PROJECT PART 1

GROUP MEMBERS:

SOHAIB TALIB, MOHAMMAD SUFYAN, NAZIR AHMED.

SUBMIT TO:

SIR SHAKIR RASHEED.

SUBJECT:

Introduction to DataBase.

TOPIC:

ER Diagram on Society/Property Management System

Introduction

The Society / Property Management System is meant to oversee and maintain all operations regarding Housing societies in electronic form.

It manages properties, tenants, members, employees and owners; it keeps track of payments as well as maintenances and complaints.

The product has provided effective data processing, so that there is no interference between society administration and the members.

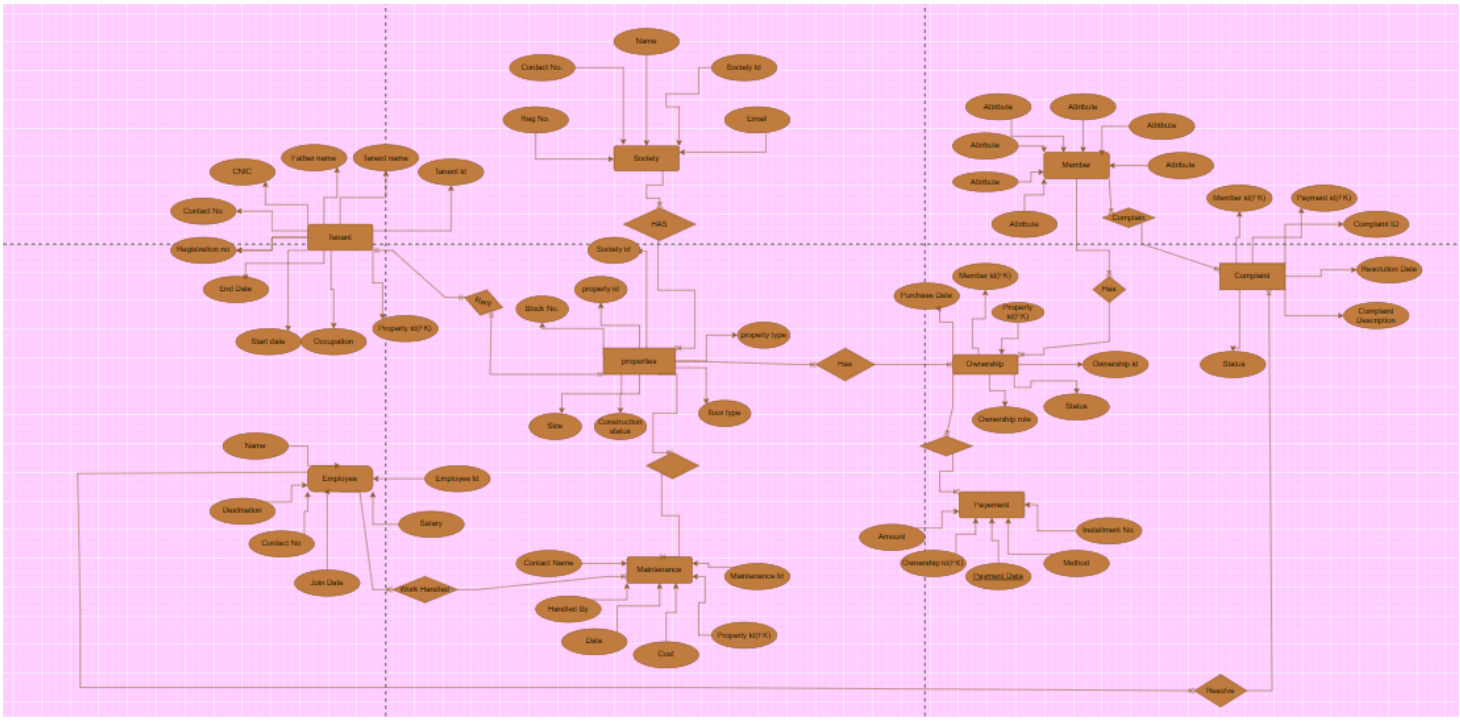
The overall objective is to eliminate manual work, keep accurate records and impart transparency in the administrative process. Using this system, storing, accessing and updating data is easier – thereby streamlining the housing society functions in general.

Objectives of the System

The main objectives of this system are:

- To store detailed records of societies and their properties.
- To manage member and tenant information digitally.
- To record ownership details and link them with members.
- To handle payments and installments securely.
- To maintain a record of complaints and maintenance tasks.
- To assign employees for maintenance handling.
- To create a database model that reflects real-world relationships clearly.

ER Diagram:



Entities and Attributes

1. Society

- Society_ID (PK)
- Name
- Reg_No
- Contact_No
- Email

2. Property

- Property_ID (PK)
- Block_No
- Floor_Type
- Size
- Construction_Status
- Society_ID (FK)

3. Tenant

- Tenant_ID (PK)
- Name
- Father_Name
- CNIC
- Contact_No
- Occupation
- Registration_No
- Start_Date
- End_Date
- Property_ID (FK)

4. Member

- Member_ID (PK)
- Name
- Contact_No
- Email
- Address
- CNIC

5. Ownership

- **Ownership_ID (PK)**
- **Ownership_Role**
- **Purchase_Date**
- **Status**
- **Property_ID (FK)**
- **Member_ID (FK)**

6. Payment

- Payment_ID (PK)
- Amount
- Installment_No

- Method
- Date
- Ownership_ID (FK)

7. Complaint

- Complaint_ID (PK)
- Description
- Resolution_Date
- Status
- Member_ID (FK)
- Payment_ID (FK)

8. Maintenance

- Maintenance_ID (PK)
- Date
- Cost
- Contact_Name
- Property_ID (FK)
- Employee_ID (FK)

9. Employee

- Employee_ID (PK)
- Name
- Designation
- Salary
- Contact_No
- Join_Date

Relationship	Type	Description
Society → Property	1–M	One society can have multiple properties.
Property → Tenant	1–M	A property can have multiple tenants over time.
Property → Ownership	1–M	A property can have multiple ownership records.
Ownership → Member	M–1	Each ownership belongs to one member.
Ownership → Payment	1–M	One ownership can include multiple payments.
Member → Complaint	1–M	Members can make multiple complaints.
Payment → Complaint	1–1 (optional)	Complaints may be linked to specific payments.
Property → Maintenance	1–M	Each property can have multiple maintenance records.
Employee → Maintenance	1–M	Each employee can handle multiple maintenance activities.

Cardinalities:

- One-to-Many: One society → many properties
- One-to-Many: One property → many tenants
- One-to-Many: One property → many maintenance records
- Many-to-One: Many ownerships → one member
- One-to-Many: One ownership → many payments
- One-to-Many: One member → many complaints

System Description:

The *Society / Property Management System* helps manage daily operations in housing societies. It connects multiple entities such as *society, property, tenant, member, ownership, payment, employee,* and *maintenance* to ensure proper data organization.
This system improves record-keeping by maintaining all society-related data in one place, linking each entity through defined relationships. It can be further implemented as a relational database to store, retrieve, and update information efficiently.

Conclusion:

Through the creation of this ER diagram, the structure of the Society Management System becomes clear. All entities are properly linked using relationships to show how data flows within the system. This project demonstrates the importance of ER modeling in database design and provides a strong foundation for future database implementation in SQL or any DBMS tool.

GITHUB LINKS:

Nazir Ahmed: https://github.com/Nazirahmad1122

Muhammad Sufyan: https://github.com/sirsyedan

Sohaib Talib: https://github.com/Allcodewithsohaib