Database Management Systems

Winter 2026

PropertyDekho.in

A Centralized Real Estate Management System

Project Assignment 2 Report

E-R Modeling & Relational Schema Design

Team Members:

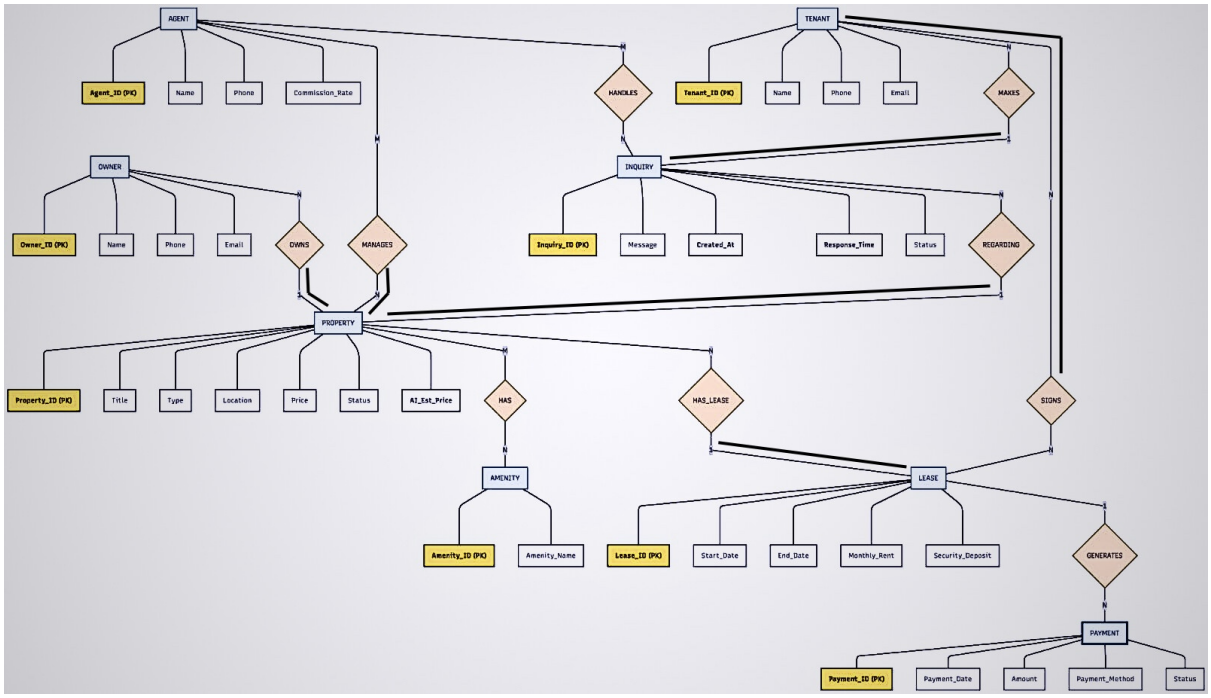
Shashank Kumar – 2024521

Namish Batra – 2024369

Mehardeep Singh Bhalla – 2024346

ENTITY-RELATIONSHIP (E-R) DIAGRAM

The following E-R diagram represents the conceptual design of the *PropertyDekho* system.



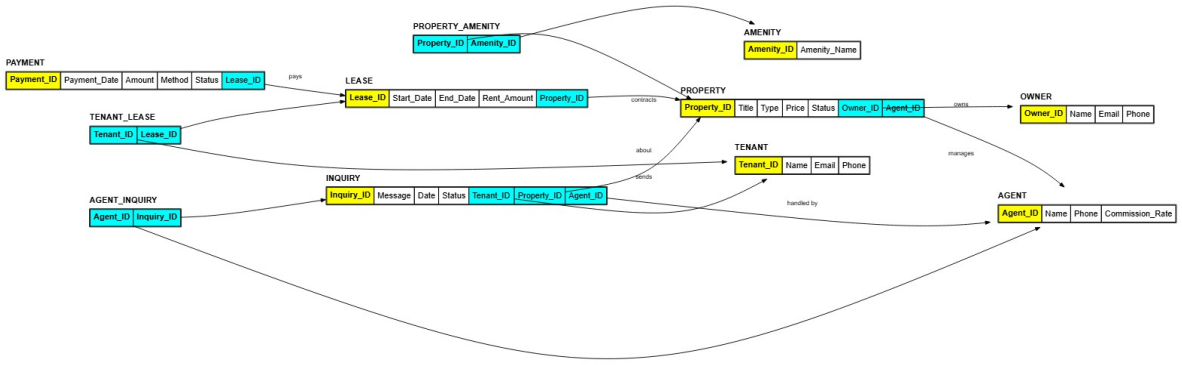
Note on Cardinality: The cardinality ratios (1, N) are placed near the target entity. For example, the 'N' near Property in the Owner-Property relationship indicates that one Owner can have N Properties.

E-R Constraints

Double lines in the diagram indicate Total Participation (min=1).

- **Agent & Property:** Agent (0, N) — Property (1, 1)
- **Owner & Property:** Owner (0, N) — Property (1, 1)
- **Property & Amenity:** Property (1, N) — Amenity (1, N)
- **Property & Inquiry:** Property (1, N) — Inquiry (1, 1)
- **Tenant & Inquiry:** Tenant (1, N) — Inquiry (1, 1)
- **Property & Lease:** Property (1, N) — Lease (1, 1)
- **Tenant & Lease:** Tenant (1, N) — Lease (1, 1)
- **Lease & Payment:** Lease (1, N) — Payment (1, 1)

RELATIONAL SCHEMA DESIGN



Business Rules

Primary Keys (PK) are underlined. Foreign Keys (FK) are marked with (*). Additional domain constraints (CHECK, UNIQUE) are specified below each table.

1. **AGENT** (Agent_ID, Name, Phone, Commission_Rate)

- **CHECK:** `Commission_Rate BETWEEN 0 AND 25` (Realistic percentage cap).
- **UNIQUE:** `Phone` must be unique for every agent.

2. **OWNER** (Owner_ID, Name, Phone, Email)

- **UNIQUE:** `Email` and `Phone` must be unique.

3. **TENANT** (Tenant_ID, Name, Phone, Email)

- **UNIQUE:** `Email` and `Phone` must be unique.

4. **AMENITY** (Amenity_ID, Amenity_Name)

5. **PROPERTY**
(Property_ID, Title, Type, Location, Price, Status, AI_Est_Price, Owner_ID*, Agent_ID*)

- FK `Owner_ID` references `OWNER` (NOT NULL).
- FK `Agent_ID` references `AGENT` (NOT NULL).
- **CHECK:** `Price > 0` (Price cannot be negative or zero).
- **CHECK:** `Status IN ('Available', 'Sold', 'Rented', 'Pending')`.

6. **PROPERTY_AMENITY** (Property_ID*, Amenity_ID*)

- Composite PK maps M:N relationship.

7. **INQUIRY**
(Inquiry_ID, Message, Date, Status, Tenant_ID*, Property_ID*, Agent_ID*)

- FKs reference `TENANT`, `PROPERTY`, and `AGENT` (All NOT NULL).
- **CHECK:** `Status IN ('New', 'Responded', 'Closed')`.

8. **AGENT_INQUIRY** (Agent_ID*, Inquiry_ID*)

9. LEASE
(Lease_ID, Start_Date, End_Date, Monthly_Rent, Security_Deposit, Property_ID*)

- FK `Property_ID` references `PROPERTY` (NOT NULL).
- **CHECK:** `End_Date > Start_Date` (Logical validity).
- **CHECK:** `Monthly_Rent > 0`.

10. TENANT_LEASE (Tenant_ID*, Lease_ID*)

11. PAYMENT
(Payment_ID, Payment_Date, Amount, Method, Status, Lease_ID*)

- FK `Lease_ID` references `LEASE` (NOT NULL).
- **CHECK:** `Amount > 0`.
- **CHECK:** `Status IN ('Success', 'Failed', 'Pending')`.