



Hospital Management System

Prepared By

**Reem Raft sadik Mohamed
Manar Awad Mohamed Mohamed
Yousef Hany Elsayed Ahmed
Omar Tharwat Ibrahiem Abdulla
Ahmed Mohamed Said Ahmed Salem
Ibrahem Mahmoud Mohamed Elsayed**

**Supervised By
Eng. Eman Gomaa**

**INFORMATION TECHNOLOGY INSTITUTE 4-MONTHS
GRADUATION PROJECT**

With sincere thanks to Dr. Rami Abou-nagi and Dr. Ayman Lotfy for their support and leadership.

Table of Contents

1. Project Overview	3
2. Database Design	4
3. Stored Procedures	22
4. Data Warehouse Design	25
5. ETL Process in DWH	29
6. SSRS Reports	32
7. Power BI Dashboards	39
8. Website	48

1. Project Overview

The Hospital Management System is a comprehensive digital solution designed to streamline and manage hospital operations through a centralized platform. The system is built around a **relational database**, a **web-based interface**, an **AI-powered disease diagnosis module**, and **real-time dashboards using Power BI**.

A key component of the system is the **web interface**, which acts as a **database engine** that facilitates all essential data operations. Through this interface, authorized users can **insert**, **update**, and **delete** records in the database securely and efficiently. It provides an intuitive and user-friendly way for hospital staff to manage critical information such as patient details, appointments, medical records, billing transactions, room allocations, and staff schedules without directly interacting with the underlying database.

Additionally, the system integrates AI capabilities to assist in **early disease diagnosis** and employs Power BI for **real-time analytics and dashboards**, enabling data-driven decision-making and improving overall operational efficiency and medical accuracy.

<i>Role</i>	<i>Can Manage</i>
<i>Admin</i>	Everything
<i>Doctor</i>	Diagnoses, Prescriptions, Records
<i>Nurse</i>	Assignments, Vitals, Basic Records
<i>HR Manager</i>	Employees, Leave, Shifts
<i>Finance Officer</i>	Bills, Payments, Insurance
<i>Dispatcher</i>	Ambulances, Requests, Logs
<i>Lab Technician</i>	Test Results
<i>Radiology Tech</i>	Radiology Reports
<i>Patient</i>	Show his record and his notifications

2. Database Design

The Management system is backed by a relational SQL database with interrelated tables. Below is an overview and explanation of the key tables used in the database design.

Users

Purpose:

Represents system users who can log in. These users can be employees, patients, or roles defined in the system (like admin, doctor, etc.).

Key Columns:

- `user_id`: Unique identifier for each user (auto-incremented).
- `username`: Unique login name.
- `password_hash`: Hashed password for security.
- `employee_id`: Links to an employee (for staff users).
- `patient_id`: Links to a patient (for patient users).
- `role_id`: Links to the role defining permissions.

Roles

Purpose:

Defines different user roles within the system, like admin, doctor, patient, etc.

Key Columns:

- `role_id`: Primary key.
- `role_code`: Code name like 'admin', 'doctor'.
- `role_name_en`: Role's full name in English.

Permissions

Purpose:

Lists all available system permissions, such as creating prescriptions or viewing patient data.

Key Columns:

- permission_id: Primary key.
- permission_code: Permission identifier like 'create_prescription'.
- permission_description: Description of the permission

Notifications

Purpose:

Stores messages/notifications for users, with read/unread status.

Key Columns:

- notification_id: Primary key.
- user_id: FK to Users.
- message: Notification content.
- is_read: Boolean flag if notification was read.
- created_at: Timestamp of creation.

AuditLogs

Purpose:

Logs user actions for auditing (e.g., data changes).

Key Columns:

- log_id: Primary key, auto-incremented.
- user_id: FK to Users who performed the action.
- action: Action type (e.g., 'INSERT', 'UPDATE').
- entity_type: Type of entity affected (e.g., 'Patient').
- entity_id: ID of the affected entity.
- timestamp: Action timestamp.
- details: Additional info.

Employees

Purpose:

Stores employee details in the hospital (doctors, nurses, admin staff).

Key Columns:

employee_id: PK, auto-incremented.

Personal info: full_name, national_id, gender, birth_date, phone_number, email, address.

Work info: hire_date, job_title, department_id, status (active, on leave, terminated).

Tracking: created_at, updated_at.

Leave_Requests

Purpose:

Manages employee leave applications and their status.

Key Columns:

leave_id: PK, auto-incremented.

employee_id: FK to Employees.

leave_type: e.g., annual, sick.

start_date, end_date: Leave period.

reason: Optional explanation.

status: Pending, Approved, Rejected.

requested_at: Timestamp of request.

Shifts

Purpose:

Tracks employee shift schedules.

Key Columns:

- shift_id: PK, auto-incremented.
- employee_id: FK to Employees.
- shift_day: Day name, e.g., Monday.
- start_time, end_time: Shift hours.
- location: Work location.
- created_at, updated_at: Tracking timestamps.

Departments

Purpose:

Hospital departments like Cardiology, Radiology, etc.

Key Columns:

- department_id: PK, auto-incremented.
- department_name: Unique department name.
- description: Department description.
- managr_id: FK to Employees for department manager.
- phone_number, location: Contact info.
- created_at, updated_at: Timestamps.

Radiology_Types

Purpose:

Types of radiology exams offered.

Key Columns:

- radiology_type_id: PK.
- type_name: e.g., X-Ray, MRI.
- description.
- department_id: FK to Departments

Radiology_Reports

Purpose:

Stores reports of radiology exams.

Key Columns:

- report_id: PK, auto-incremented.
- visit_id: FK to Visits.
- radiology_type_id: FK to Radiology_Types.
- image_url: Image location.
- report_text: Text report.
- report_date: Date of report.

Lab_Tests

Purpose:

Types of lab tests available.

Key Columns:

- lab_test_id: PK.
- test_name, description.
- department_id: FK to Departments.

Test_Result_Details

Purpose:

Records detailed lab test results per patient visit.

Key Columns:

- detail_id: PK, auto-incremented.
- visit_id: FK to Visits.
- lab_test_id: FK to Lab_Tests.
- attribute_name: Measured attribute, e.g., glucose.
- attribute_value: Test result value.
- normal_range: Reference range.
- result_date: Date test was recorded.

Bills

Purpose:

Billing information per patient visit.

Key Columns:

- bill_id: PK.
- patient_id: FK to Patients.
- visit_id: FK to Visits.
- total_amount.
- insurance_id: Optional FK to Insurance.
- created_by: Employee who created the bill.
- created_at.

Bill_Items

Purpose:

Details of each service or item charged on a bill.

Key Columns:

- bill_item_id: PK.
- bill_id: FK to Bills.
- service_description.
- amount.
- department_id: FK to Departments.
- created_at.

Payments

Purpose:

Payment transactions related to bills.

Key Columns:

- payment_id: PK.
- bill_id: FK to Bills.
- method_id: FK to Payment_Methods (cash, insurance, credit).
- amount.
- payment_date.
- received_by: Employee who received payment.

Insurance

Purpose:

Insurance companies and policies.

Key Columns:

- insurance_id: PK.
- company_name, policy_number, coverage_type (full/partial), contact_info.

Payment_Methods

Purpose:

Payment methods supported.

Key Columns:

- method_id: PK.
- method_name: e.g., Cash, Credit Card, Insurance.

Ambulances ok

Purpose:

Details about ambulances in the hospital fleet.

Key Columns:

- ambulance_id: PK, auto-incremented.
- vehicle_number, model.
- driver_id: FK to Employees.
- is_available: Availability status.
- notes.

Ambulance_Requests ok

Purpose:

Requests for ambulance services.

Key Columns:

- request_id: PK, auto-incremented.
- patient_id: FK to Patients.
- ambulance_id: FK to Ambulances (may be null if not assigned).
- request_time.
- status: Pending, Accepted, etc.
- destination.

Ambulance_Logs

Purpose:

Logs trips made by ambulances. Key Columns:

log_id: PK, autoincremented.

ambulance_id, driver_id: FK references.

log_time.

trip_summary.

Doctors

Purpose:

Doctors in the hospital, tied to employee records.

Key Columns:

-doctor_id: PK.

-employee_id: FK to Employees.

-license_number.

-years_of_experience.

-education.

-specialization_id: FK to Specializations.

Specializations

Purpose:

Medical specialties.

Key Columns:

-specialization_id: PK, auto-incremented.

-name: Specialty name.

-created_at

-updated_at.

Appointments

Purpose:

Records scheduled patient appointments with doctors.

Key Columns:

- appointment_id: PK, auto-incremented.
- patient_id: FK to Patients.
- doctor_id: FK to Doctors.
- appointment_time.
- status: Scheduled, Completed, Cancelled.
- notes.
- created_at.
-

Diagnoses

Purpose:

Records of diagnoses made during visits.

Key Columns:

- diagnosis_id: PK, auto-incremented.
- visit_id: FK to Visits.
- doctor_id: FK to Doctors.
- diagnosis_text.
- diagnosis_date.

Prescriptions

Purpose:

Records prescriptions issued during visits.

Key Columns:

- prescription_id: PK, auto-incremented.
- visit_id: FK to Visits.
- doctor_id: FK to Doctors.
- issue_date.
- notes.

PrescriptionItems

Purpose:

Details of medicines prescribed in each prescription.

Key Columns:

- item_id: PK, auto-incremented.
- prescription_id: FK to Prescriptions.
- medicine_name, dosage, duration, instructions.

Nurses

Purpose:

Nurses in the hospital linked to employees.

Key Columns:

- nurse_id: PK.
- employee_id: FK to Employees.
- license_number.
- experience_years.
- specialization.

Nurse_Room_Assignments

Purpose:

Assigns nurses to rooms, optionally linked to shifts.

Key Columns:

- assignment_id: PK, auto-incremented.
- nurse_id: FK to Nurses.
- room_id: FK to Rooms.
- assigned_at, shift_id (FK), notes.

Patients ok

Purpose:

Patient demographic and contact info.

Key Columns:

- patient_id: PK, auto-incremented.
- first_name, last_name, date_of_birth, gender.
- phone_number, email, address.
- Emergency contact info.
- insurance_id: FK to Insurance.
- created_at, updated_at.

Allergies ok

Purpose:

Patient allergies with severity.

Key Columns:

- allergy_id: PK, auto-incremented.
- patient_id: FK to Patients.
- allergy_name.
- severity: Mild, Moderate, Severe.

Beds ok

Purpose:

Beds within rooms.

Key Columns:

- bed_id: PK, auto-incremented.
- room_id: FK to Rooms.
- bed_number.
- status: Available, Occupied, Maintenance.

Visits

Purpose:

Records patient visits to the hospital.

Key Columns:

- visit_id: PK, auto-incremented.
- patient_id: FK to Patients.
- visit_date.
- visit_type: Emergency, Follow-up, etc.
- reason: Chief complaint.
- created_at, updated_at.

Visit_Rooms

Purpose:

Records patient visits in rooms with start and end times.

Key Columns:

- visit_room_id: PK, auto-incremented.
- visit_id: FK to Visits.
- room_id: FK to Rooms.
- start_time, end_time.

Rooms

Purpose:

Hospital rooms with info about location, capacity, type, and status.

Key Columns:

- room_id: PK, auto-incremented.
- room_number, floor, capacity, room_type (ICU, Standard, etc.).
- department_id: FK (optional).
- status: Available, Occupied, Maintenance.
- created_at, updated_at.

Chronic_Conditions

Purpose:

Chronic diseases/conditions recorded per patient.

Key Columns:

- condition_id: PK, auto-incremented.
- patient_id: FK.
- condition_name.
- diagnosis_date.
- created_at, updated_at.

Medical_Records ok

Purpose:

Detailed medical record summaries per patient visit.

Key Columns:

- record_id: PK, auto-incremented.
- visit_id: FK to Visits (unique, one record per visit).
- Vital signs (blood pressure, heart rate, etc.).
- Summaries: diagnosis, tests, radiology, chronic conditions, allergies.
- General health status (Good, Fair, Critical).
- Treatment plan, doctor notes, lifestyle notes.
- Follow-up flags and dates (with constraint ensuring consistency).
- created_by: FK to employee who created record.
- created_at, updated_at.

Patient_Insurance

Purpose:

Stores insurance policy details specific to each patient.

Key Columns:

- patient_insurance_id: PK, auto-incremented unique ID for each patient insurance record.
- patient_id: FK to Patients.
- insurance_id: FK to Insurance.
- policy_number: Insurance policy number for the patient.
- coverage_type: Type of coverage for this patient (e.g., Full, Partial).
- is_active: Indicates if the insurance is currently active (default = 1).
- start_date: Coverage start date.
- end_date: Coverage end date.

Visit_Vitals

Purpose:

Captures vital signs recorded during a patient's visit. These measurements help monitor the patient's health status and are typically recorded by nursing staff or medical assistants.

Key Columns:

- vital_id: Unique identifier for each vital record (auto-incremented).
- visit_id: Foreign key linking to the visit during which the vitals were recorded.
- blood_type: Patient's blood group (e.g., A+, O-).
- body_temperature: Body temperature in Celsius.
- blood_pressure_systolic: Systolic value of blood pressure.
- blood_pressure_diastolic: Diastolic value of blood pressure.
- heart_rate: Number of heartbeats per minute.
- respiratory_rate: Number of breaths per minute.
- blood_oxygen_level: Blood oxygen saturation percentage.
- recorded_by: User or employee who entered the vitals.
- recorded_at: Timestamp indicating when the vitals were recorded.

Services

Purpose:

Stores details of all medical and non-medical services provided by the hospital, such as lab tests, radiology scans, procedures, or consultations. Each service includes pricing and categorization information.

Key Columns:

- service_id: Unique identifier for each service (auto-incremented).
- service_name: Name of the service (e.g., Blood Test, MRI, Consultation).
- description: Additional details describing the service.
- standard_price: Default cost of the service.
- service_category: Category of the service (e.g., Lab, Radiology, Consultation).
- department_id: Foreign key linking the service to the department responsible for it.
- is_active: Indicates whether the service is currently available (1 = active, 0 = inactive).
- created_at: Timestamp of when the service was added.

Expenses

Purpose:

Stores records of all financial expenditures made by the hospital, including operational, administrative, and medical costs. Each expense is linked to a specific category for better financial tracking and reporting.

Key Columns:

- expense_id: Unique identifier for each expense (auto-incremented).
- category_id: Foreign key linking to Expense_Categories to classify the type of expense.
- amount: The monetary value of the expense.
- description: Text description providing context or details about the expense.
- expense_date: The actual date the expense occurred.
- recorded_by: References the user or employee who recorded the expense.
- created_at: Timestamp of when the expense entry was added.
- updated_at: Timestamp of the last update made to the record.

Expense_Categories

Purpose:

Defines standardized categories for classifying hospital expenses. Helps in organizing financial reports and understanding how resources are allocated.

Key Columns:

- category_id: Unique identifier for each category (auto-incremented).
- category_name: Descriptive name of the category (e.g., Utilities, Supplies, Maintenance).
- description: Optional description explaining the scope or use of the category.

RolePermissions

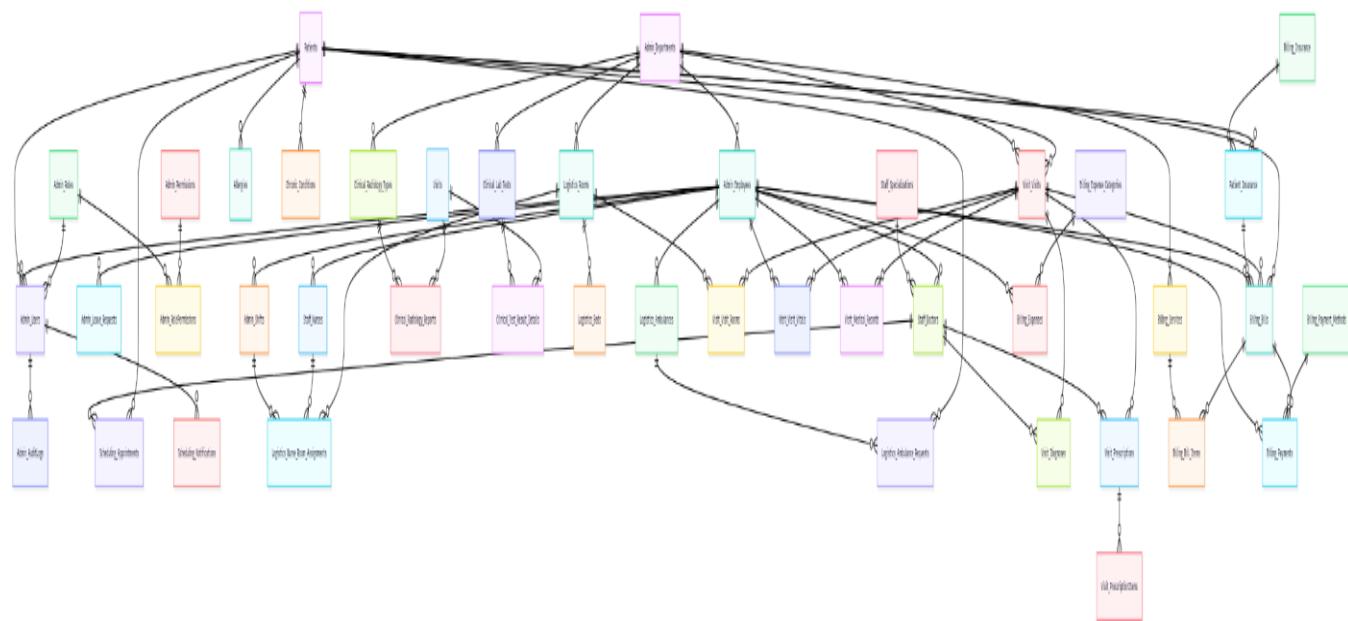
Purpose:

Defines the many-to-many relationship between roles and permissions. This table specifies which permissions are assigned to each system role, enabling role-based access control (RBAC).

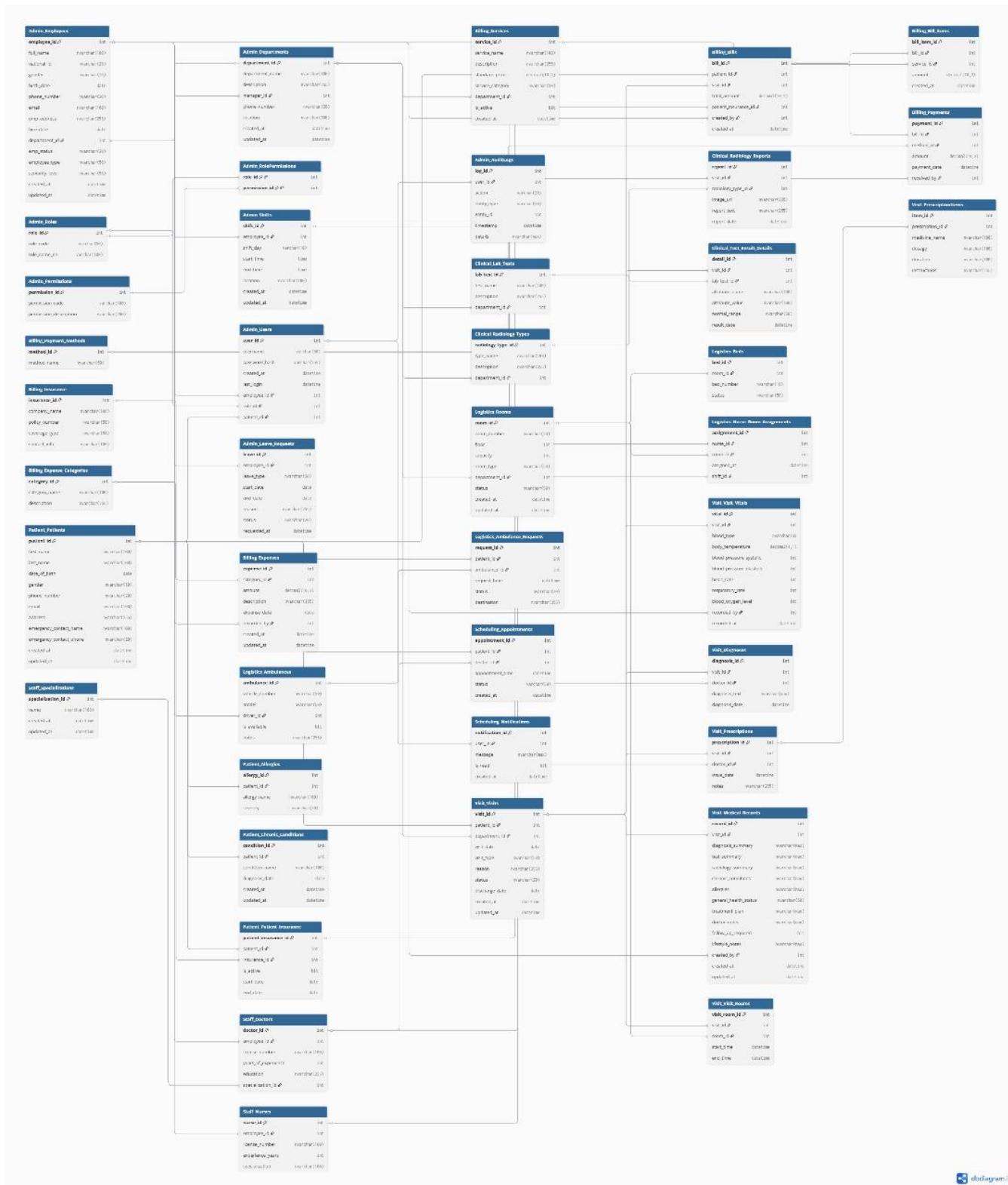
Key Columns:

- role_id: Foreign key referencing the Roles table.
- permission_id: Foreign key referencing the Permissions table.

ERD Diagram



Database Diagram



3. Stored Procedures

Stored Procedures

Stored procedures are precompiled blocks of SQL code that encapsulate logic for performing specific operations within the database. They provide a secure, reusable, and performance-efficient way to interact with data. By centralizing data logic inside stored procedures, the system reduces code duplication, enforces consistency, and simplifies maintenance.

In this project, we implemented **core stored procedures for each main table**, covering the essential operations:

- INSERT
- UPDATE
- DELETE
- SELECT

These core procedures ensure that data manipulation follows standardized rules and access logic throughout the system.

In addition to basic CRUD operations, we also developed custom stored procedures to support specific use cases and reporting needs. For example:

- Retrieving all lab tests performed for a specific patient
- Displaying nurse profiles along with their assigned shifts
- Calculating total bill amounts with insurance discounts
- Fetching visit history with vital signs and medical notes

Stored procedures play a key role in improving security (by limiting direct access to tables) performance (as they're precompiled), and scalability (logic changes can be managed centrally without affecting the UI).

Examples:

```
-- 1. Add New Lab Test
CREATE PROCEDURE Add_Lab_Test
    @test_name NVARCHAR(100),
    @description NVARCHAR(255),
    @department_id INT
AS
BEGIN
    INSERT INTO Lab_Tests (test_name, description, department_id)
        VALUES (@test_name, @description, @department_id)
END
GO

-- 2. Update Lab Test
CREATE PROCEDURE Update_Lab_Test
    @lab_test_id INT,
    @test_name NVARCHAR(100),
    @description NVARCHAR(255),
    @department_id INT
AS
BEGIN
    UPDATE Lab_Tests
        SET test_name = @test_name,
            description = @description,
            department_id = @department_id
    WHERE lab_test_id = @lab_test_id
END
GO

-- 3. Delete Lab Test
CREATE PROCEDURE Delete_Lab_Test
    @lab_test_id INT
AS
BEGIN
    IF EXISTS (
        SELECT 1 FROM Test_Result_Details WHERE lab_test_id = @lab_test_id
    )
    BEGIN
        RAISERROR('Cannot delete the lab test because it is associated with test results.') WITH NOWAIT
        RETURN
    END
    DELETE FROM Lab_Tests
    WHERE lab_test_id = @lab_test_id
END
GO

-- 4. View All Lab Tests
CREATE PROCEDURE Get_All_Lab_Tests
AS
BEGIN
    SELECT * FROM Lab_Tests
END
GO
```

```

-- 9. View Lab Tests for a Specific Patient
CREATE VIEW vw_Patient_Lab_Tests AS
SELECT
    p.patient_id,
    p.first_name + ' ' + p.last_name AS patient_name,
    v.visit_id,
    lt.test_name,
    trd.attribute_name,
    trd.attribute_value,
    trd.normal_range,
    trd.result_date
FROM Patient.Patients p
JOIN Visit.Visits v ON p.patient_id = v.patient_id
JOIN Test_Result_Details trd ON v.visit_id = trd.visit_id
JOIN Lab_Tests lt ON trd.lab_test_id = lt.lab_test_id;
-
Go

CREATE PROCEDURE GetLabTestsForPatient
    @patient_id INT
AS
BEGIN
    SELECT *
    FROM vw_Patient_Lab_Tests
    WHERE patient_id = @patient_id
END
GO

```

4. Data Warehouse Design

As part of the Hospital Management System, a robust Data Warehouse was designed and implemented to support historical analysis, real-time reporting, and strategic decision-making. While the operational database is optimized for daily transactions and data entry, the Data Warehouse provides a denormalized structure ideal for analytics, monitoring, and business intelligence (BI).

To achieve this, we adopted a Galaxy Schema (also known as a Fact Constellation Schema), which includes multiple fact tables sharing common dimension tables. This design allows flexible, multidimensional analysis across different hospital domains such as billing, clinical care, staff workload, visits, and resource utilization.

Purpose and Benefits

- Historical Analysis: Tracks patient visits, diagnoses, vitals, and billing history over time to identify trends and outcomes.
- Operational Monitoring: Supports performance tracking of departments, doctors, staff shifts, and resource occupancy.
- Aggregated Financial Insights: Analyzes expenses, payments, and billing by category, time period, or department.
- Improved Decision Support: Provides hospital administrators and department heads with actionable insights.
- Data Consistency & Centralization: Integrates data from various sources into a single source of truth for reporting.

Architecture Overview

- ETL Process (SSIS): Data is extracted from the transactional database, transformed (cleaned, enriched, aggregated), and loaded into the Data Warehouse using SQL Server Integration Services (SSIS).
- Galaxy Schema Design: Multiple fact tables (e.g., 'Fact_Billing', 'Fact_Clinical', 'Fact_Visits') are linked to shared dimension tables such as 'Dim_Patient', 'Dim_Doctor', 'Dim_Date', and 'Dim_Department'.
- Scheduled Batch Load: ETL jobs are scheduled regularly to update the warehouse while minimizing performance impact on the OLTP system.

Main Components

Fact Tables:

- 'Fact_Billing': Tracks billing transactions, total paid and billed amounts, services count, payment methods, and insurance details.

- 'Fact_Clinical': Contains clinical metrics per visit, including vitals, chronic condition flags, lab results, and radiology indicators.
- 'Fact_Expense': Records expense amounts linked to categories and departments.
- 'Fact_Employee_Workload': Monitors staff attendance, shifts, leave days, and workload metrics.
- 'Fact_Resource': Analyzes room and bed occupancy rates and turnover metrics.
- 'Fact_Visits': Summarizes visit duration, emergency status, follow-ups, outcomes, and patient return patterns.

Dimension Tables:

- 'Dim_Patient': Stores patient demographic details, insurance status, and chronic condition flags.
- 'Dim_Doctor': Includes doctor info, specialization, and department.
- 'Dim_Department': Describes hospital departments and locations.
- 'Dim_Bed', 'Dim_Room': Capture resource structure for occupancy analysis.
- 'Dim_Employee': Represents hospital staff with history tracking ('EffectiveFrom', 'EffectiveTo', 'IsCurrent').
- 'Dim_Insurance': Contains insurance provider data and policy info.
- 'Dim_LabTest', 'Dim_Radiology', 'Dim_PrescriptionDrug': Describe medical services used in analytics.
- 'Dim_Date': A time dimension used to support all date-based analysis.

ETL and Data Flow

1. Extraction: Data is pulled from the OLTP system, including medical records, appointments, bills, and vitals.
2. Transformation: Data is standardized (e.g., formatting, null handling), deduplicated, enriched with keys, and aggregated where needed.
3. Loading: Clean data is inserted into dimension and fact tables in the Data Warehouse, optimized for Power BI and SSRS.

Reporting and Analytics

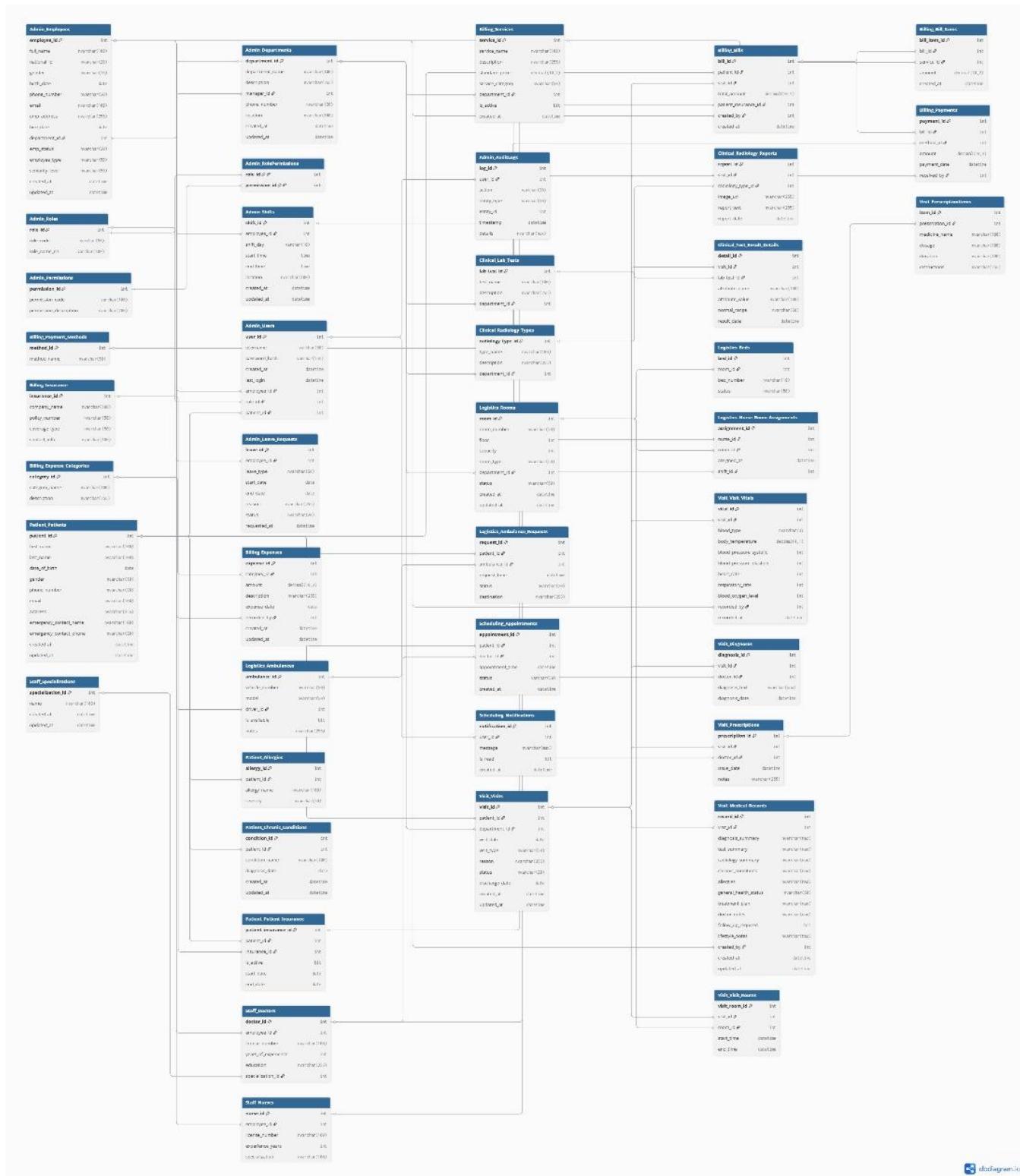
- Power BI Dashboards: Deliver interactive insights into patient care, hospital finances, staff performance, and resource usage.
- SSRS Reports: Provide static and printable reports for billing summaries, department statistics, and clinical metrics.
- Advanced Analytics**: Enables trend analysis, follow-up prediction, workload optimization, and KPI monitoring across hospital services.

Security and Maintenance

- Access Control: BI reports and dashboards are restricted to authorized roles such as Admin, HR, Finance, or Department Head.
- Scheduled ETL Refresh: Data pipelines run during off-hours to update analytical data with minimal disruption.
- Backup & Recovery: Regular backups are maintained to protect the warehouse and support disaster recovery.

The Data Warehouse is a foundational component of the Hospital Management System. It enables hospital leaders and stakeholders to make data-driven decisions, monitor performance, and continuously improve healthcare delivery, operations, and financial outcomes.

Data Warehouse Diagram



5. ETL Process in DWH

The ETL (Extract, Transform, Load) process forms the core pipeline of the Hospital Data Warehouse. It ensures that structured and accurate data from the operational hospital system is extracted, cleaned, transformed, and loaded into the analytical environment to support business intelligence, clinical insights, and strategic decision-making.

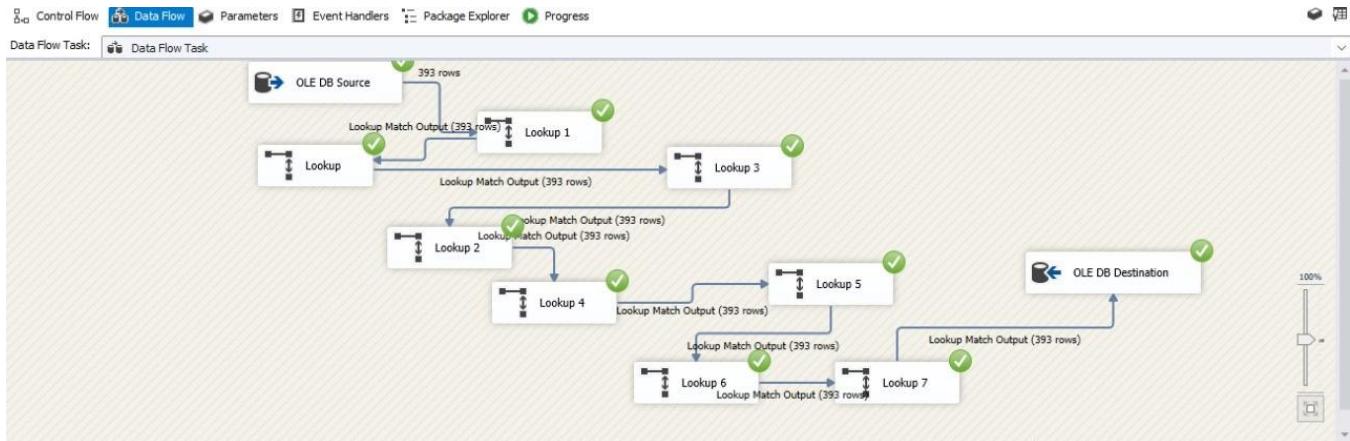
Using SQL Server Integration Services (SSIS), we developed a set of automated ETL packages, each responsible for a specific dimension or fact table. These packages handle the complexities of data integration, such as handling historical changes (SCD - Slowly Changing Dimensions), deriving surrogate keys, managing lookup operations, and ensuring referential integrity.

For example, patient information, visits, billing, clinical records, staff workload, and room occupancy are all extracted from normalized transactional tables, transformed into analysis-ready structures, and loaded into corresponding fact and dimension tables.

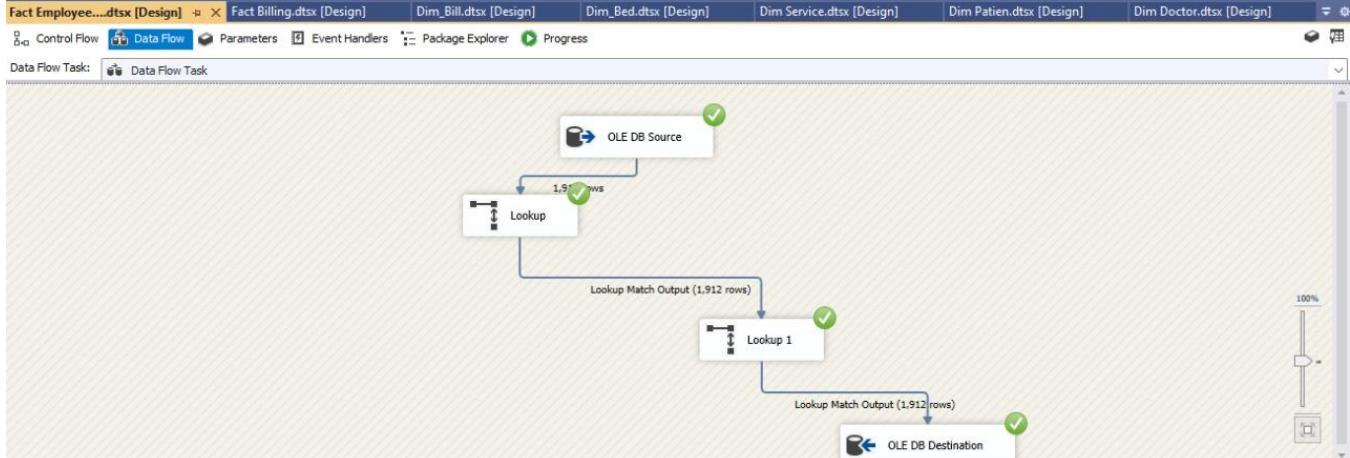
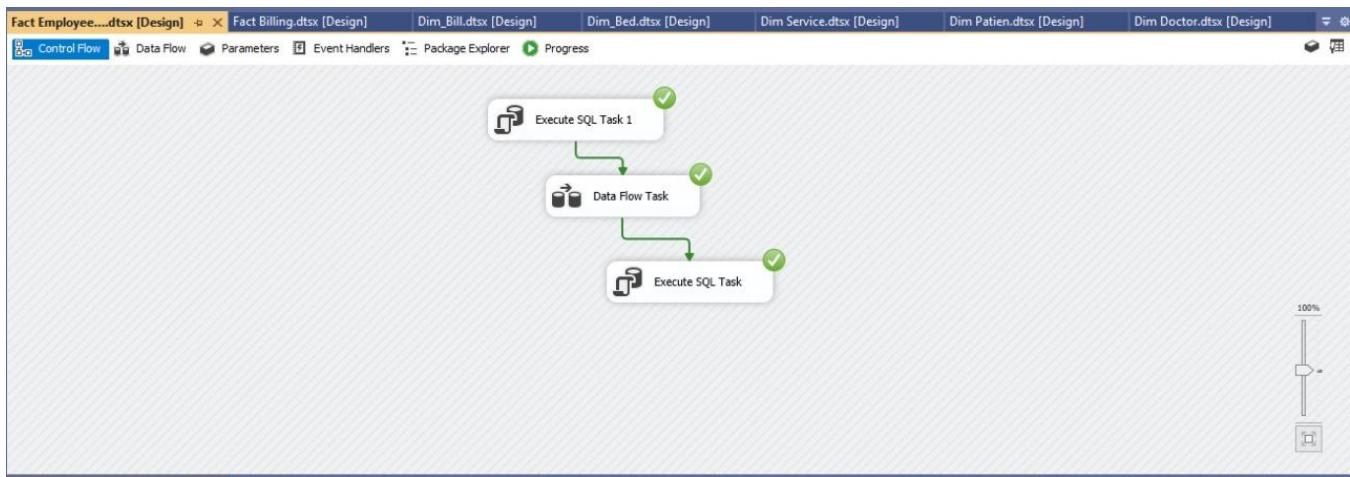
Each ETL package is scheduled to run periodically (e.g., daily or weekly), ensuring the Data Warehouse remains up to date without affecting the performance of the live hospital system.

The following are examples of the SSIS packages implemented to load and transform data into the Data Warehouse.

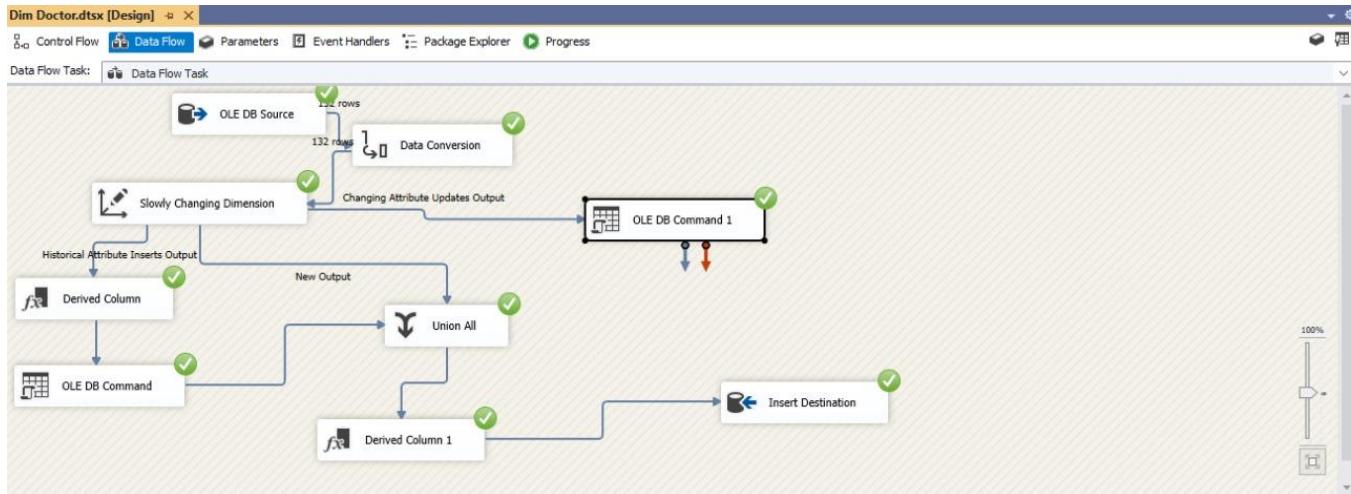
Fact billing



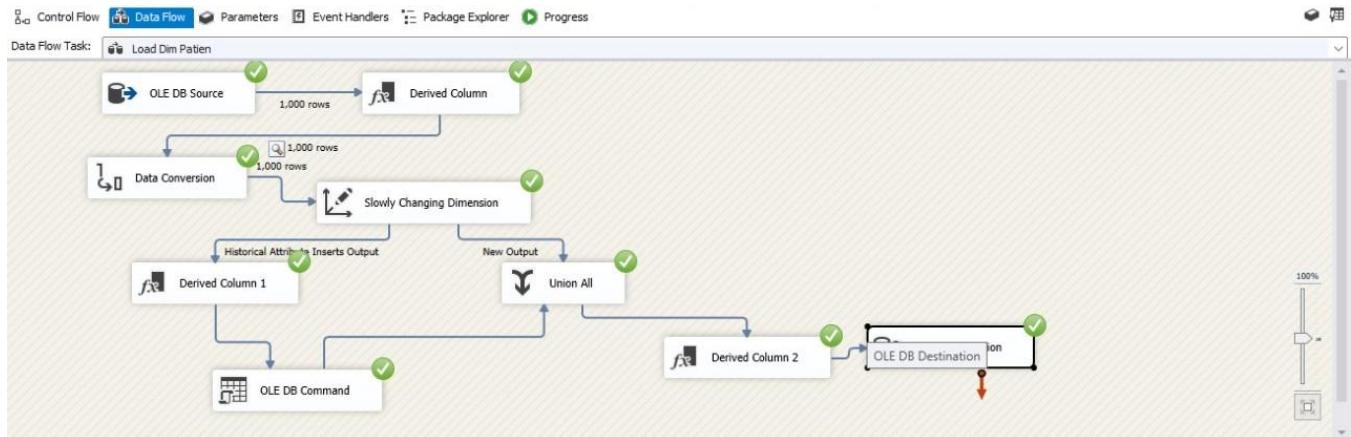
Fact employee



Dim doctor



Dim patient



6. SSRS Reports

The Hospital Management System leverages SQL Server Reporting Services (SSRS) to deliver dynamic, parameterized, and visually rich reports for healthcare operations and decision-making. These reports are directly powered by stored procedures and data warehouse views, ensuring accurate and real-time insights for hospital administrators, doctors, nurses, and other staff.

Overview of SSRS Reports

SSRS is used to create, deploy, and manage a suite of reports that address the core operational, clinical, and financial needs of the hospital. Reports are designed to be interactive, exportable (PDF, Excel, etc.), and accessible through a secure web portal or integrated within the hospital's internal systems.

Reports Implemented

1- Radiology Reports for a Specific Patient

This report displays all radiology reports associated with a specific patient across different visits. It includes the report type, associated images, findings, and report dates. The report is parameterized by patient ID , It helps doctors and radiology staff track imaging results and monitor patient progress over time.

Radiology Report For Patient



Patient ID : 12

patient id	patient name	visit id	report id	radiology type	image url	report text	report date
12	Khaled Khalil	3	5	MRI	https://hospital-imaging.example.com/image_3_3.jpg	MRI lumbar spine reveals disc herniation.	7/11/2025 3:17:00 PM
12	Khaled Khalil	3	6	DEXA Scan	https://hospital-imaging.example.com/image_3_8.jpg	Severe osteoporosis noted in femoral neck.	7/11/2025 2:51:00 PM
12	Khaled Khalil	3	7	X-Ray	https://hospital-imaging.example.com/image_3_1.jpg	Fracture of the distal radius with minimal displacement.	7/11/2025 10:43:00 AM

2- Bill Summary for a Specific Patient with Payment Details

This report displays a summary of all bills issued for a specific patient, including total billed amount, insurance company, payment method, amount paid, remaining balance, and overall payment status. The report helps the finance department and administrative staff monitor the patient's financial obligations and payment history.

Bill details



Invoice No. 15

bill id	created at	patient id	patient name	insurance id	insurance company	visit id	created by	payment method	service name	service price	billed amount	paid amount	remaining amount	payment status	is active
15	7/12/2025 5:11:49 PM	50	Nour Hassan	4	Safenet Care	15	Salma Salem	Debit Card	Fracture Immobilization	400.00	438.27	367.80	70.47	Partially Paid	1
15	7/12/2025 5:11:49 PM	50	Nour Hassan	4	Safenet Care	15	Salma Salem	Debit Card	General Physician Consultation	200.00	438.27	367.80	70.47	Partially Paid	1
15	7/12/2025 5:11:49 PM	50	Nour Hassan	4	Safenet Care	15	Salma Salem	Debit Card	TOTAL		876.54	735.60	140.94	Partially Paid	

3- All Bills Summary with Payment Filters

This report provides a high-level summary of all bills issued in the hospital, including patient details, billing amounts, insurance company, payment status, and payment method. It allows users to filter results based on payment status (e.g., All, Fully Paid, Partially Paid, Unpaid), helping the finance team track and manage billing activity across the organization.

All Bill Summary



Payment Status : All

bill id	created at	patient name	total billed	total paid	remaining amount	payment status
1	7/12/2025 5:11:49 PM	Waleed El-Sayed	756.23	706.77	49.46	Partially Paid
7	7/12/2025 5:11:49 PM	Waleed Farouk	678.90	339.81	339.09	Partially Paid
13	7/12/2025 5:11:49 PM	Mariam Ibrahim	987.65	747.01	240.64	Partially Paid
19	7/12/2025 5:11:49 PM	Waleed Nasser	432.10	375.46	56.64	Partially Paid
25	7/12/2025 5:11:49 PM	Youssef Adel	678.90	381.18	297.72	Partially Paid
31	7/12/2025 5:11:49 PM	Heba El-Sayed	789.01	497.69	291.32	Partially Paid
37	7/12/2025 5:11:49 PM	Aya Hassan	234.56	169.59	64.97	Partially Paid
43	7/12/2025 5:11:49 PM	Mohamed Ibrahim	543.21	271.60	271.61	Partially Paid
49	7/12/2025 5:11:49 PM	Mariam Nasser	543.21	276.43	266.78	Partially Paid
55	7/12/2025 5:11:49 PM	Aya Nasser	987.65	527.70	459.95	Partially Paid
61	7/12/2025 5:11:49 PM	Mona Salama	432.10	240.23	191.87	Partially Paid
67	7/12/2025 5:11:49 PM	Fatma Ibrahim	678.90	410.87	268.03	Partially Paid

4- All Patients with Visits and Diagnoses

This report provides a detailed view of all patients along with their visit history and corresponding diagnoses. For each patient, it displays visit details such as date, type, reason, status, and discharge date, along with any diagnoses made during the visit. It enables doctors, nurses, and administrative staff to track patient history and clinical findings across multiple encounters.

All Patients



patient id	full name	visit id	visit date	discharge date	visit type	reason	status	diagnosis text
4	Waleed El-Sayed	1	4/23/2023 3:00:00 PM	4/23/2023 4:00:00 PM	Outpatient	Fever	Completed	Fracture of the distal radius
7	Amr El-Sayed	2	1/23/2023 3:00:00 PM	1/23/2023 4:00:00 PM	Outpatient	Follow-up Visit	Completed	Physiotherapy session for post-stroke rehabilitation
12	Khaled Khalil	3	11/22/2023 10:00:00 AM	12/15/2023 10:00:00 AM	Inpatient	Skin Rash	Completed	Severe pneumonia with respiratory distress
19	Tarek Hassan	4	3/12/2023 4:00:00 PM	3/12/2023 5:00:00 PM	Outpatient	Surgery Consultation	Completed	Pregnancy - second trimester routine check-up
21	Hassan Mahmoud	5	7/17/2023 2:00:00 PM	7/17/2023 3:00:00 PM	Outpatient	Headache	Completed	Physiotherapy for lower back pain
22	Mariam Mahmoud	6	10/11/2023 8:00:00 AM	10/11/2023 10:00:00 AM	Outpatient	Back Pain	Completed	ACL tear with knee instability
27	Waleed Farouk	7	11/3/2023 1:00:00 PM	11/3/2023 2:00:00 PM	check_up	Routine Checkup	Completed	Degenerative disc disease in lumbar spine
32	Heba Adel	8	7/24/2023 1:00:00 PM	9/16/2023 1:00:00 PM	Inpatient	Skin Rash	Completed	Uncontrolled diabetes mellitus
36	Khaled Ibrahim	9	9/6/2023 11:00:00 AM	9/6/2023 12:00:00 PM	Outpatient	Chest Pain	Completed	Pediatric asthma exacerbation
40	Mahmoud Nasser	10	4/9/2023 11:00:00 AM	4/9/2023 12:00:00 PM	Follow_up	Follow-up Visit	Completed	Severe dehydration

5- All Doctors by Specialization

This report lists all medical specializations along with the doctors assigned to each. It includes the doctor's name, ID, and years of experience. Specializations without any assigned doctors are also displayed, making it easy to identify staffing gaps. The report is useful for HR planning, department management, and administrative oversight.

Doctors by specialization



specialization name	specialization id	doctor name	doctor id	years of experience
Allergy and Immunology	19	No doctors added yet	Not available	Not available
Anesthesiology	25	No doctors added yet	Not available	Not available
Cardiology				
	1	Ahmed Salem	5	11
	1	Aya Farouk	6	6
	1	Esraa Hussein	16	8
	1	Esraa Saad	15	6
	1	Fatma Abdelrahman	14	10
	1	Khaled Abdelrahman	7	9
	1	Khaled Farouk	9	5
	1	Khaled Lotfy	1	12
	1	Mahmoud Hussein	10	14
	1	Mahmoud Saad	13	15
	1	Marwa Abdelrahman	11	12
	1	Salma Abdallah	2	8
	1	Salma Abdelrahman	8	13
	1	Salma Lotfy	4	7

6- All Nurses with Room and Shift Assignments

This report provides a comprehensive view of all nurses along with their assigned rooms and shift schedules. It includes nurse names, specializations, room details (number, floor, type), and shift timing (day, start time, end time). Nurses with no current assignments are also listed for better staffing visibility. This report supports hospital scheduling, nurse management, and operational planning.

Nurse room assignment report



nurse id	nurse name	specialization	room number	floor	room type	shift day	start time	end time
1	Mahmoud Farouk	Emergency Nursing	PT101	1	Treatment Room	Sunday	08:00	16:00
2	Esraa Saad	Emergency Nursing	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned
3	Mahmoud Kamel	Emergency Nursing	G303	3	Patient Room	Thursday	07:00	15:00
3	Mahmoud Kamel	Emergency Nursing	ICU-2	2	ICU	Tuesday	15:00	23:00
4	Mohamed Saad	Emergency Nursing	P203	2	Patient Room	Wednesday	15:00	23:00
4	Mohamed Saad	Emergency Nursing	PT102	1	Treatment Room	Wednesday	08:00	16:00
4	Mohamed Saad	Emergency Nursing	P201	2	Patient Room	Tuesday	07:00	15:00
4	Mohamed Saad	Emergency Nursing	G306	3	Delivery Room	Monday	16:00	00:00
5	Mahmoud Abdallah	Emergency Nursing	G304	3	Patient Room	Monday	15:00	23:00
6	Esraa Kamel	Emergency Nursing	E101	1	Patient Room	Tuesday	07:00	15:00
6	Esraa Kamel	Emergency Nursing	P204	2	Patient Room	Wednesday	08:00	16:00
6	Esraa Kamel	Emergency Nursing	O303	3	Patient Room	Thursday	16:00	00:00
6	Esraa Kamel	Emergency Nursing	OR-4	3	Operating Room	Saturday	16:00	00:00
6	Esraa Kamel	Emergency Nursing	PT101	1	Treatment Room	Wednesday	15:00	23:00
7	Hassan Hussein	Emergency Nursing	P201	2	Patient Room	Saturday	07:00	15:00

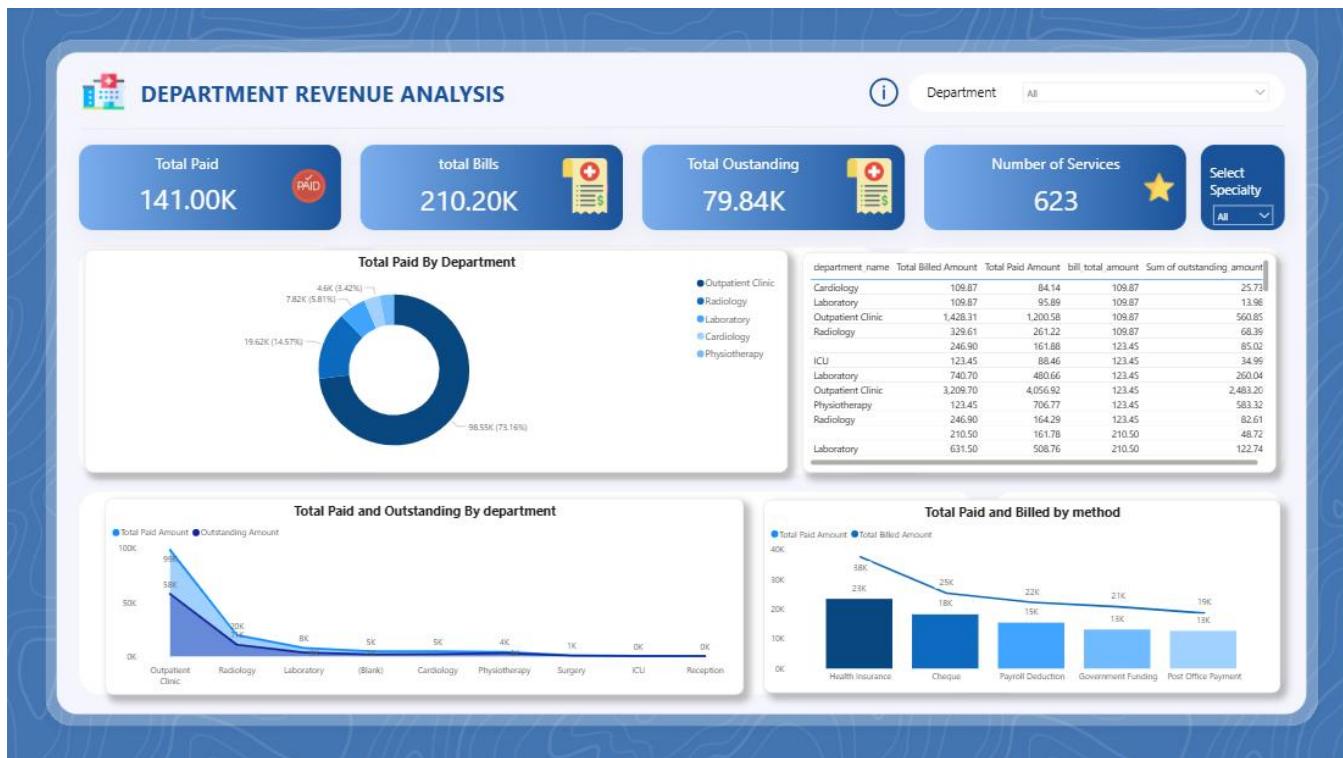
7. Power BI Dashboard

To provide real-time insights and support data-driven decision-making, we developed a series of interactive dashboards using Microsoft Power BI. These dashboards visualize key hospital metrics such as patient visits, billing performance, expenses, employee workload, and resource utilization. They are designed to help hospital management monitor operations efficiently and identify trends, bottlenecks, and areas for improvement at a glance.

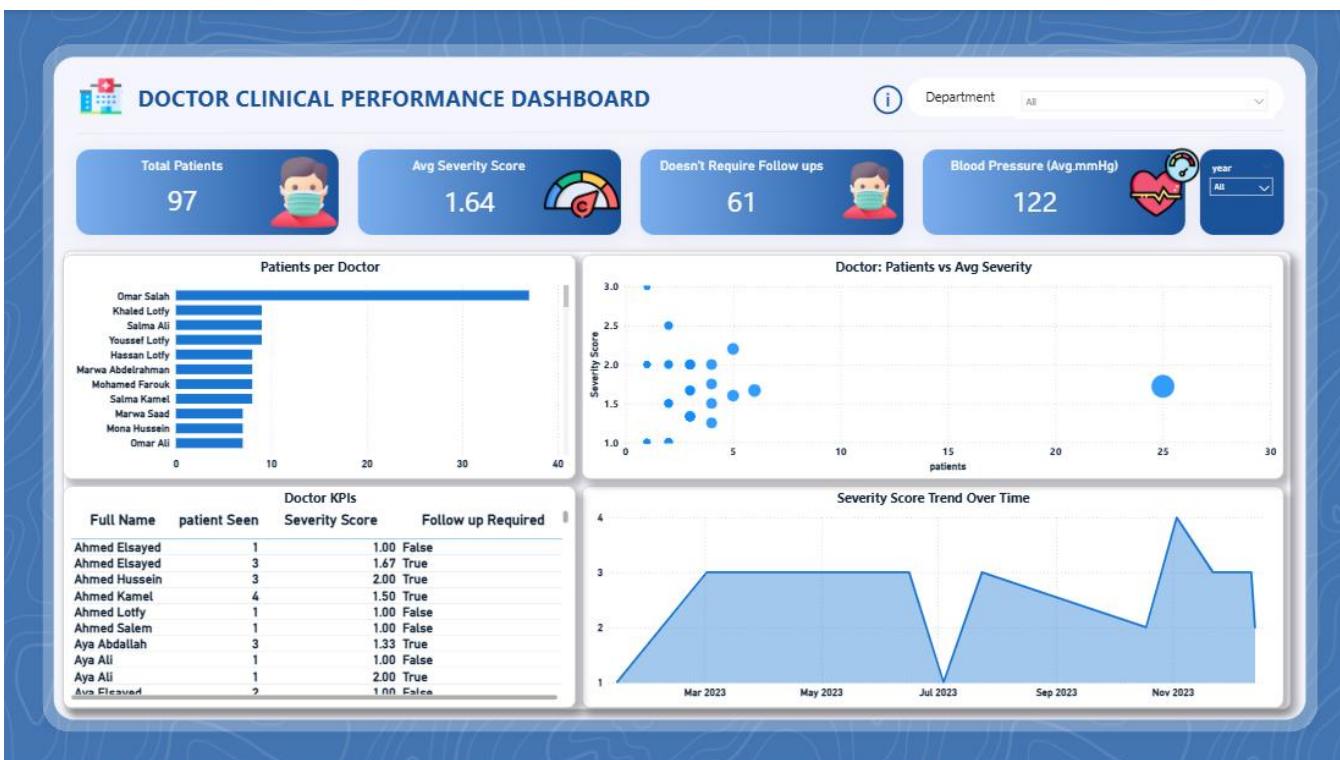
A total of 21 dashboards were created, each focusing on a specific aspect of hospital operations, ensuring comprehensive and focused analytics coverage.

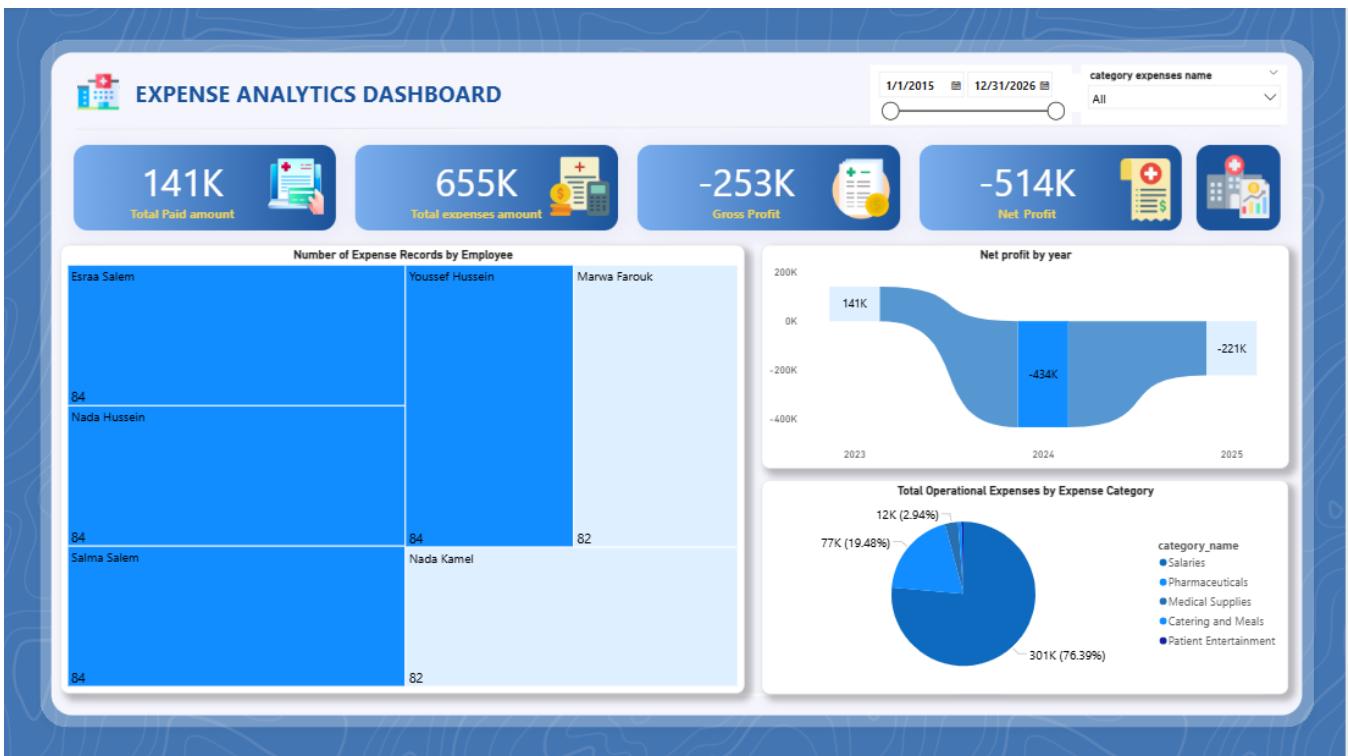
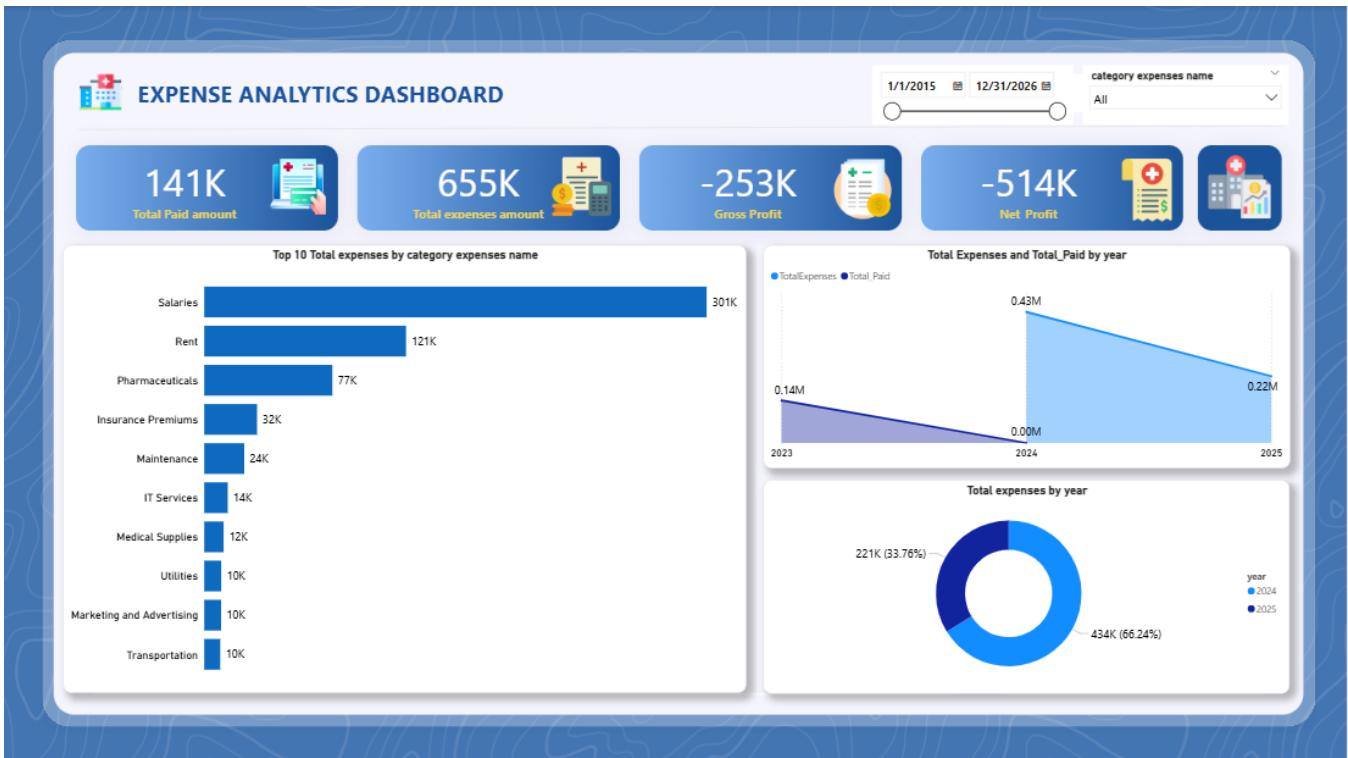


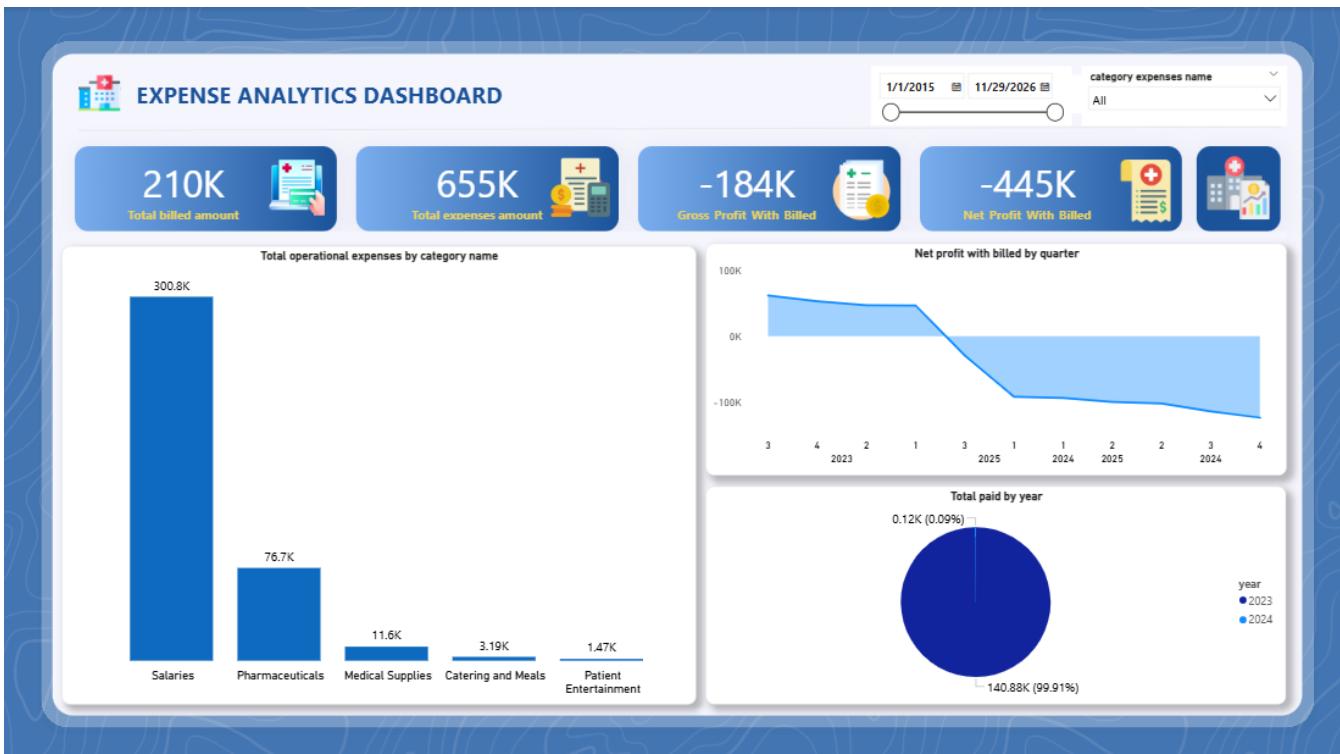
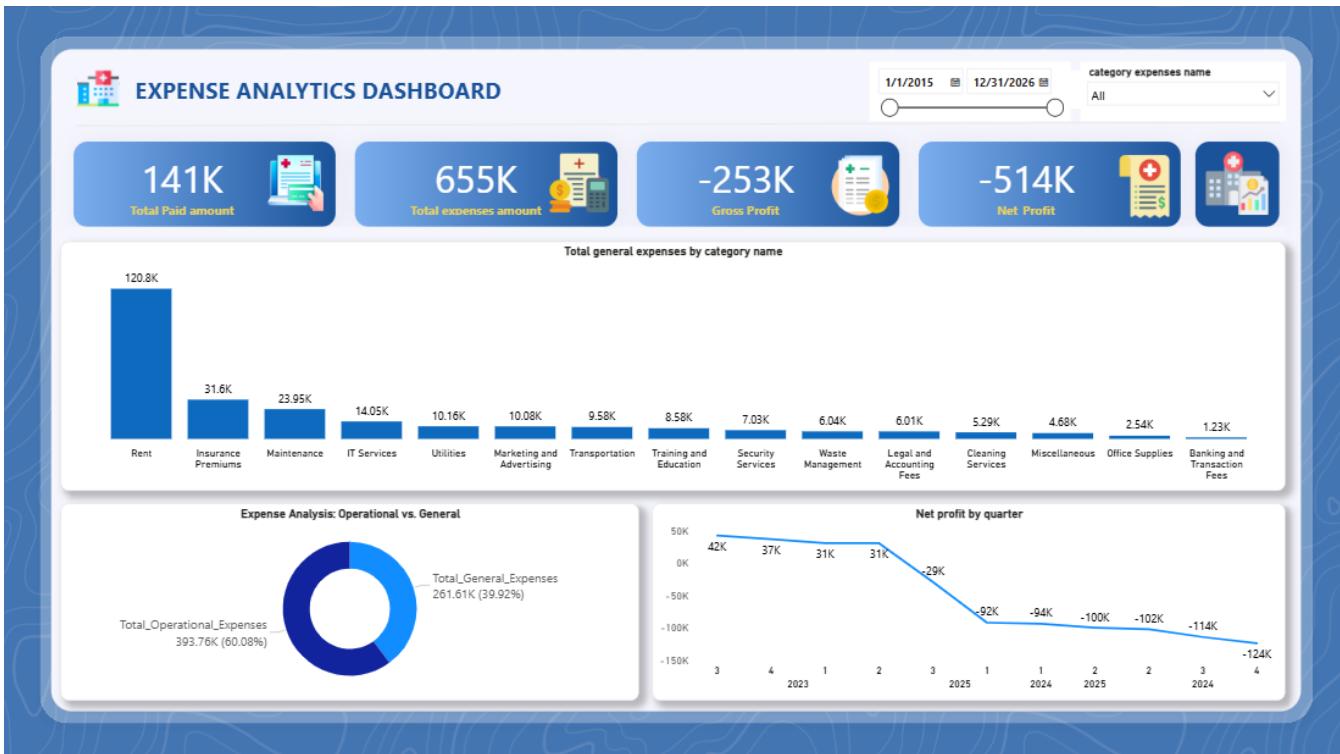


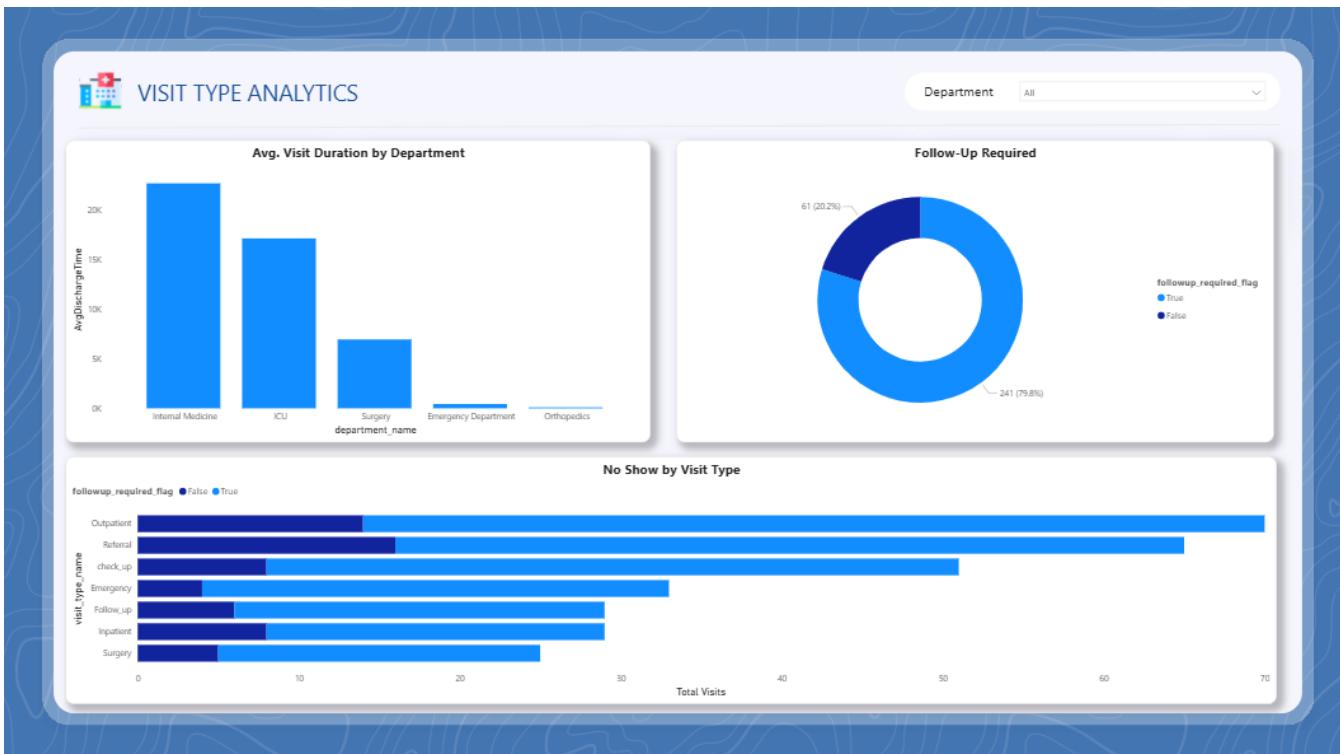




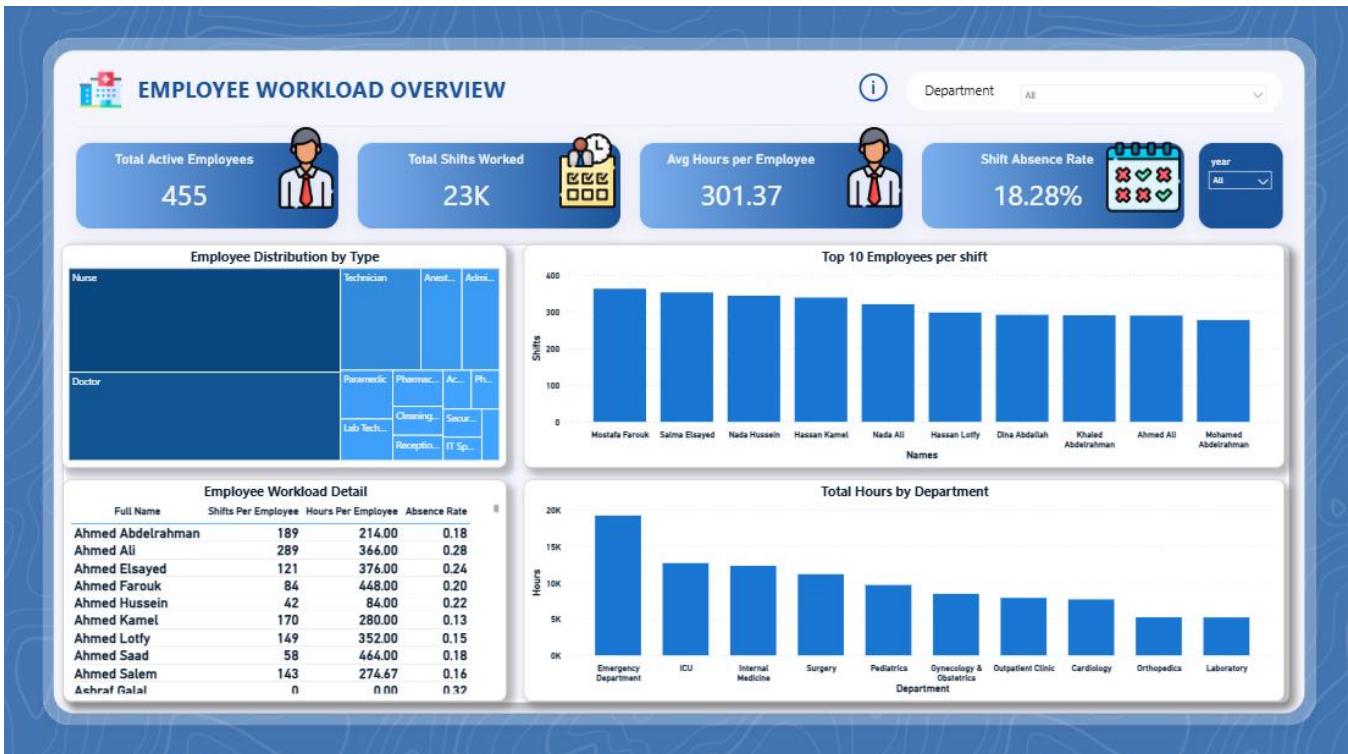


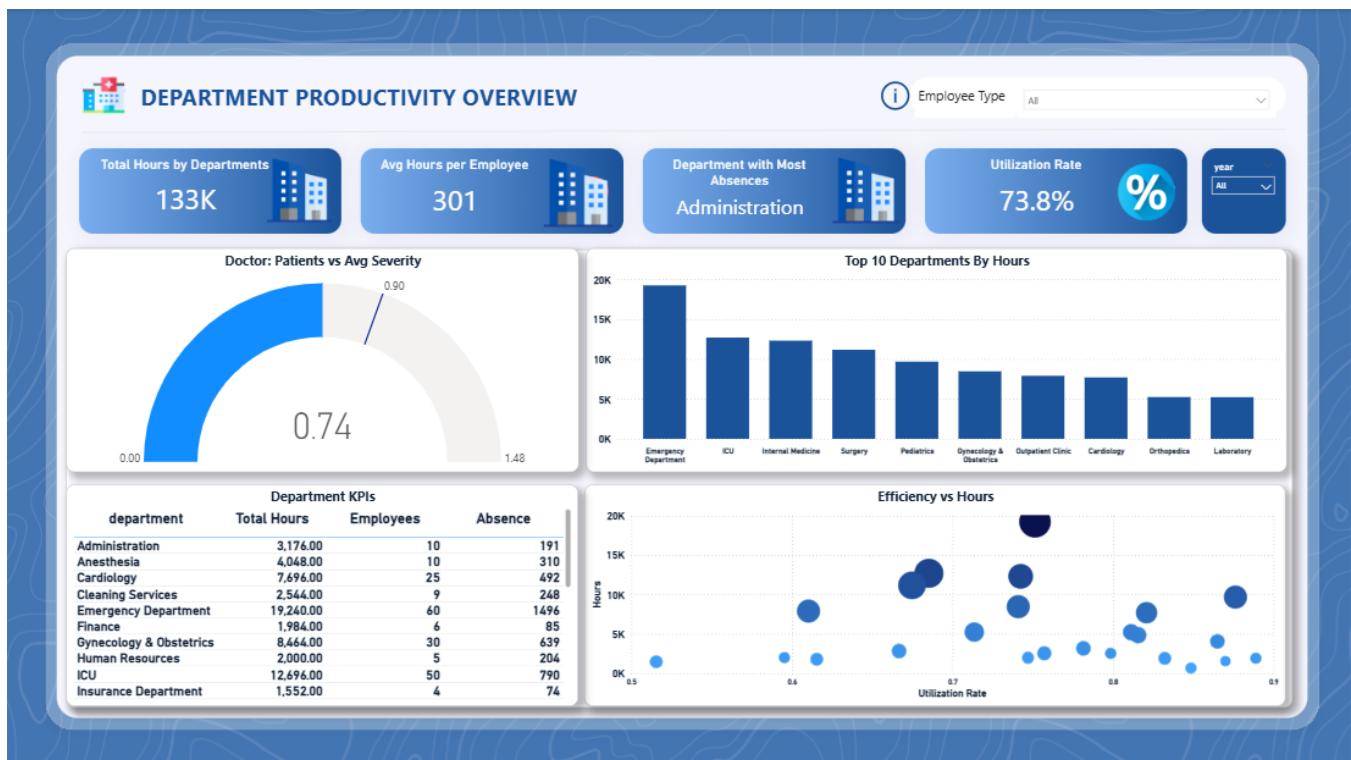












8. Website

This web-based application is designed for use by hospital staff to:

- Maintain electronic medical records.
- Track and manage billing and payments.
- Monitor vitals and lab results.
- Generate reports and summaries.
- Allow secure role-based access.

Overall Description:

1- Product Perspective

The system is a modular web application with different roles (e.g., admin, nurse, doctor). It contains several modules (described below).

2- Product Functions

- Patient Management: Add, update, and view patient data.
- Medical Records: View diagnoses, allergies, chronic conditions, and overall health status.
- Vitals: Record and monitor vital signs (not shown, assumed based on menu).
- Lab Tests: Display blood glucose, kidney function, and coagulation profiles.
- Billing: Manage invoices, track paid/unpaid balances, and view service details.
- Reports: Generate statements and reports (assumed).
- Settings: Configure system or user preferences.

3- User Classes and Characteristics

- Admin: Full system access.
- Doctor: Access to patients, medical records, lab results.
- Nurse: Access to vitals, patients, and reports.
- Accountant: Access to billing and reports.

System Features:

1- Patient Management

- View list of patients.
- Search by name, ID.
- View and edit basic demographic info.

2- Medical Records

- Show diagnosis, allergies, chronic diseases, and general health status.
- Group records by visit ID.
- Tabs for overview, details, and test results.

3- Vitals (assumed)

- Record vitals like blood pressure, heart rate, temperature.
- View trends over time.

4- Lab Tests

- Categorized lab results (Blood Glucose, Kidney, Coagulation).
- Normal/Low/High status indicators.
- Date of test included.

5- Billing Management

- Multiple bills per patient.
- Service breakdown with cost.
- Payment history (e.g., Government Funding).
- Summary: Total, Paid, Balance.

6- Reports (assumed)

- Export data (PDF, Excel).
- Print billing or test summaries.

External Interface Requirements:

1- User Interface

Clean, card-based UI with tabs and toggles.

Sidebar menu with icons and labels.

Responsive layout for web use.

2- Hardware Interface

Web browser-based; works on PCs, tablets.

3- Software Interface

- Backend: Likely .NET Core (based on earlier convo).
- Database: Microsoft SQL Server.
- Frontend: HTML/CSS/JS + React (assumed from design).

Non-functional Requirements:

- Performance: Should handle 100+ concurrent users.
- Security: Role-based access, session timeout.
- Usability: Simple navigation for non-technical users.
- Availability: 99.5% uptime expected.
- Scalability: Allow future modules like Appointments or Prescriptions.

Assumptions and Constraints:

- The system is hosted internally or on a private cloud.
 - All users must be registered and authenticated.
 - Internet connection required.
7. Appendix
- Sample user: Marwa Farouk (seen in screenshots).
 - Sample test results: INR value flagged low.
 - Sample bill: #1 with a government payment.

MediCare Hospital System

Patients

Saturday, July 19, 2025

Search... M Marwa Farouk Role

Patients Management

Search by ID or name... + Add New Patient

ID	Name	Date of Birth	Gender	Phone	Email	Actions
1	Dalia Mohamed	1935-08-29	Female	01273998299	dalia.mohamed@example.com	View Edit Delete
2	Tarek Ibrahim	1929-07-20	Male	01532484865	tarek.ibrahim@example.com	View Edit Delete
3	Waleed Mahmoud	2001-08-13	Male	01162948146	waleed.mahmoud@example.com	View Edit Delete
4	Waleed El-Sayed	1925-11-18	Male	01584455388	waleed.el-sayed@example.com	View Edit Delete
5	Amr Ibrahim	1967-07-17	Male	01095011823	amr.ibrahim@example.com	View Edit Delete
6	Waleed Salama	2003-08-20	Male	01144567065	waleed.salama@example.com	View Edit Delete
7	Amr El-Sayed	1993-05-10	Male	01084065793	amr.el-sayed@example.com	View Edit Delete
8	Mariam Nasser	1948-08-31	Female	01238737358	mariam.nasser@example.com	View Edit Delete
9	Mariam Salama	1928-04-22	Female	01109243439	mariam.salama@example.com	View Edit Delete

Activate Windows
Go to Settings to activate Windows.

MediCare Hospital System

Medical Records

Saturday, July 19, 2025

Search... M Marwa Farouk Role

Medical Record

Visit ID: 1

Overview Details Test Results

Diagnosis Fracture of the distal radius	Allergies Pollen	Chronic Conditions Asthma	Health Status Good
--	--	---	---

Activate Windows
Go to Settings to activate Windows.

Medical Record

Visit ID: 1

Overview Details Test Results

Treatment Plan
Immobilization and follow-up with orthopedic specialist.

Doctor's Notes
Patient stable. Radiology shows healing fracture and no immediate concerns with hernia or obstetric findings. Follow-up scheduled for cast removal and physical therapy.

Lifestyle Notes
Maintain healthy diet and regular exercise. Avoid pollen.

Follow-up
Required

Activate Windows
Go to Settings to activate Windows.

Medical Records

Saturday, July 19, 2025

Search...

Medical Record

Visit ID: 1

Overview Details Test Results

Radiology Results
Small hiatal hernia seen on contrast study.
Obstetric ultrasound: fetus 20 weeks, normal biometry.

Lab Tests
NULL

Activate Windows
Go to Settings to activate Windows.

MediCare Hospital System

Vitals

Saturday, July 19, 2025

Search... M Marwa Farouk Role

Patient Vitals Dashboard

Body Temperature: **37.5 °C** Above Normal
Normal range: 36.1-37.2°C

Blood Pressure: **133/100 mmHg** Above Normal
Normal range: 90-120mmHg

Heart Rate: **84 bpm** Within Normal Range
Normal range: 60-100bpm

Respiratory Rate: **14 breaths/min** Within Normal Range
Normal range: 12-20breaths/min

Blood Oxygen: **85 %** Below Normal
Normal range: 95-100%

Blood Type: **A-** Recorded

Vitals Summary
● Normal ● Caution ● Critical

Activate Windows
Go to Settings to activate Windows.

MediCare Hospital System

Lab Tests

Saturday, July 19, 2025

Search... M Marwa Farouk Role

Laboratory Tests

Blood Glucose

Fasting	90.96	70-100 mg/dL	● Normal
Postprandial	105.48	100-140 mg/dL	● Normal

7/2/2025

Kidney Function Test

Creatinine	1.26	0.6-1.3 mg/dL	● Normal
Urea	11.75	7-20 mg/dL	● Normal
BUN	10.2	7-22 mg/dL	● Normal

7/2/2025

Coagulation Profile

PT	12.77	11-13.5 sec	● Normal
INR	0.48	0.8-1.1	● Low
aPTT	31.8	30-40 sec	● Normal

7/2/2025

Activate Windows
Go to Settings to activate Windows.

Billing Management

Search by Bill ID or Amount...

Bill #1 Total: \$756.23 Paid: \$706.77 Balance: \$49.46

Services

Ultrasound Abdomen	\$378.11
General Physician Consultation	\$378.12

Payment History

Government Funding	\$706.77	4/23/2023
--------------------	----------	-----------

Bill #2 Total: \$890.15 Paid: \$461.34 Balance: \$428.81

Bill #3 Total: \$345.67 Paid: \$298.54 Balance: \$47.13

Bill #4 Total: \$---.--- Paid: \$---.--- Balance: \$---.---

Activate Windows
Go to Settings to activate Windows.

Print Statements Export