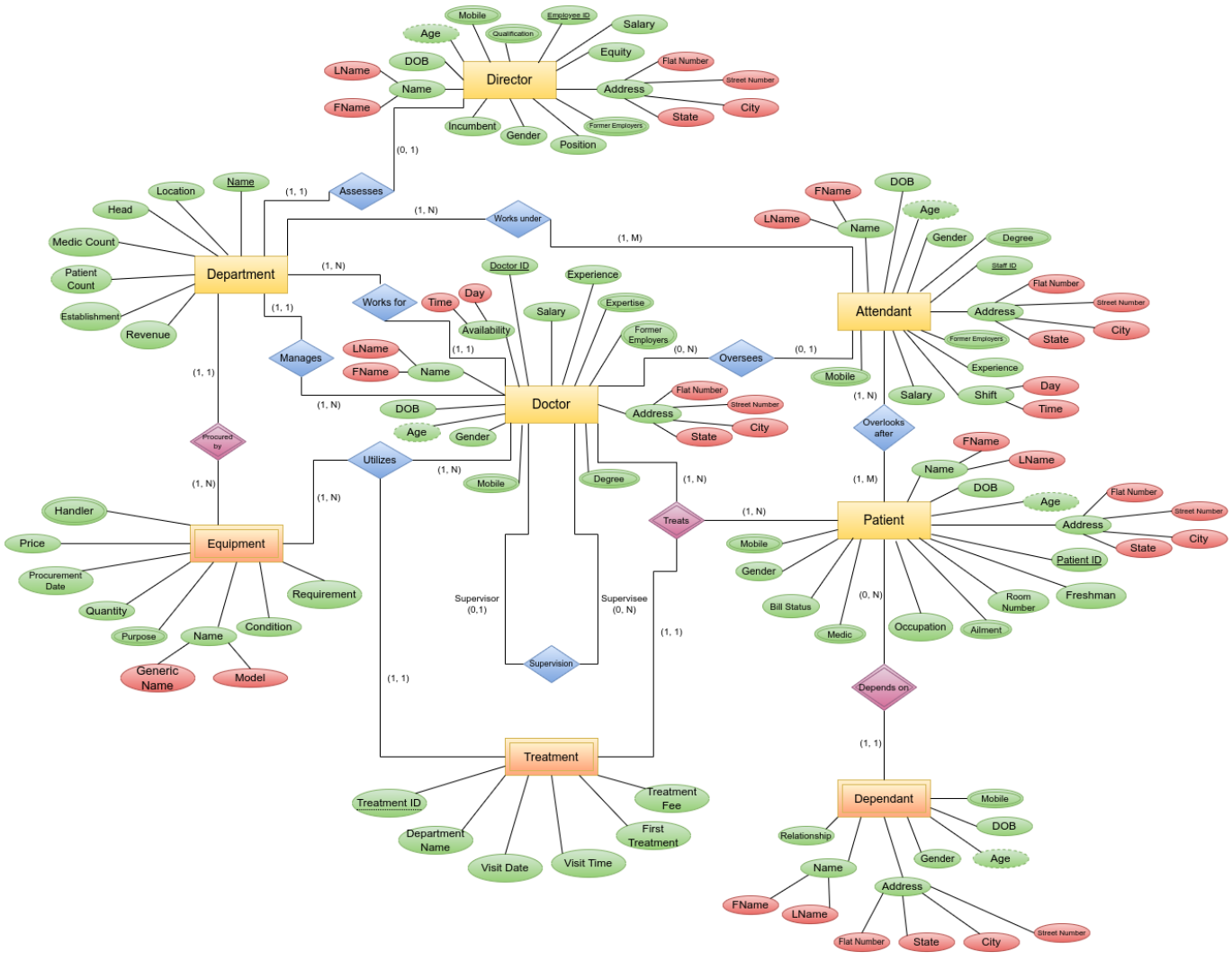
CS4.301 Data and Applications - Project Phase 3

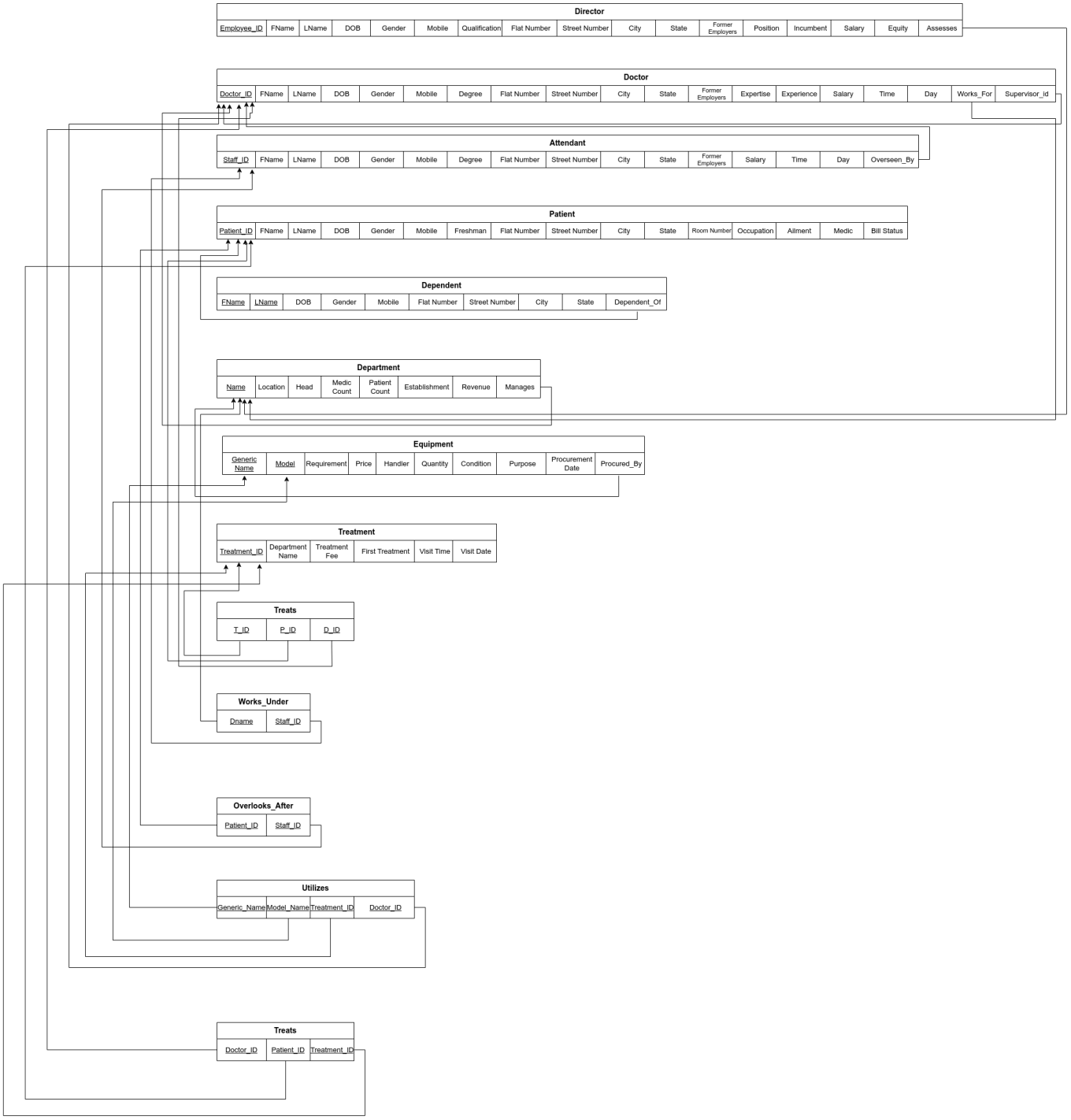
Gargie Tambe - 2022111008
Chetan Mahipal - 2022101096
Yash Shinde - 2022111009
Abhinav Raundhal - 2022101089

November 18, 2023

ER Diagram:



Relational Model:



Explanation

A relational diagram serves as a visual representation of the interconnections among tables in a relational database. In this context, a relational database organizes data into tables, with each table comprising rows and columns. The relationships between these tables are established through shared fields, commonly known as keys, which encompass primary and foreign keys. These keys act as bridges, linking data in one table to the corresponding data in another.

Hospital Management System Relational Model

In the relational model designed for the hospital management system, each entity from the original ER diagram has been translated into a corresponding table along with its associated attributes. To address relationships with m:n cardinality ratios and ternary relationships, we have introduced new tables. Additionally, attributes from various tables have been amalgamated as new attributes within the respective entity tables. To maintain referential integrity, foreign keys and primary keys from different tables have been meticulously mapped, visually represented by arrows pointing from foreign keys (FK) to primary keys (PK).

Changes Made in the Hospital Management System:

The process of translating the ER model into the relational model involved several key changes to enhance data organization and reduce redundancy.

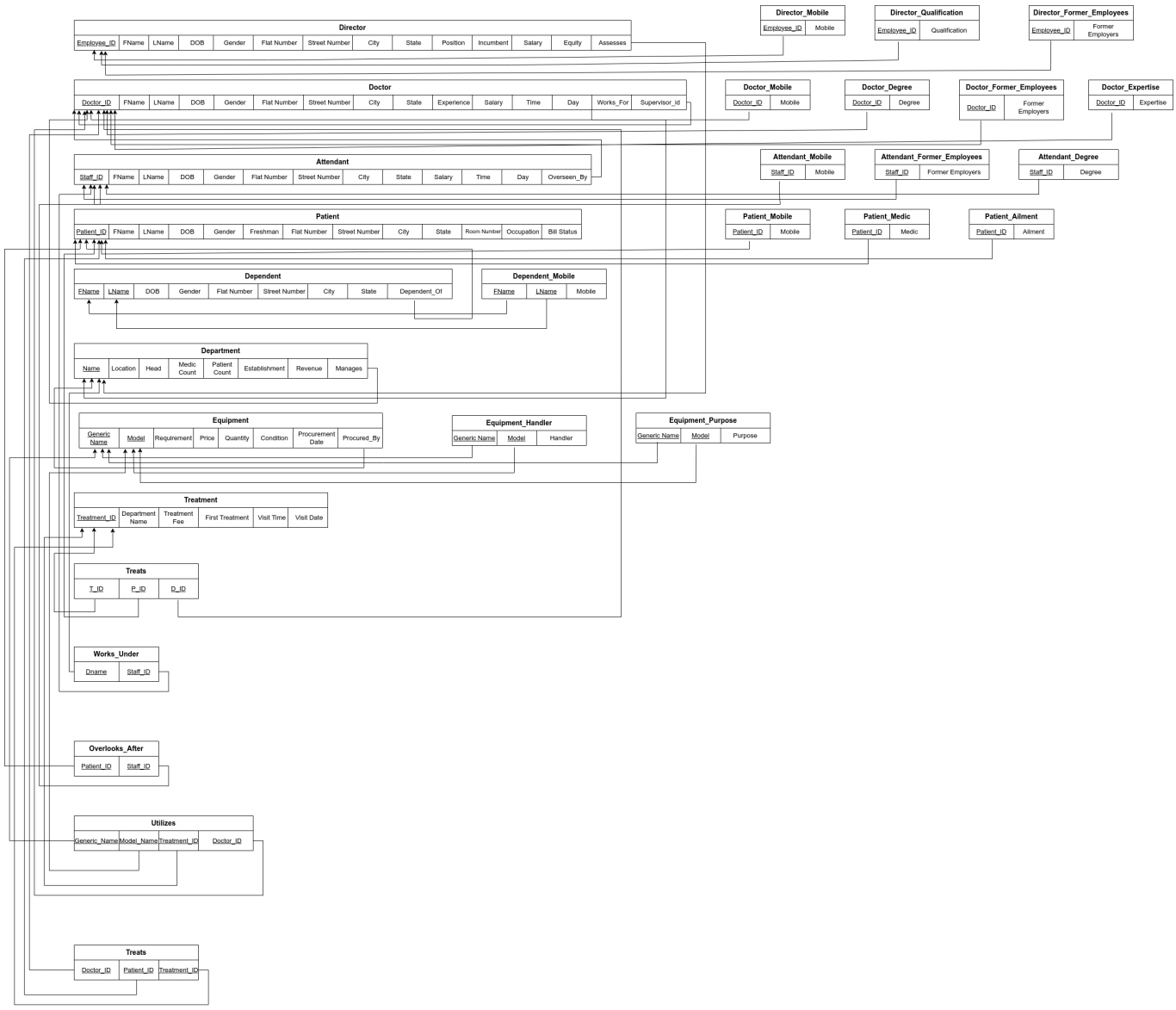
1. New Tables for Complex Relationships: Tables representing m:n cardinality ratios and ternary relationships were introduced to maintain a clear and normalized structure.

2. Attribute Merging: Attributes from different tables were merged into entity tables to simplify data retrieval and improve overall database performance.

3. Foreign Key Mapping: Foreign keys and primary keys were meticulously mapped to establish relationships between tables, ensuring data consistency and integrity.

These changes are aimed at creating a relational model that not only accurately represents the hospital management system but also adheres to best practices in database design.

1NF form:



Explanation

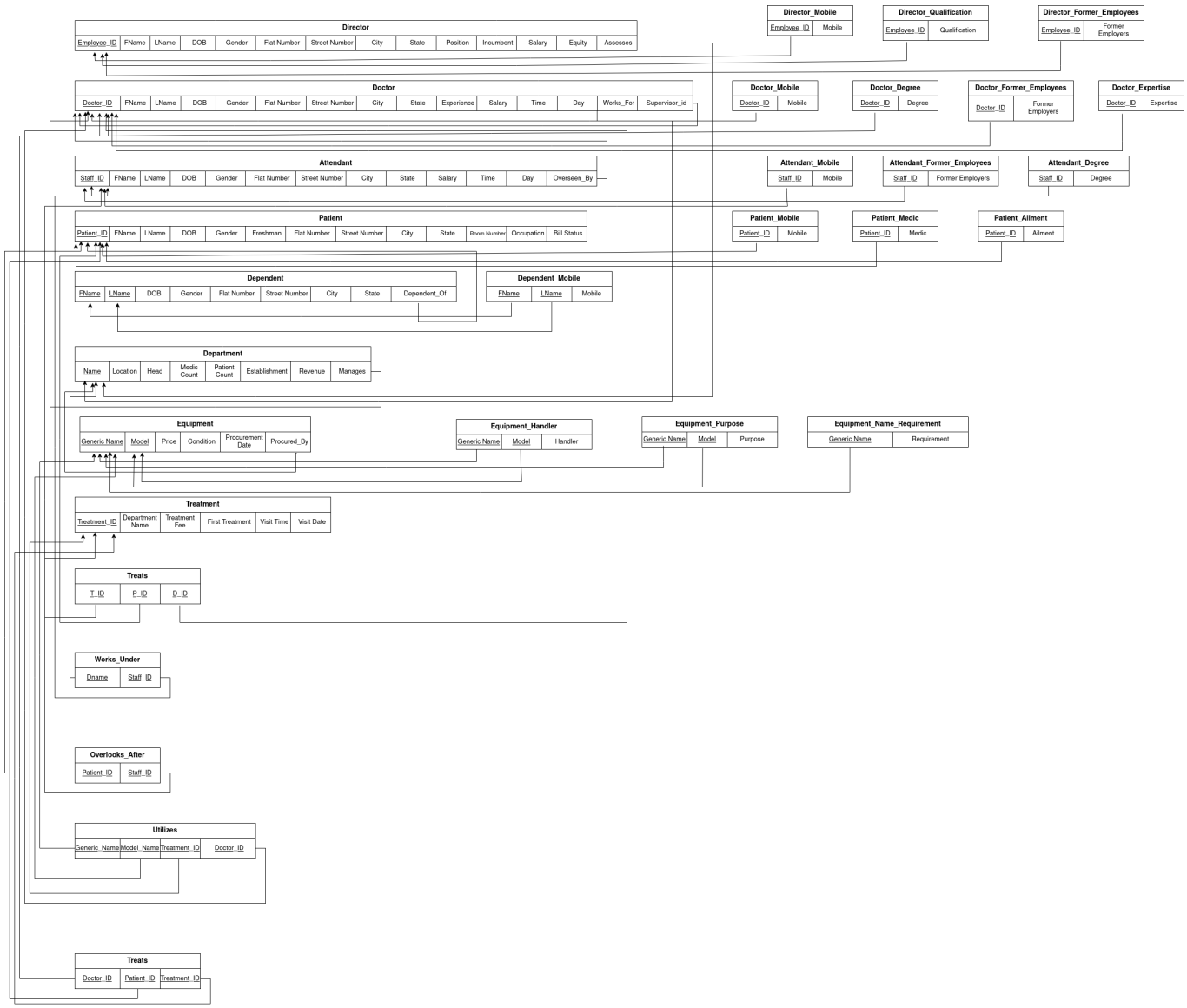
In the context of achieving the First Normal Form (1NF), the objective is to ensure that each table's entries are atomic, meaning they cannot be further divided. This necessitates that every column in a table contains only simple, indivisible values, thereby eliminating the presence of multi-valued or composite attributes. The primary goal of attaining 1NF is to reduce redundancy and enhance data reliability by structuring tables in a way that avoids repeating groups or complex data types.

Changes Made in the Hospital Management System

In the 1NF form for the hospital management system, several changes have been implemented to address multi-valued attributes. These attributes have been relocated to new tables, where the key is the same as the primary key of the original table (now acting as a foreign key). The relationship between the new and original tables is represented by an arrow from the foreign key to the primary key.

As an example, the Degrees that a Doctor holds, considered a multi-valued attribute, has been moved to a new table named Doctor_Degree. In this table, Doctor_ID serves as the foreign key and is linked to the Doctor table, where Doctor_ID functions as the primary key. This restructuring enhances the database's adherence to 1NF principles and contributes to a more streamlined and reliable data organization.

2NF Form:



Explanation

In the process of achieving the second Normal Form (2NF) for the hospital management system, our focus is on eliminating partial dependencies within relations with composite keys. 2NF is crucial for ensuring that each non-prime attribute (attributes not part of any candidate key) is fully functionally dependent on the entire composite primary key.

Changes Made in the Hospital Management System

To adhere to the principles of 2NF, we made specific changes in the hospital management system's database structure. Notably, in the Equipment table, which utilizes composite keys (Generic_Name and Model), we observed that the attribute 'Requirement' is dependent only on 'Generic_Name.' To address this partial dependency, we introduced a new table named 'Equipment_Name_Requirement.' This modification ensures that each attribute is fully dependent on the composite primary key, aligning the system with the requirements of the second Normal Form.

3NF Form:

Explanation:

Normalization is an essential process in database design to enhance data integrity and minimize redundancy. While 2NF addresses partial dependencies, 3NF goes further to eliminate transitive dependencies. In a relational database, transitive dependencies occur when an attribute is dependent on another attribute, which, in turn, is dependent on a third attribute. This can lead to update anomalies, impacting the reliability of the system.

In 3NF, we organize the data to ensure that for any functional dependencies like $X \rightarrow Y$ and $Y \rightarrow Z$, Z is neither partially nor transitively dependent on X. This organizational structure creates a more refined and stable database, reducing the risk of update anomalies and improving overall system reliability.

Changes Made in the Hospital Management System:

To achieve the third normal form (3NF) in our Hospital Management System, we carefully examined the existing relational model after the 2NF conversion. Since our 2NF already satisfies all the conditions of 3NF, there was no need to create a separate 3NF diagram. The thorough normalization process undertaken during the transition from ER to 1NF and subsequently to 2NF ensured that the database structure is free from transitive dependencies, thereby eliminating the possibility of update anomalies.

In the 2NF diagram, we addressed partial dependencies by restructuring tables and in doing so, we inadvertently eliminated transitive dependencies as well. As a result, the database is now in compliance with both 2NF and 3NF, providing a robust and reliable foundation for the Hospital Management System.