maple St', 'Male', 101112233, 5, 101, '2023-06-10', '2023-06-10', 254, 'C', 'Dog bite on right leg.', 5),

('Kate', 'Anderson', '121 Pine St', 'Female', 121314151, 1, 121, '2023-06-11', NULL, 254, 'D', 'Broke his ribcage from fighting.', 1),

('Leo', 'Moore', '141 Oak St', 'Male', 141516171, 2, 141, '2023-06-12', NULL, 261, 'A', 'Throat infection.', 2),

('Mia', 'Thomas', '161 Elm St', 'Female', 161718192, 3, 161, '2023-06-13', NULL, 261, 'B', 'STD', 3),

('Noah', 'Lee', '181 Cedar St', 'Male', 181920212, 4, 181, '2023-06-14', NULL, 261, 'C', 'STD', 4),

('Olivia', 'Roberts', '201 Maple St', 'Female', 201212223, 5, 201, '2023-06-15', NULL, 262, 'A', 'Goneria', 5),

('Paul', 'Baker', '221 Pine St', 'Male', 221222233, 1, 221, '2023-06-16', '2023-06-16', 262, 'B', 'Head stuck in washing machine.', 1),

('Quinn', 'Clark', '241 Oak St', 'Female', 241252263, 2, 241, '2023-06-17', NULL, 262, 'C', 'Left eye not working.', 2),

('Ryan', 'Hill', '261 Elm St', 'Male', 261272283, 3, 261, '2023-06-18', NULL, 263, 'A', 'Can not feel his face.', 3),

('Sofia', 'Ward', '281 Cedar St', 'Female', 281292303, 4, 281, '2023-06-19', '2023-06-19', 263, 'B', 'Burned his righ hand on stove.', 4),

('Tyler', 'Cooper', '301 Maple St', 'Male', 301312323, 5, 301, '2023-06-20', '2023-06-20', 263, 'C', 'Burned his left hand on stove.', 5),

('Uma', 'Reed', '321 Pine St', 'Female', 321322333, 1, 321, '2023-06-21', NULL, 264, 'A', 'Left torso is injured.', 1),

('Victor', 'Morgan', '341 Oak St', 'Male', 341352363, 2, 341, '2023-06-22', NULL, 265, 'A', 'Getting tests done.', 2),

('Wendy', 'Hayes', '361 Elm St', 'Female', 361372383, 3, 361, '2023-06-23', NULL, 266, 'A', 'Staying at the hospital.', 3),

('Xander', 'Fisher', '381 Cedar St', 'Male', 381392403, 4, 381, '2023-06-24', NULL, 267, 'A', 'Rich guy trying to fake his death.', 4),

('Yara', 'Stewart', '401 Maple St', 'Female', 401412423, 5, 401, '2023-06-25', NULL, 268, 'A', 'REDACTED', 5);

-- Inserts for appointments table

INSERT INTO appointments (clerk\_id, appointment\_date, appointment\_notes, patient\_no)

VALUES

(1, '2023-01-05 10:00:00', 'Came in with a broken thumb.', 1),

(2, '2023-02-10 14:30:00', 'Came in with a cough.', 2),

(3, '2023-03-15 11:45:00', 'Came in with broken leg. Fixed.', 3),

(4, '2023-04-20 09:30:00', 'Removed stitches from previous incidient.', 4),

(5, '2023-05-25 13:15:00', 'Came in with their tongue stuck on a pole.', 5),

(1, '2023-06-01 11:00:00', 'Follow-up appointment for recovering from a car accident.', 6),

(2, '2023-06-02 14:30:00', 'Follow-up appointment for recovering from a car accident.', 7),

(3, '2023-06-03 10:45:00', 'Follow-up appointment for a left foot sprain.', 8),

(4, '2023-06-04 09:30:00', 'Follow-up appointment for a butt injury.', 9),

(5, '2023-06-05 13:45:00', 'Follow-up appointment for a head injury.', 10),

(1, '2023-06-06 15:00:00', 'Follow-up appointment for a stuffy nose.', 11),

(2, '2023-06-07 12:30:00', 'Follow-up appointment for a broken middle finger.', 12),

(3, '2023-06-08 14:45:00', 'Follow-up appointment for a broken neck.', 13),

(4, '2023-06-09 11:15:00', 'Follow-up appointment for a cat scratch on the face.', 14),

(5, '2023-06-10 13:30:00', 'Follow-up appointment for a dog bite on the right leg.', 15),

(1, '2023-06-11 10:00:00', 'Follow-up appointment for a broken ribcage.', 16),

(2, '2023-06-12 14:15:00', 'Follow-up appointment for a throat infection.', 17),

(3, '2023-06-13 13:45:00', 'Follow-up appointment for STD treatment.', 18),

(4, '2023-06-14 11:30:00', 'Follow-up appointment for STD treatment.', 19),

(5, '2023-06-15 16:00:00', 'Follow-up appointment for STD treatment.', 20),

(1, '2023-06-16 09:45:00', 'Follow-up appointment for a head stuck in the washing machine.', 21),

(2, '2023-06-17 12:00:00', 'Follow-up appointment for a left eye not working.', 22),

(3, '2023-06-18 15:30:00', 'Follow-up appointment for not feeling the face.', 23),

(4, '2023-06-19 14:15:00', 'Follow-up appointment for a burned right hand.', 24),

(5, '2023-06-20 11:00:00', 'Follow-up appointment for a burned left hand.', 25),

(1, '2023-06-21 12:45:00', 'Follow-up appointment for a left torso injury.', 26),

(2, '2023-06-22 14:30:00', 'Getting tests done.', 27),

(3, '2023-06-23 16:00:00', 'Staying at the hospital.', 28),

(4, '2023-06-24 11:30:00', 'Rich guy trying to fake his death.', 29),

(5, '2023-06-25 13:15:00', 'REDACTED', 30);

-- Inserts for prescription table

INSERT INTO prescription (patient\_no, physician\_id, prescription\_type, prescription\_notes, prescription\_date, appointment\_id)

VALUES

(1, 1, 'Pain Medication', 'Pain medication', '2023-01-05 10:30:00', 1),

(2, 2, 'Cough Syrup', 'Cough Syrup', '2023-02-10 15:00:00', 2),

(3, 3, 'Pain medication & healing cream', 'Pain medication & healling cream.', '2023-03-15 12:15:00', 3),

(4, 4, 'Healing Cream', 'Healing cream', '2023-04-20 10:00:00', 4),

(5, 5, 'Heat Pills', 'Heat pills', '2023-05-25 13:45:00', 5);

-- Inserts for lab\_tests table

INSERT INTO lab\_tests (patient\_no, physician\_id, lab\_room\_num, test\_date, lab\_test\_notes, result)

VALUES

(1, 1, 101, '2023-01-07 11:00:00', 'Tests came back normal', 'Normal'),

(2, 2, 102, '2023-02-12 14:45:00', 'Tests showed just common cold.', 'Normal'),

(3, 3, 103, '2023-03-17 12:30:00', 'Showed infection on scar.', 'Abnormal'),

(4, 4, 102, '2023-04-22 09:45:00', 'Leg shows no infection.', 'Normal'),

(5, 5, 101, '2023-05-27 13:30:00', 'Tongue shows bruising and infection', 'Abnormal');

-- Inserts for transactions table

INSERT INTO transactions (patient\_no, item\_code, transaction\_date, transaction\_total\_cost, insurance\_provider\_id)

VALUES

(1, 101, '2023-01-10 13:30:00', 50.00, 1),

(2, 102, '2023-02-15 16:15:00', 20.99, 2),

(3, 103, '2023-03-20 14:00:00', 10.50, 3),

(4, 104, '2023-04-25 11:30:00', 40.75, 4),

(5, 105, '2023-05-30 15:45:00', 75.00, 5);

INSERT INTO patient\_insurance\_provider (insurance\_provider\_id, patient\_no)

VALUES

(1, 1),

(2, 2),

(3, 3),

(4, 4),

(5, 5);

**Queries for room utilization**

-- Discharge Date

DECLARE @dischargeDate DATE = '2023-12-06';

SELECT COUNT(\*) AS discharging\_beds\_today

FROM bed b

JOIN patient p ON b.room\_num = p.room\_num AND b.bed\_letter = p.bed\_letter

WHERE CONVERT(DATE, p.discharge) = @dischargeDate;

**-- Occupancy by room type**

SELECT cost\_center\_name, COUNT(\*) AS empty\_rooms\_count

FROM bed b

RIGHT JOIN cost\_centers c ON b.room\_num = c.cost\_center\_id

WHERE b.bed\_vacancy = 1 OR b.bed\_vacancy IS NULL

GROUP BY cost\_center\_name;

**-- Bed Vacancies**

SELECT COUNT(\*) AS occupied\_beds

FROM bed

WHERE bed\_vacancy = 0;

**-- Occupancy by rooms**

SELECT room\_num, COUNT(\*) AS occupancy\_count

FROM bed

WHERE bed\_vacancy = 0

GROUP BY room\_num;

**-- Empty Rooms by Room Type**

SELECT c.cost\_center\_name, COUNT(\*) AS empty\_rooms\_count

FROM cost\_centers c

LEFT JOIN bed b ON c.cost\_center\_id = b.room\_num

AND b.bed\_vacancy = 1

GROUP BY c.cost\_center\_name;

**Queries for Physician-Patient Details**

**-- Query to view all patients for that physician**

-- Change physician\_id to the physician who wishes to see their patients

DECLARE @physician\_id INT;

SET @physician\_id = 1;

-- View all patients of the physician currently in the hospital discharge is null because we are looking for patients currently in the hospital

SELECT \*

FROM patient

WHERE physician\_id = @physician\_id

AND discharge IS NULL;

**-- Query to view total number of appointments per patient**

-- Change the patient number based on which patient you want to see

DECLARE @patient\_no INT;

SET @patient\_no = 1;

SELECT

p.patient\_no,

p.patient\_first\_name,

p.patient\_last\_name,

COUNT(a.appointment\_id) AS total\_appointments

FROM

patient p

LEFT JOIN

appointments a ON p.patient\_no = a.patient\_no

WHERE

p.patient\_no = @patient\_no

GROUP BY

p.patient\_no, p.patient\_first\_name, p.patient\_last\_name;

-- Query to view specific patient data

-- Type the patient no you wish to see patient information for

DECLARE @patient\_no INT = 1;

SELECT

patient\_no,

patient\_first\_name,

patient\_last\_name,

patient\_address,

patient\_sex,

patient\_hcn,

physician\_id,

patient\_extension,

date\_admitted,

discharge,

room\_num,

bed\_letter,

patient\_notes,

clerk\_id

FROM

patient

WHERE

patient\_no = @patient\_no;

**-- Query to view past and future appointments for a patient**

-- Set to the patient id you wish to see their appointment for

DECLARE @patient\_no INT = 1;

-- Retrieve previous appointments

SELECT

appointment\_id,

appointment\_date,

appointment\_notes

FROM

appointments

WHERE

patient\_no = @patient\_no

AND appointment\_date < GETDATE()

ORDER BY

appointment\_date DESC;

-- Retrieve upcoming appointments

SELECT

appointment\_id,

appointment\_date,

appointment\_notes

FROM

appointments

WHERE

patient\_no = @patient\_no

AND appointment\_date >= GETDATE()

ORDER BY

appointment\_date ASC;

**-- Query to assign new appointment to patients**

-- Replace id's with the id of the clerk who is assignning the appointment and the patient no

DECLARE @clerk\_id INT = 1; -- Clerk assigning the appointment

DECLARE @patient\_no INT = 6; -- Patient for whom the appointment is assigned

DECLARE @appointment\_date DATETIME = '2023-07-01 14:00:00'; -- Desired appointment date and time

DECLARE @appointment\_notes VARCHAR(1024) = 'Follow-up appointment';

-- Insert the new appointment

INSERT INTO appointments (clerk\_id, appointment\_date, appointment\_notes, patient\_no)

VALUES (@clerk\_id, @appointment\_date, @appointment\_notes, @patient\_no);

-- Optional: Display the newly assigned appointment

SELECT \* FROM appointments

WHERE appointment\_id = SCOPE\_IDENTITY();

**-- Query to change the note of a patient**

-- Replace the placeholders with actual values

DECLARE @patient\_no INT = 6; -- Patient number to update

DECLARE @new\_notes VARCHAR(1024) = 'This patient has a new disease.';

-- Update the patient notes

UPDATE patient

SET patient\_notes = @new\_notes

WHERE patient\_no = @patient\_no;

-- Display new updated patient

SELECT \*

FROM patient

WHERE patient\_no = @patient\_no;

**Query Used in Demo**

SELECT

patient.room\_num + patient.bed\_letter AS Location,

bed.bed\_type as Type,

patient.patient\_no AS [Patient-No],

patient.patient\_first\_name + ', ' + patient.patient\_last\_name AS [Patient-Name],

CONVERT(VARCHAR(10), patient.date\_admitted, 101) AS [Date-Admitted]

FROM

patient

JOIN bed ON patient.room\_num = bed.room\_num AND patient.bed\_letter = bed.bed\_letter

;

**Picture of Application**

A screenshot of a medical report

Description automatically generated

**Conclusion**

To sum up, the difficulties that Lake Ridge Community Hospital (LRCH) is facing highlight how urgently an improved and integrated information system is needed. Our methodical approach, which is based on thorough understanding of the business challenge and conversations with stakeholders, has cleared the path for a mission-driven strategy that is centered on providing a reliable database solution.

Our objective is very clear: to give LRCH a cutting-edge information system that not only solves the existing shortcomings but also supports the hospital's dedication to excellence, staff empowerment, and better patient care. Our objective is to improve decision-making, encourage process efficiency, and guarantee data accuracy, usability, and security through patient management, physician management, reporting, revenue management, and other crucial capabilities.

Beyond the current initiative, we are committed to finding a reliable, scalable, and long-lasting solution. By following industry best practices for database creation and encouraging openness, cooperation, and creativity, we want to make a major contribution to LRCH's success. The intended database serves as a basis for future growth and quality in healthcare service in addition to being a useful tool for the here and now.

Our dedication to enabling efficient operations, bolstering research endeavors, and eventually improving patient outcomes is unwavering as we proceed. In addition to showcasing a database solution, the final presentation to LRCH will signify a collaboration toward realizing the hospital's goal of offering the community access to top-notch medical care. We look forward to having a long-lasting and beneficial influence on Lake Ridge Community Hospital and its mission in the healthcare industry with openness, transparency, and a commitment to continual progress.