Data Management and Database Design
P2: Database Design, Initial ERD

Group 20:

- **Pritesh Nimje – 002817324**
- **Kamal Kotgire - 002879990**
- **Neha Manjunath – 002879249**
- **Chengjia Feng - 002746829**
- **Yashvardhan Limbodiya – 002284031**

Community Management System

Business Problems Addressed:

- **Inefficient Resident Management:**

  Without a centralized database, managing resident information, such as contact details, lease agreements, and payment histories, becomes cumbersome and prone to errors. This can lead to delays in communication, inaccuracies in billing, and difficulty in tracking resident-related issues.

- **Poor Service Request Handling:**

  In the absence of a structured system, service requests from residents may get lost or overlooked, resulting in delayed responses and dissatisfaction among residents. This can impact the overall reputation of the community and lead to decreased resident satisfaction.

- **Security and Access Control Challenges:**

  Manual tracking of visitor and vehicle movements makes it challenging to maintain security and control access to the community premises. Without proper documentation and monitoring, unauthorized entry or security breaches may occur, compromising the safety of residents and their property.

- **Amenity Mismanagement:**

  Without a system to manage amenity bookings and usage, there may be instances of conflicts, double bookings, or misuse of community facilities. This can lead to frustration among residents and inefficient utilization of shared resources.

- **Lack of Incident Tracking and Reporting:**

  In the absence of a structured logging system, security incidents within the community may go undocumented or unaddressed. This hampers the ability to track patterns, identify potential risks, and take proactive measures to enhance security and compliance.

Entities:

- **Resident:**

  Members of the community with rights and responsibilities, including financial contributions and lodging service requests.

- **Apartment:**

  Units of residence within buildings, linked to residents and critical for defining living space allocation.

- **Building:**

  Architectural entities containing apartments, forming the community's physical and locational framework.

- **Staff:**

  Personnel managing daily operations, servicing requests, and maintaining community welfare and order.

- **Visitor:**

  Guests recorded upon entry for security purposes and community access management.

- **Vehicle:**

  Transportation means associated with residents and visitors, integral to parking allocation and security strategies.

- **ServiceRequest:**

  Initiated by residents for maintenance actions, these requests are addressed and resolved by the staff.

- **Payment:**

  Reflects residents' financial engagements with the community, encompassing various fees and charges.

- **Amenity:**

  Shared facilities within the community, available to residents under certain terms and conditions.

- **AmenityBooking:**

  Scheduling entity for amenities, documenting residents' reservations and utilization of shared spaces.

- **SecurityLog:**

Documentation of security events and measures, vital for safety oversight and community standards enforcement.

Relationships:

- **Resident-Apartment:**

  A resident may occupy multiple apartments, while each apartment is designated for at least one resident. This suggests a many-to-one relationship from Resident to Apartment.

- **Apartment-Building:**

  Each apartment is situated within a single building, and a building encompasses multiple apartments, reflecting a many-to-one relationship from Apartment to Building.

- **Resident-Payment:**

  A resident may conduct numerous financial transactions, but each payment record is uniquely linked to one resident, indicating a one-to-many relationship from Resident to Payment.

- **Payment-Apartment:**

  Payments made by residents correspond to specific apartments, representing a one-to-many relationship where each payment is associated with one apartment.

- **Resident-ServiceRequest:**

  Residents may raise various service requests; however, each service request is associated with only one resident, establishing a one-to-many relationship from Resident to ServiceRequest.

- **Staff-ServiceRequest:**

  Staff members are tasked with addressing multiple service requests, but each service request is handled by only one staff member, creating a one-to-many relationship from Staff to ServiceRequest.

- **Resident-Visitor:**

  A resident can host several visitors, whereas each visitor registers their visit with one resident at a time, forming a one-to-many relationship from Resident to Visitor.

- **Resident-Vehicle:**

  A resident may possess multiple vehicles, yet each vehicle is registered to a singular resident, denoting a one-to-many relationship from Resident to Vehicle.

- **Visitor-Vehicle:**

  A visitor may arrive with only one vehicle per visit, and each vehicle brought by a visitor is associated with that visit, signifying a one-to-one relationship from Visitor to Vehicle.

- **Resident-AmenityBooking:**

A resident can schedule numerous bookings for amenities, while each booking is arranged by an individual resident, resulting in a one-to-many relationship from Resident to AmenityBooking.

- **Amenity-AmenityBooking:**

  An amenity can be booked for multiple time slots, but each booking pertains to a single amenity, showing a one-to-many relationship from Amenity to AmenityBooking.

- **Resident-SecurityLog:**

  Residents may be involved in or report security incidents, creating a many-to-many relationship between residents and security logs.

- **Staff-SecurityLog:**

  A staff member may generate various entries in the security log, with the potential for multiple logs to be reported by a single staff member, establishing a one-to-many relationship from Staff to SecurityLog. Additionally, staff members may resolve multiple security incidents, and a security log may be resolved by a single staff member, indicating a one-to-many relationship from Staff to SecurityLog.

Key Design Decisions:

- **Entity Selection:**

  The entities included in the database were carefully chosen based on their significance in managing the residential community's operations effectively.

- **Resident, Apartment, and Payment:**

  Entities form the core of the database, facilitating the management of resident data, apartment allocations, and financial transactions crucial for maintaining smooth operations within the complex.

- **ServiceRequest and Staff:**

  Incorporating these entities allows for the efficient handling of service requests raised by residents and the assignment of staff members responsible for addressing these requests, ensuring timely resolution and resident satisfaction.

- **Visitor and Vehicle:**

  By including entities for visitors and vehicles, the database can track and manage access to the community, enhancing security measures and providing residents with peace of mind.

- **Amenity and AmenityBooking:**

  These entities enable the scheduling and utilization of community amenities, optimizing their usage and ensuring fair access for all residents.

- SecurityLog:

The inclusion of the SecurityLog entity allows for the recording and monitoring of security incidents within the community, contributing to the overall safety and well-being of residents.

Initial E-R Diagram: