Project: Patient Health Records and Treatment Analytics in a Multi-specialty Hospitals

Student Names:

- Vivekbhai Ashokbhai Radadiya[002856027]
- Vishwaben Umeshbhai Bhadiyadara[002636314]

Group No: 08

I. Exclusive Summary:

In the current healthcare landscape, our hospital holds extensive patient data yet struggles to leverage its full potential for improved patient care, resource allocation, and proactive disease trend prediction. To address this challenge, the institution aims to employ advanced database management, analytics techniques, and data governance strategies. The goal is to structure, integrate, and analyze data effectively to enhance patient care quality, streamline operations, and forecast future resource needs. This initiative focuses on cutting-edge database design, comprehensive data warehousing, and robust governance to bridge the gap between raw data and actionable insights.

II. Introduction:

In today's dynamic healthcare environment, the pivotal role of data-driven insights in shaping patient care and operational strategies cannot be overstated. However, despite our hospital's possession of a vast repository of patient health records and treatment histories, there's a glaring discrepancy - we're not harnessing this data's full potential. This disconnect not only impacts immediate patient care outcomes but also impedes efficient resource allocation and our ability to foresee disease trends over time.

To confront this multifaceted challenge head-on, our institution recognizes the urgent need to leverage advanced database management, analytics techniques, and robust data governance strategies. Our goal is clear: meticulously structure, integrate, and deeply analyze this wealth of data. Through this concerted effort, we aim to achieve several critical objectives - notably, to significantly elevate the quality of patient care, streamline operational workflows, and accurately predict future resource needs.

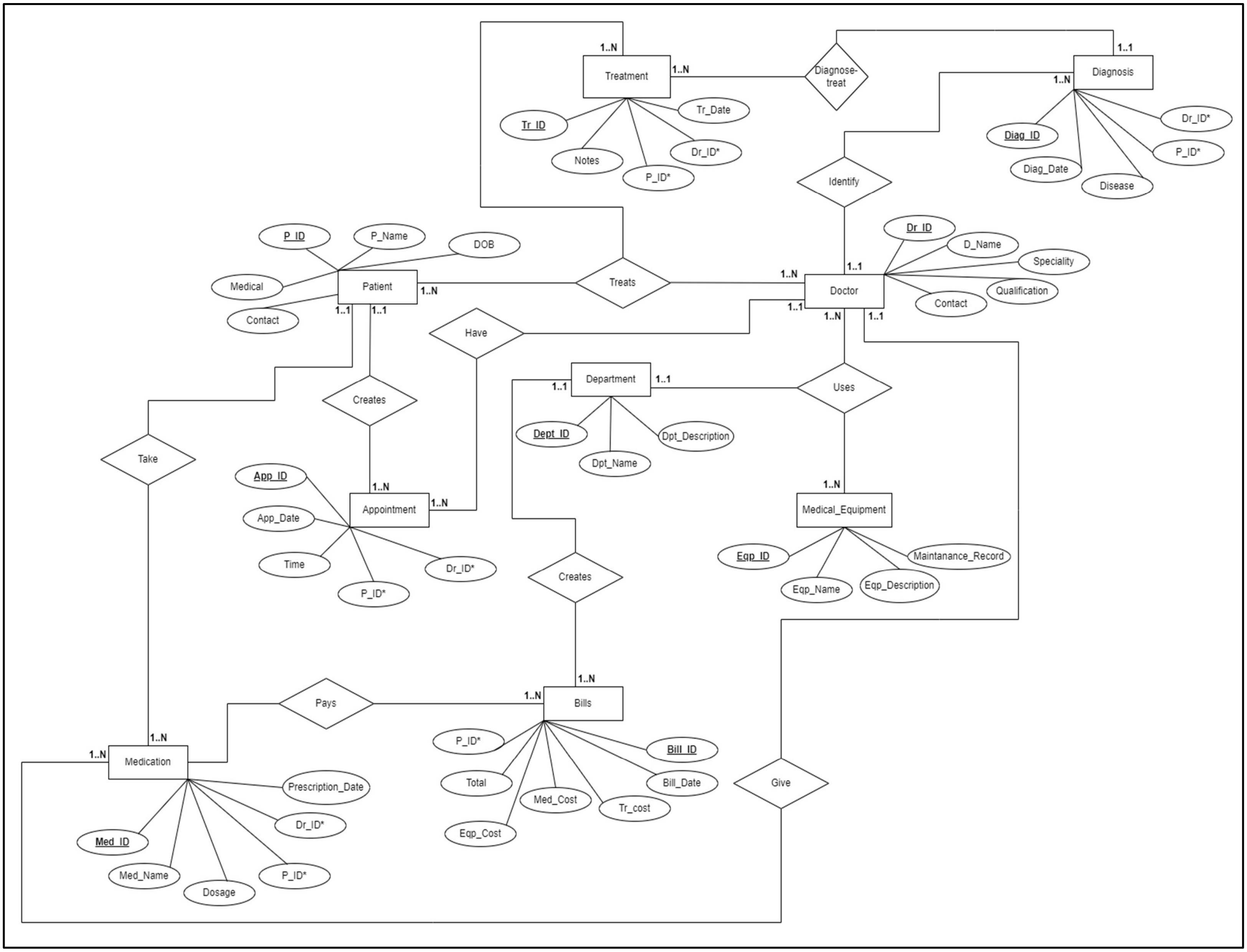
This ambitious initiative is rooted in the fundamental principles of cutting-edge database design, comprehensive data warehousing, and robust governance practices. The ultimate aim is to bridge the prevailing gap between raw data and actionable insights, enabling our hospital to transform information into tangible improvements in patient care and operational efficiency. This endeavor embodies our commitment to unlocking the latent potential within our data repository, propelling us toward a future where data empowers us to make informed, impactful decisions in healthcare delivery.

Milestone: 1

III. Real-World Business Problem Definition:

In today's healthcare landscape, where data-driven insights play a pivotal role in shaping patient care and operational strategies, our hospital finds itself at a crossroads. Despite possessing a vast repository of patient health records and detailed treatment histories, the institution grapples with harnessing this data's potential to its fullest. This disconnect not only impacts the immediate patient care outcomes but also hinders efficient resource allocation and the capability to proactively anticipate disease trends and patterns over the years. To address this multifaceted challenge, there's an imperative need to employ advanced database management, analytics techniques, and data governance strategies. By meticulously structuring, integrating, and subsequently analyzing this treasure trove of data, the hospital aspires to achieve a manifold objective: significantly enhance the quality of patient care, refine and streamline operational workflows, and accurately forecast future resource requirements. This endeavor is rooted in the foundational principles of cutting-edge database design, comprehensive data warehousing, and robust governance, aiming to bridge the existing gap between raw data and actionable insights.

IV. <u>EER Diagram:</u>



<u>In EER: PK – Primary Key (   ) & FK – Foreign Key (-------)</u>

<u>Business Entities & Attributes:</u>

1. **Patient:** Individuals who receive medical care.
   - P_ID (PK), P_Name, DOB, P_Contact, Medical_History
2. **Doctor:** Medical professionals providing care and treatment.
   - Dr_ID (PK), D_Name, Specialties, Dr_Contact, Qualifications
3. **Department:** Various medical specialties like Cardiology, Neurology, etc.
   - Dpt_ID (PK), Dpt_Name, Dpt_Description

4. **Treatment:** Medical procedures or interventions given to patients.
   - Tr_ID (PK), Tr_Date, Notes, P_ID (FK), Dr_ID (FK)
5. **Diagnosis:** The identification of a disease or condition.
   - Diag_ID (PK), Diag_Date, Disease, P_ID (FK), Dr_ID (FK)
6. **Appointment:** Scheduled visits of patients with doctors.
   - App_ID (PK), App_Date, Time, P_ID (FK), Dr_ID (FK)
7. **Medication:** Prescribed drugs to patients.
   - Med_ID (PK), Med_Name, Dosage, Prescription_Date, P_ID (FK), Dr_ID (FK)
8. **Medical Equipment:** Equipment used in treatments.
   - Eqp_ID (PK), Eqp_Name, Eqp_ Description, Maintenance_Record
9. **Bills:** Financial transactions for treatments provided.
   - Bill_ID (PK), Bill_Date, Tr_Cost, Med_Cost, Eqp_Cost, Total, P_ID (FK)

Primary Key: Bold & Underline

Foreign Key: * & Underline

Relationships:

1. **Patient – Appointment (One – Many):**
   - A patient (Patient) can schedule multiple appointments (Appointment).
   - Each appointment is associated with one patient.
2. **Patient – Medication (One – Many):**
   - A patient (Patient) can have multiple medicines (Medication).
   - Each Medicine is associated with one patient.
3. **Patient – Bill (One – Many):**
   - A Patient receives Bills for Treatments, Medications, and other services.
   - A Bill is associated with one Patient.
4. **Patient – Doctor (Many – Many):**
   - A patient (Patient) can be treated by multiple doctors (Doctor).
   - A doctor can have multiple patients.
5. **Patient – Treatment (One – Many):**
   - A patient (Patient) can receive multiple medical treatments (Treatment).
   - Each treatment is associated with one patient.
6. **Patient – Diagnosis (One – Many):**
   - A patient (Patient) can have multiple diagnoses (Diagnosis).
   - Each diagnosis is associated with one patient.
7. **Appointment – Doctor (Many – One):**
   - A Doctor can have multiple appointments (Appointment).
   - Each appointment is associated with one Doctor.
8. **Doctor – Treatment (One – Many):**
   - A Doctor can do multiple medical treatments (Treatment).
   - Each treatment is associated with one Doctor.
9. **Doctor – Medication (One – Many):**

- A doctor (Doctor) can give multiple medicines (Medication).
- Each Medicine is associated with one Doctor.

10. Doctor - Medical Equipment (Many – Many):

- Doctors may use Medical Equipment in their treatments.
- Medical Equipment may be associated with specific Doctors who use it.

11. Doctor – Diagnosis (One – Many):

- Each doctor can make multiple diagnoses over time.
- Each diagnosis is associated with one doctor.

12. Department – Doctor (One - Many):

- A Doctor belongs to a specific Department or Medical Specialty.
- A Department has multiple Doctors.

13. Department – Appointment (One – Many):

- Appointments are scheduled within specific Departments.
- A Department can have multiple Appointments.

14. Diagnosis – Treatment (One – Many):

- Many treatments (Treatment) may be associated with one specific diagnosis (Diagnosis).
- Each treatment is typically linked to one diagnosis that indicates the specific condition or disease being addressed by that treatment.

15. Treatment – Appointment (Many – One):

- A single appointment can result in one or more treatments or medical procedures.
- Each treatment can be associated with a single appointment.

16. Treatment – Medication (Many – Many):

- A single treatment can involve the administration of multiple medications.
- A single medication can be part of multiple treatments.

17. Treatment - Medical Equipment (Many – Many):

- Some Treatments may require the use of specific Medical Equipment.
- Medical Equipment is used in Treatments.
- A Treatment may be associated with one or more pieces of Medical Equipment.

18. Treatment – Bill (One – Many):

- A single treatment can lead to the creation of one or more bills.
- Each bill is associated with a specific treatment.

19. Diagnosis – Medication (One – Many):

- One diagnosis can result in multiple medications being prescribed to the same patient by the same doctor.
- Each medication entry corresponds to a particular prescription for the same diagnosis.

20. Diagnosis – Treatment (Many – Many):

- Each Diagnosis can be led to multiple treatments.
- Each Treatment can be linked to multiple Diagnoses.

21. Medication – Bill (One – Many):

- A single medication prescription can result in one or more billing records.
- For Each bill, there is a single medication prescription.

Reference Data:

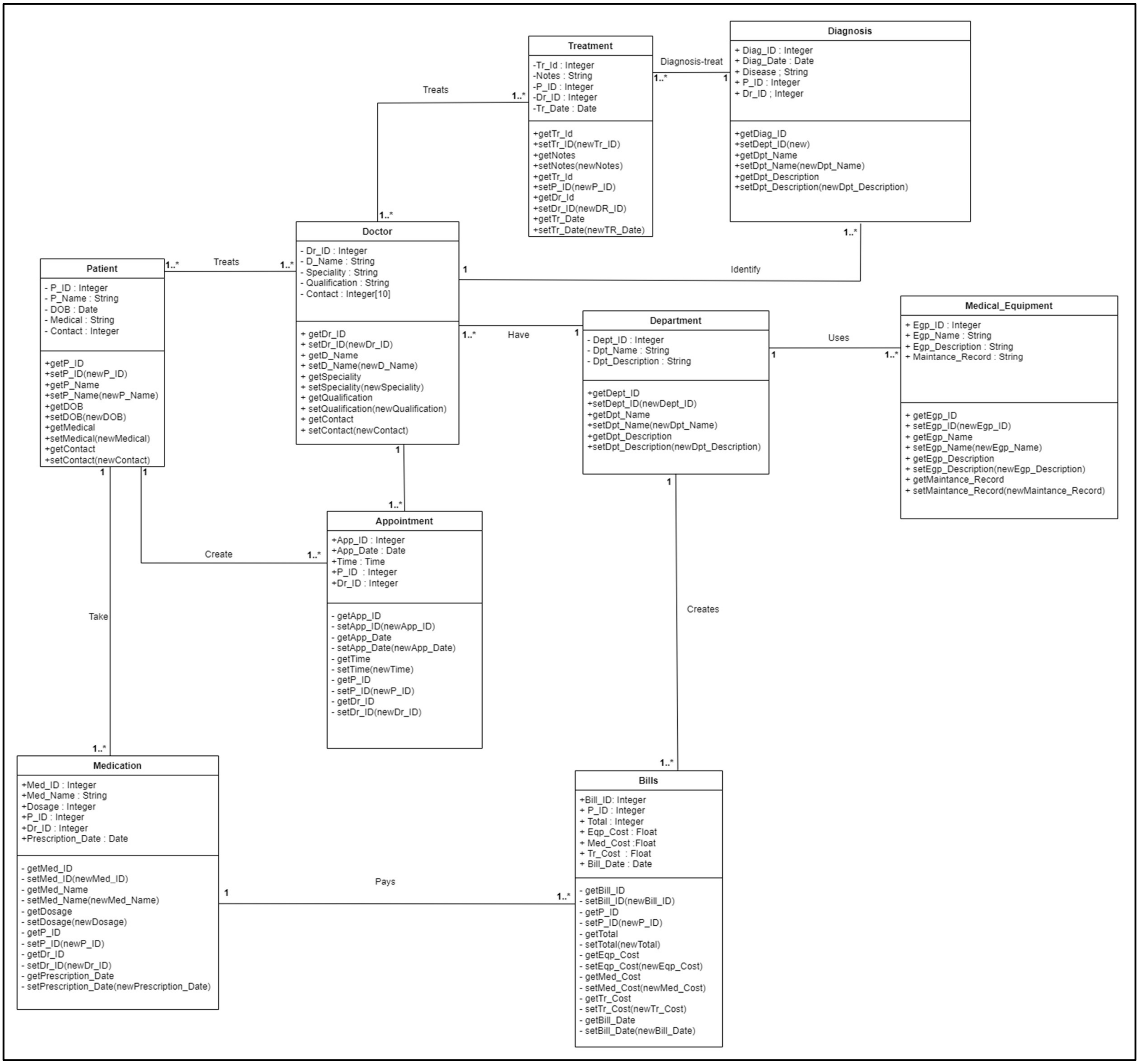
- Department (Department information like Cardiology, Neurology, etc.)
- Medical Equipment (Information about medical equipment used in treatments)
- Doctor (Information about medical professionals)

Transactional Data:

- Patient (Information about individuals who receive medical care)
- Treatment (Information about medical procedures or interventions given to patients)
- Diagnosis (Information about the identification of diseases or conditions)
- Appointment (Scheduled visits of patients with doctors)
- Medication (Information about prescribed drugs to patients)
- Bills (Financial transactions for treatments provided)

Milestone: 2

V. UML Class Diagram:



A Relational Model accurately mapped from the Conceptual Model:

1. Treatment (**Treatment_ID**, Notes, Patient_ID*, Doctor_ID*, Treatment_Date)
2. Diagnosis (**Diagnosis_ID**, Diagnosis_Date, Disease, Patient_ID*, Doctor_ID*)
3. Patient (**Patient_ID**, Patient_Name, Patient_DOB, Medical, Patient_Contact)
4. Doctor (**Doctor_ID**, Doctor_Name, Speciality, Qualification, Doctor_Contact, Department_ID*)
5. Appointment (**Appointment_ID**, Appointment_Date, Appointment_Time, Patient_ID*, Doctor_ID*)
6. Department (**Department_ID**, Department_Name, Department_Description)
7. Medical_Equipment (**Equipment_ID**, Equipment_Name, Equipment_Description, Maintance_Record, Department_ID*)
8. Medication (**Medication_ID**, Medication_Name, Dosages, Patient_ID*, Doctor_ID*, Prescription_Date, Treatment_ID*)
9. Bills (**Bill_ID**, Patient_ID*, Total, Equipment_Cost, Medical_Cost, Treatment_Cost, Bill_Date, Department_ID*)
10. Treats (Patient_ID*, Treatment_ID*, Doctor_ID*)
11. Uses (Doctor_ID*, Equipment_ID*)

Normalization up to 3.5 NF:

1. Treatment (**Treatment_ID**, Notes, Patient_ID*, Doctor_ID*, Treatment_Date)
2. Diagnosis (**Diagnosis_ID**, Diagnosis_Date, Disease, Patient_ID*, Doctor_ID*)
3. Patient (**Patient_ID**, Patient_Name, Patient_DOB, Patient_Contact)
4. Doctor (**Doctor_ID**, Doctor_Name, Speciality, Qualification, Doctor_Contact)
5. Appointment (**Appointment_ID**, Appointment_Date, Appointment_Time, Patient_ID*, Doctor_ID*)
6. Department (**Department_ID**, Department_Name, Department_Description)
7. Medical_Equipment (**Equipment_ID**, Equipment_Name, Equipment_Description, Maintance_Record)
8. Medication (**Medication_ID**, Medication_Name, Dosages, Patient_ID*, Doctor_ID*, Prescription_Date, Treatment_ID*)
9. Bills (**Bill_ID**, Patient_ID*, Total, Equipment_Cost, Medical_Cost, Treatment_Cost, Bill_Date, Department_ID*)
10. Treats (Patient_ID*, Treatment_ID*, Doctor_ID*)
11. Uses (Doctor_ID*, Equipment_ID*)
12. Doctor_Department (Doctor_ID*, Department_ID*)

Milestone: 3

VI. Implementation of Relational Model via MySQL:

Query 1: List doctors who have treated patients for multiple different diseases in a single appointment:

```
SELECT DISTINCT D.Doctor_ID, D.Doctor_Name
FROM Treats T
JOIN Diagnosis DS ON T.Patient_ID = DS.Patient_ID AND T.Doctor_ID = DS.Doctor_ID
JOIN Doctor D ON T.Doctor_ID = D.Doctor_ID
WHERE T.Treatment_ID IN (
   SELECT Treatment_ID
   FROM Diagnosis
   GROUP BY Treatment_ID
   HAVING COUNT(DISTINCT Disease) > 1
);
```

Output:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Doctor_ID	Doctor_Name
D003	Karina Reid
D009	Joseph Miller
D011	Nathan Bautista
D019	Robert Brown
D020	Ariana Bautista

- The query analyzes the relationships between doctors, patients, treatments, and diagnoses to find doctors who have treated patients with a variety of distinct diseases. This type of analysis can be useful for identifying doctors with broad expertise or experience in handling diverse medical conditions.

<u>Query 2:</u> Find the top 5 departments with the LOWEST average treatment cost:

SELECT D.Department_ID, D.Department_Name,

AVG(B.Treatment_Cost) **AS** Avg_Treatment_Cost

FROM Department D

JOIN Bills B **ON** D.Department_ID = B.Department_ID

GROUP BY D.Department_ID, D.Department_Name

ORDER BY Avg_Treatment_Cost **ASC**

LIMIT 5;

<u>Output:</u>

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

Department_ID	Department_Name	Avg_Treatment_Cost
Dpt01	Cardiology	960.700000
Dpt03	Orthopedics	983.217391
Dpt04	Pediatrics	1087.380952
Dpt05	Oncology	1150.235294
Dpt02	Neurology	1202.263158

- The query provides information about the average treatment cost for each department and presents the top 5 departments with the lowest average treatment costs. This type of analysis can be useful for identifying departments that are more cost-effective in terms of treatment expenses.

<u>Query 3:</u> Retrieve doctors who have not performed any treatments:

SELECT D.*

FROM Doctor D

LEFT JOIN Treats T **ON** D.Doctor_ID = T.Doctor_ID

WHERE T.Doctor_ID **IS NULL**;

Output:

	Doctor_ID	Doctor_Name	Speciality	Qualification	Doctor_Contact	Department_ID
▶	D007	Steven Cox	Pediatrics	MSc in Cardiology	(288)130-5986x58681	Dpt04

- This type of analysis can be useful for various purposes, such as identifying doctors who may be new to the system, doctors who are not currently actively treating patients, or for auditing purposes to ensure that all doctors have relevant treatment records.

Query 4: Comprehensive Patient Billing Report:

SELECT

b.Bill_ID,

p.Patient_Name,

b.Patient_ID,

SUM(b.Equipment_Cost) **AS** Equipment_Costs,

SUM(b.Medical_Cost) **AS** Medical_Costs,

SUM (b.Treatment_Cost) **AS** Treatment_Costs,

SUM (b.Total) **AS** Total_Bill

FROM

Bills b

JOIN

Patient p **ON** b.Patient_ID = p.Patient_ID

GROUP BY

b.Bill_ID, p.Patient_Name, b.Patient_ID;

Output:

Bill_ID	Patient_Name	Patient_ID	Equipment_Costs	Medical_Costs	Treatment_Costs	Total_Bill
B0017	Frederick Solis	P001	3409.00	2210.00	701.00	9427.00
B0026	Frederick Solis	P001	879.00	2046.00	1394.00	4324.00
B0045	Frederick Solis	P001	2825.00	1518.00	709.00	5402.00
B0071	Frederick Solis	P001	4264.00	2993.00	1791.00	9084.00
B0085	Frederick Solis	P001	1483.00	1674.00	1577.00	7529.00
B0033	Jacob Joseph	P003	3747.00	1709.00	1515.00	9029.00
B0037	Brooke Carson	P004	3139.00	2732.00	1004.00	5358.00
B0090	Jeffery Hughes	P005	2707.00	1136.00	249.00	5314.00
B0093	Jeffery Hughes	P005	3811.00	2255.00	462.00	1980.00
B0021	Margaret Wilson	P006	4433.00	1005.00	929.00	1810.00
B0056	Margaret Wilson	P006	3748.00	1546.00	1954.00	2504.00
B0010	Joseph Davis	P007	4578.00	1250.00	1371.00	3937.00

- The query provides a summary of bill information, breaking down the costs (equipment, medical, treatment, and total) for each bill, along with patient information. This type of analysis is useful for understanding the distribution of costs across different bills and associating them with specific patients.

<u>Query 5:</u> Patient Demographic Analysis for Targeted Healthcare Programs:

SELECT

YEAR(**CURRENT_DATE**) - **YEAR**(Patient_DOB) **AS** Age,

COUNT(*) **AS** Patient_Count,

d.Disease

FROM

Patient p

JOIN

Diagnosis d **ON** p.Patient_ID = d.Patient_ID

GROUP BY

Age, d.Disease;

Output:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Age	Patient_Count	Disease
17	2	Diabetes Type 2
40	1	Coronary Artery Disease
82	3	Hypertension
17	2	Osteoarthritis
52	1	Diabetes Type 2
34	1	Coronary Artery Disease
101	1	Asthma
60	1	Diabetes Type 2
17	1	Hypertension
101	1	Diabetes Type 2
22	1	Coronary Artery Disease
27	1	Asthma
97	2	Hypertension
81	1	Hypertension
48	1	Hypertension
95	1	Hypertension
82	2	Diabetes Type 2
20	2	Diabetes Type 2
70	1	Coronary Artery Disease
0	2	Coronary Artery Disease

Result 1

- The query's scope is to analyze and present data on the age distribution of patients diagnosed with different diseases. This type of analysis can be valuable for healthcare professionals, researchers, and policymakers to understand the epidemiology of diseases across different age cohorts and tailor healthcare strategies accordingly.

Query 6: Doctor Specialization and Patient Outcomes:

SELECT

dpt.Department_Name,

d.Speciality,

d.Doctor_Name,

COUNT(t.Treatment_ID) **AS** Number_of_Treatments,

AVG(b.Total) **AS** Average_Treatment_Cost

FROM

Doctor d

JOIN

Treatment t **ON** d.Doctor_ID = t.Doctor_ID

JOIN

Bills b **ON** t.Patient_ID = b.Patient_ID

JOIN

Department dpt **ON** d.Department_ID = dpt.Department_ID

GROUP BY

dpt.Department_Name, d.Speciality, d.Doctor_Name;

Output:

Department_Name	Speciality	Doctor_Name	Number_of_Treatments	Average_Treatment_Cost
Cardiology	Cardiology	Wayne Meyer	6	6482.666667
Cardiology	Cardiology	Karina Reid	3	5202.666667
Neurology	Neurology	Kenneth Ruiz	13	4332.538462
Neurology	Neurology	Ronald Carter	10	7220.200000
Neurology	Neurology	Ariana Bautista	10	5016.000000
Orthopedics	Orthopedics	James Martin	15	5459.400000
Orthopedics	Orthopedics	Joseph Miller	11	5787.272727
Orthopedics	Orthopedics	Sara Smith	20	5680.500000
Orthopedics	Orthopedics	Bridget Green	17	4468.235294
Orthopedics	Orthopedics	Edward Petersen	8	4834.250000
Orthopedics	Orthopedics	Brenda Brown	12	4128.833333
Pediatrics	Pediatrics	Garrett Frank	12	4351.250000
Pediatrics	Pediatrics	Steven Cox	4	8161.750000
Pediatrics	Pediatrics	Angel Castaneda	5	3941.800000
Pediatrics	Pediatrics	Nathan Bautista	6	5849.000000
Pediatrics	Pediatrics	Mr. Tyler Lucas	17	5101.470588
Oncology	Oncology	Joshua Harris	21	5281.857143
Oncology	Oncology	James Harris	9	3415.888889
Oncology	Oncology	Robert Brown	5	5053.200000

- The analytical scope of the query is to provide insights into the number of treatments and the average treatment cost associated with each doctor within different departments. This type of analysis can help in assessing the workload and cost-effectiveness of doctors in various specialties across different departments within a healthcare system. It also allows for comparisons and assessments related to treatment patterns and costs within the context of departmental and specialty differences.

Query 7: Appointment Scheduling Efficiency:

SELECT

Doctor_ID,

COUNT(Appointment_ID) **AS** Total_Appointments,

AVG(**TIMESTAMPDIFF**(**MINUTE**, Appointment_Time,

Next_Appointment_Time)) **AS** Average_Wait_Time

FROM

(**SELECT**

a.Doctor_ID,

a.Appointment_ID,

a.Appointment_Time,

LEAD(a.Appointment_Time) **OVER** (**PARTITION BY** a.Doctor_ID

ORDER BY a.Appointment_Time) **AS** Next_Appointment_Time

FROM

Appointment a) **AS** Appointments

GROUP BY

Doctor_ID;

Output:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Doctor_ID	Total_Appointments	Average_Wait_Time
D001	3	150.0000
D002	5	75.0000
D003	3	30.0000
D004	5	75.0000
D005	8	60.0000
D006	3	30.0000
D007	7	80.0000
D008	3	210.0000
D009	9	52.5000
D010	8	60.0000
D011	7	80.0000

- Based on their appointment calendars, the analytical scope of the query is to provide insights into the workload and typical wait times for doctors. It assists in determining how busy each physician is and how long people often must wait between appointments with the same physician. The patient's experience can be enhanced, and scheduling procedures can be optimized with the use of this kind of analysis.

<u>Query 8:</u> Analysis of Disease Trends Over Time:

```
WITH MonthlyDiagnoses AS (
    SELECT
        YEAR(Diagnosis_Date) AS Year,
        MONTH(Diagnosis_Date) AS Month,
        Disease,
        COUNT(*) AS Number_of_Diagnoses,
        ROW_NUMBER() OVER (PARTITION BY YEAR(Diagnosis_Date),
MONTH(Diagnosis_Date) ORDER BY COUNT(*) DESC) AS
Disease_Rank
    FROM
        Diagnosis
    GROUP BY
        Year, Month, Disease
)
SELECT
    Year,
    Month,
    Disease,
    Number_of_Diagnoses
FROM
    MonthlyDiagnoses
WHERE
    Disease_Rank = 1;
```

Output

Year	Month	Disease	Number_of_Diagnoses
2023	1	Diabetes Type 2	4
2023	2	Coronary Artery Disease	3
2023	3	Diabetes Type 2	4
2023	4	Hypertension	4
2023	5	Diabetes Type 2	5
2023	6	Hypertension	2
2023	7	Osteoarthritis	3
2023	8	Asthma	3
2023	9	Coronary Artery Disease	2
2023	10	Coronary Artery Disease	5
2023	11	Diabetes Type 2	2

- Finding the ailment with the greatest number of diagnoses each month is the analytical goal of the query. The final query filters include only the top-ranked disease for each month. The ROW_NUMBER window function is used to help rank diseases within each month based on the number of diagnoses. This kind of study can be helpful in figuring out trends in monthly diagnoses as well as comprehending how common diseases change over time.

Milestone: 4

VII. NoSQL Implementation:

Query:1

```
([
   { $group: { _id: "$Doctor_ID", numberOfAppointments: { $sum: 1 } } },
   { $lookup: {
      from: "Doctor_Department",
      localField: "_id",
      foreignField: "Doctor_ID",
      as: "departmentData"
   }},
   { $unwind: "$departmentData" },
   { $group: { _id: "$departmentData.Department_ID", totalAppointments: { $sum:
"$numberOfAppointments" } } }
])
```

Output:

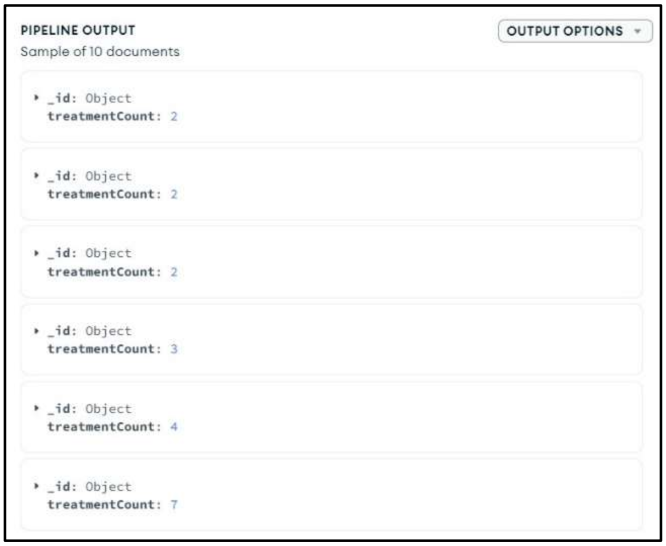
- In summary, the analytical scope of this query is to find the total number of appointments grouped by the "Department_ID" in the "Doctor_Department" collection. It involves aggregating data from the original collection based on the "Doctor_ID," joining it with the "Doctor_Department" collection, and then grouping the results by department to calculate the total number of appointments for each department.

Query:2

```
[
 {
  $lookup: {
   from: "Diagnosis",
   localField: "Treatment_ID",
   foreignField: "Treatment_ID",
   as: "diagnosisData",
  },
 },
 {
  $unwind: {
   path: "$diagnosisData",
   preserveNullAndEmptyArrays: true,
  },
 },
 {
  $group: {
   _id: {
    Patient_ID: "$Patient_ID",
    Disease: "$diagnosisData.Disease",
   },
   treatmentCount: { $sum: 1 },
  },
 },
```

```
  { $match: { treatmentCount: { $gt: 1 } } },
]
```

Output:



- In summary, the analytical scope of this query is to find patients who have received multiple treatments for the same disease. It involves joining the current collection with the "Diagnosis" collection based on the "Treatment_ID," grouping the results by patient and disease, and then filtering to include only those cases where the patient has received more than one treatment for a specific disease.

Query:3

```
[
  {
    $lookup: {
      from: "Uses",
      localField: "Equipment_ID",
      foreignField: "Equipment_ID",
      as: "equipmentUsage",
    },
```

```
  },
  { $unwind: "$equipmentUsage" },
  {
    $group: {
      _id: "$equipmentUsage.Equipment_ID",
      totalUsage: { $sum: 1 },
    },
  },
]
```

Output:

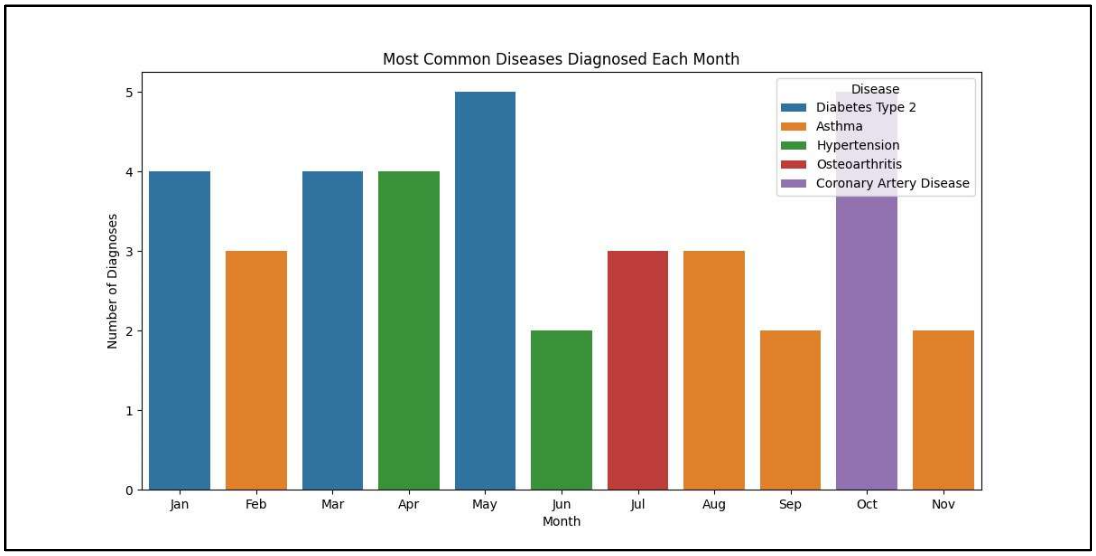
- In summary, the analytical scope of this query is to find the total usage of each equipment, aggregated from the current collection and the "Uses" collection. It involves joining the current collection with the "Uses" collection based on the "Equipment_ID," unwinding the resulting array, and then grouping the results by equipment to calculate the total usage for each equipment.

VIII. Database Access via Python

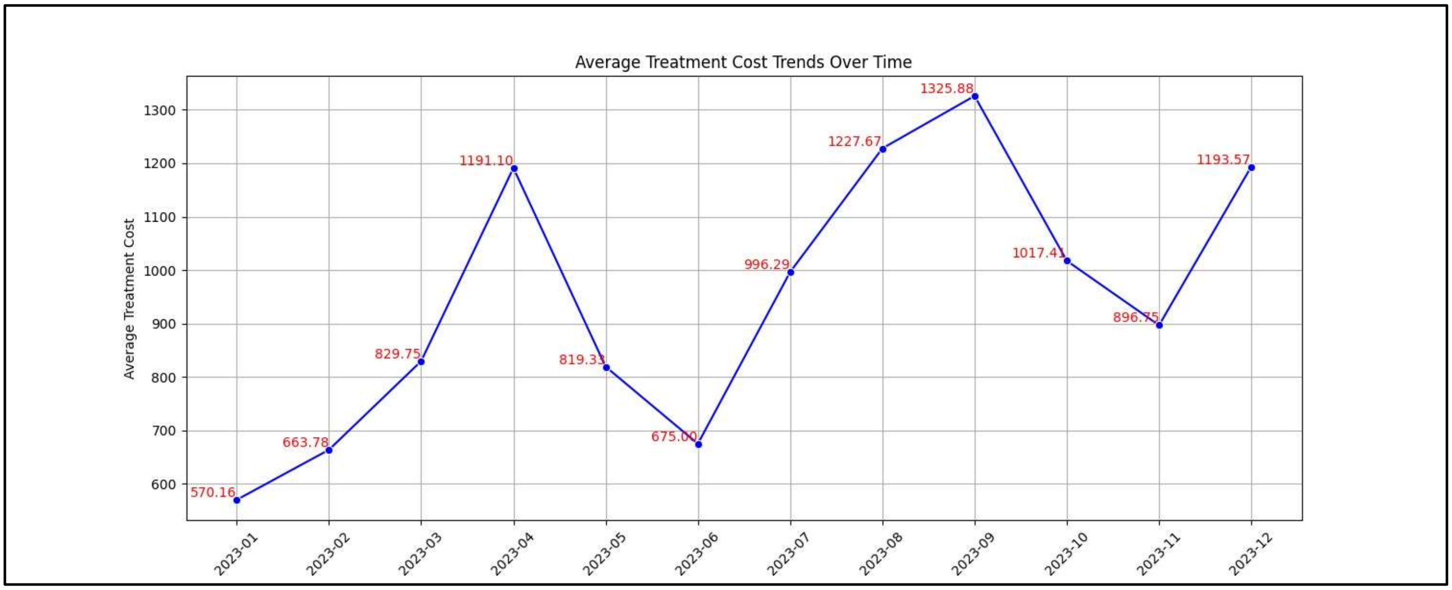
Python is employed to access the database and perform data analysis and visualization. The connectivity between MySQL and Python relies on MySQL.connector and cursor. Queries are executed and results retrieved, followed by transforming the obtained lists into a data frame using the pandas library. Matplotlib is then utilized to create graphical representations for the analytics.

.

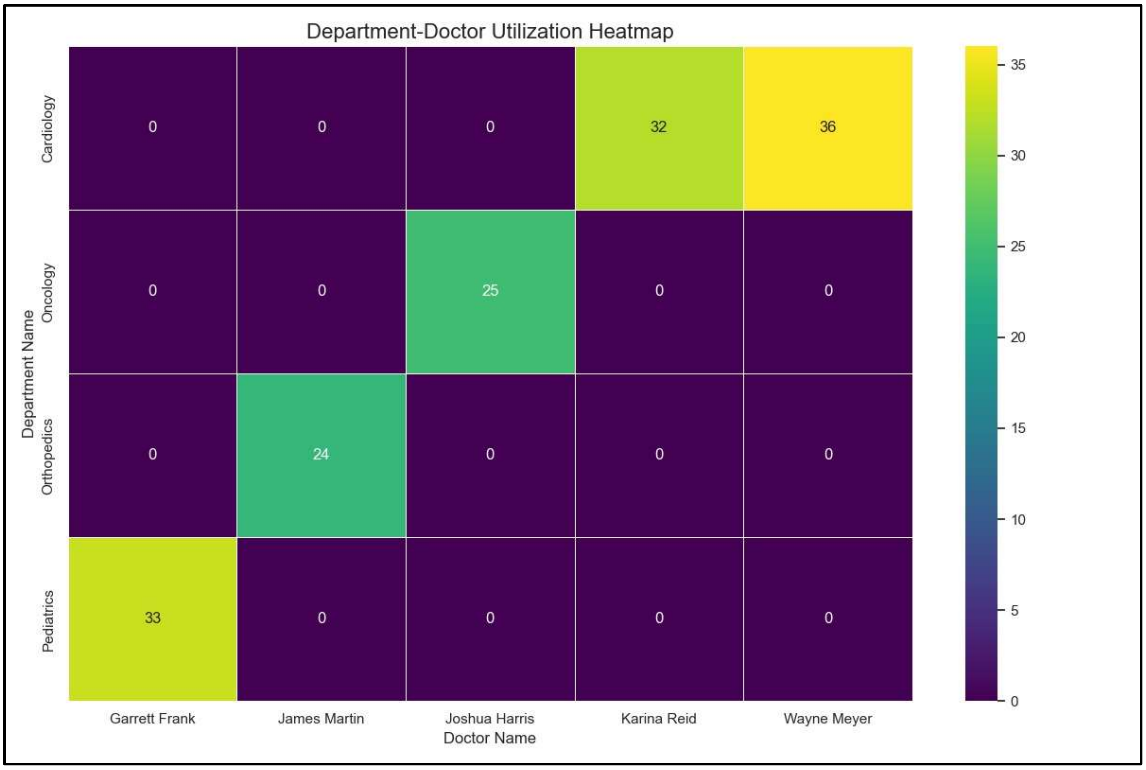
1. Distribution of Diseases Diagnosed Over Time:



2. Average Treatment Cost of Asthma:



3. Workload Distribution:



IX. Summary:

The project yielded significant outcomes across multiple fronts. Firstly, by analyzing departmental loads, particularly focusing on high loads within the medical staff, we were able to pinpoint areas needing resource redistribution. This led to a more balanced system, reducing strain on specific individuals or departments and improving overall system efficiency.

Secondly, our analysis delved into treatment costs and disease patterns, enabling us to strategize for minimizing treatment expenses, especially for prevalent diseases. This approach not only benefits patients by reducing their healthcare expenses but also optimizes the hospital's resource utilization, fostering an environmentally conscious and cost-effective system.

Thirdly, the data model developed in this project provides invaluable insights for decision-making. It enables us to forecast disease trends and predict departmental loads, empowering us to make informed decisions regarding budget allocations for departments and necessary equipment. This foresight supports the hospital in making proactive decisions, enhancing operational efficiency, and ensuring optimal resource utilization for improved patient care and system sustainability.