Topic : Hospital Management System

Team No : 1

Team Member Names:

Sujal Anil Dusane

Priyanka Basavaraj Bhadrappanavar

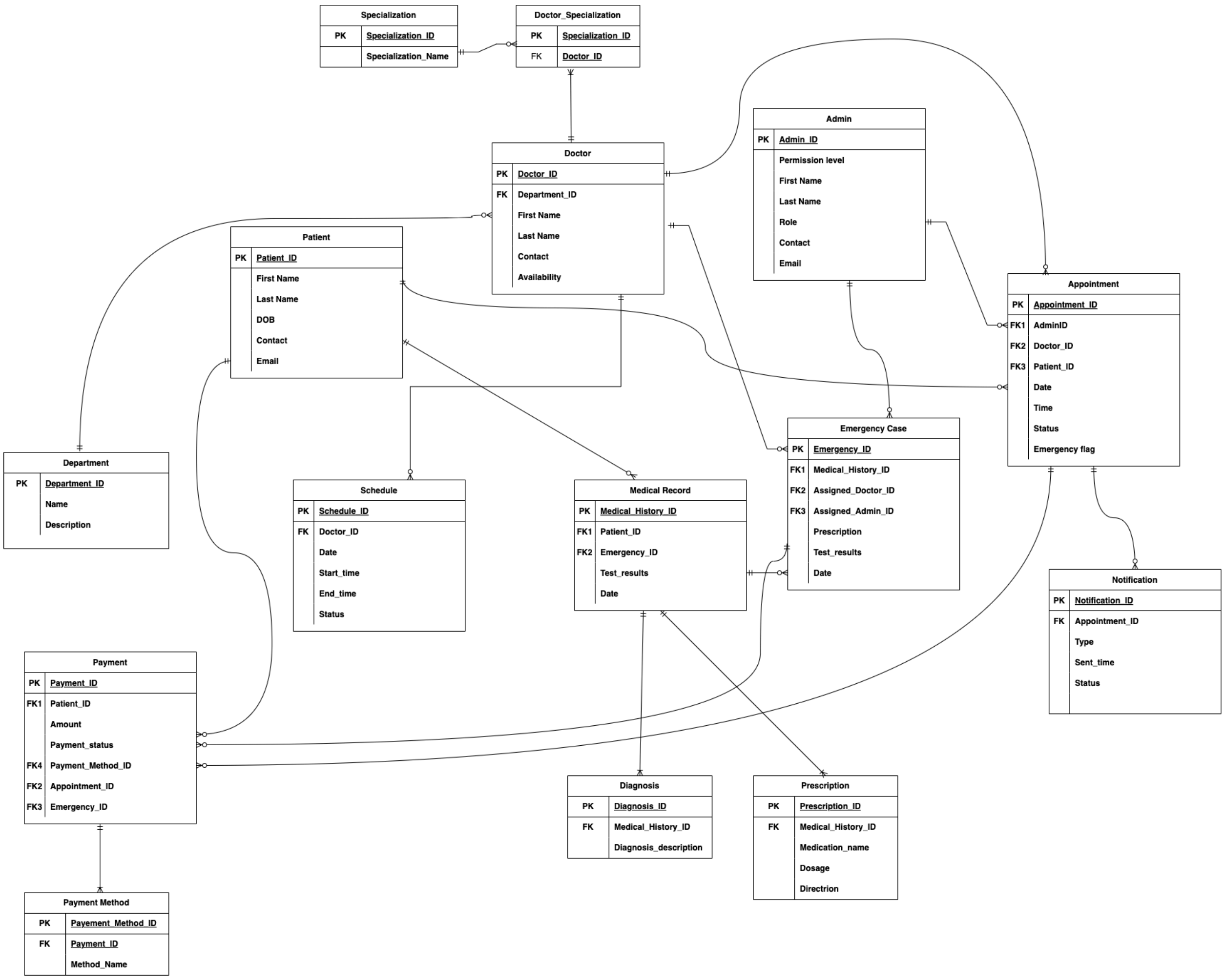
Joevita Faustina Doss Felix Anthony Raj

Vindhya Vasini Kaku

Yung-Rou Ko

Explanation of Improvements Made to the Logical ERD

The enhancements made to the Hospital Appointment Management System's Logical Entity-Relationship Diagram (ERD) are explained in depth in this paper. In order to improve normalization, remove duplications, and guarantee adherence to third normal form (3NF), these improvements are based on input and additional analysis. A summary of the particular adjustments and the justification for each enhancement may be found below.



Key Changes Made

1. Admin ID Updated

- **Change**: The Admin_ID attribute was updated in the Admin entity to ensure consistency and clarity in identifying administrative users.
- **Reason**: This ensures that each admin is uniquely identifiable and aligns with the naming conventions used across other entities.

2. Doctor-to-Department Relationship Corrected

- **Change**: The relationship between Doctor and Department was updated so that each doctor belongs to a single department, but a department can have multiple doctors.
- **Reason**: This correction reflects real-world scenarios where a doctor specializes in one department, but departments can have many doctors. It also eliminates any potential ambiguity in the relationship.

3. Emergency Linked to Medical History

- **Change**: A direct relationship was added between Emergency Case and Medical Record.
- **Reason**: Emergency cases often result in updates to a patient's medical history (e.g., new diagnoses or prescriptions). This link ensures that emergency-related medical data is properly recorded and accessible.

4. Emergency Payments Added

- **Change**: The Payment entity now includes a foreign key (Emergency_ID) to handle payments for emergency cases.
- **Reason**: Emergency cases often require immediate payment processing, which was missing in the previous design. This addition ensures that payments for emergencies are tracked separately from regular appointments.

5. Day Attribute Changed to Date

- **Change**: The Day attribute in the Schedule entity was renamed to Date.
- **Reason**: Using Date provides more precise information about the schedule and avoids confusion, especially when dealing with specific days in a calendar format.

Composite Attributes Broken Down

1. Names Split into First Name and Last Name

- **Change**: Composite attributes like Name (in entities such as Patient, Doctor, and Admin) were split into First Name and Last Name.
- **Reason**: This adheres to 3NF by ensuring atomicity of attributes, making it easier to query and manage individual name components.

2. Payment Method Normalized

- **Change**: A new table, Payment Method, was introduced with attributes Payment_Method_ID and Method_Name.
- **Reason**: Payment methods were previously stored as repetitive values in the Payment table. By creating a separate table, redundancy is eliminated, and scalability is improved (e.g., adding new payment methods).

3. Diagnosis Normalized

- **Change**: The relationship between Medical Record and diagnoses was updated so that a medical history can have multiple diagnoses via the new Diagnosis table.
- **Reason**: Patients may receive multiple diagnoses during a single visit or over time. This normalization ensures accurate representation of all diagnoses without duplication.

4. Prescription Normalized

- **Change**: Prescriptions were moved into their own table, allowing for multiple prescriptions per medical history record.
- **Reason**: Similar to diagnoses, patients may receive multiple prescriptions during treatment. This change ensures proper recording of each prescription while maintaining normalization.

5. Specialization Normalized

- **Change**: A many-to-many relationship between doctors and specializations was resolved using a junction table (Doctor_Specialization) linking the Doctor and Specialization entities.
- **Reason**: Doctors often have multiple specializations, and this structure avoids redundancy while maintaining flexibility for future updates.

Summary of Benefits

1. **Improved Data Integrity**
    - Breaking down composite attributes into atomic fields ensures compliance with 3NF.
    - Normalization reduces redundancy across all entities.

2. **Enhanced Scalability**
    - The addition of new entities like Payment Method, Diagnosis, and Prescription allows for future expansion without disrupting existing data structures.

3. **Better Emergency Case Handling**
    - Linking emergency cases to medical records and payments ensures that urgent situations are managed comprehensively within the system.

4. **Streamlined Payment Processing**

- Payments for both appointments and emergencies are now handled seamlessly through proper foreign key relationships.

5. **Clarity in Relationships**
   - Correcting relationships (e.g., doctor-to-department) ensures that real-world scenarios are accurately represented in the database design.

6. **Improved Query Efficiency**
   - Splitting attributes like names into first/last names makes it easier to search, sort, or filter data based on specific components.

Conclusion

The updated Logical ERD reflects significant improvements that address feedback from earlier iterations while adhering to best practices in database design. These changes enhance data integrity, scalability, and functionality, ensuring that the Hospital Appointment Management System meets its mission objectives effectively.