Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

AY: 2024-25

Class:	**SE**	**Semester:**	**IV**
Course Code:	**CSL402**	**Course Name:**	**Database Management System Lab**

Name of Student:	**Shruti Gauchandra**
Roll No. :	**16**
Experiment No.:	**1**
Title of the Experiment:	**Design an Entity Relationship (ER)/ Extended Entity - Relationship (EER) Model.**
Date of Performance:	**02/01/25**
Date of Submission:	**09/01/25**

Evaluation

Performance Indicator	Max. Marks	Marks Obtained
Performance	5	
Understanding	5	
Journal work and timely submission	10	
Total	20	

Performance Indicator	Exceed Expectations (EE)	Meet Expectations (ME)	Below Expectations (BE)
Performance	4-5	2-3	1
Understanding	4-5	2-3	1
Journal work and timely submission	8-10	5-8	1-4

Checked by

Name of Faculty : Ms. Neha Raut

Signature :

Date:

Experiment No 1

Aim :- Identify the case study and detailed statement of the problem. Design an EntityRelationship (ER) / Extended Entity-Relationship (EER) Model.

Objective :- To identify and explore a real world problem, and to design an Entity Relationship (ER) / Extended Entity-Relationship (EER) Model.

Theory:

1. **Entity:**

- An entity is a real-world object or concept that exists independently and has distinguishable attributes.
- In a database context, an entity represents a table, and each row in that table represents a unique instance of that entity.
- For example, in a university database, entities could include Student, Course, Professor, Department, etc.
- Each entity has a set of attributes that describe its properties.

2. **Attributes:**

- Attributes are the properties or characteristics that describe an entity.
- They represent the data we want to store about each instance of an entity.
- For example, attributes of a Student entity might include StudentID, Name, Age, GPA, etc.
- Attributes can be categorized as simple (atomic) attributes, which cannot be divided further, or composite attributes, which are made up of smaller sub-parts.

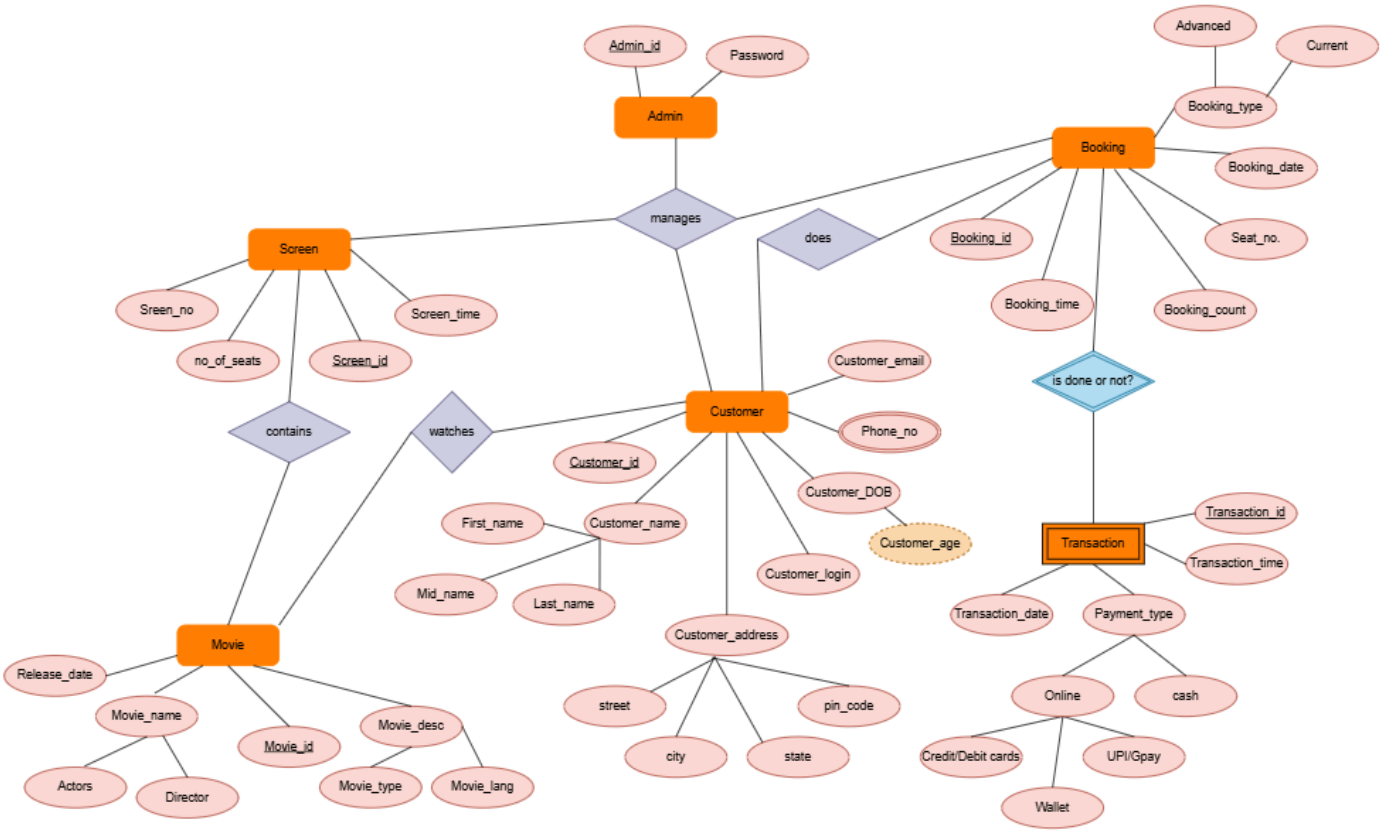
3. Relationships:

- Relationships describe how entities are related to each other or how they interact.
- They represent the associations between entities.
- Relationships are depicted as lines connecting related entities in the ER diagram.
- Each relationship has a degree, indicating the number of entities involved. It could be unary (involving one entity), binary (involving two entities), or ternary (involving three entities).
- Relationships also have cardinality, which defines the number of instances of one entity that can be associated with the number of instances of another entity through the relationship.

4. Cardinality:

- Cardinality specifies the number of instances of one entity that are related to the number of instances of another entity through a relationship.
- It defines the maximum and minimum number of occurrences of one entity that can be associated with the occurrences of another entity.
- Common cardinality constraints include:
  - I. One-to-One (1:1): Each instance of one entity is associated with exactly one instance of another entity, and vice versa.
  - II. One-to-Many (1:N): Each instance of one entity is associated with zero or more instances of another entity, but each instance of the second entity is associated with exactly one instance of the first entity.
  - III. Many-to-One (N:1): The reverse of One-to-Many; many instances of one entity are associated with one instance of another entity.
  - IV. Many-to-Many (N:N): Many instances of one entity can be associated with many instances of another entity.

Implementation



Conclusion

The design of the ER/EER model provides a clear and organized structure for the system's data. By identifying entities, relationships, attributes, and constraints, the model lays a solid foundation for efficient database development. It ensures data integrity, supports complex queries, and allows for future scalability, making it essential for building a reliable and well-structured system.

1. Define Entity, Attributes(also types) and Relationship between entities.

Ans. **Entity:**

An Entity is a real-world object or concept that can be distinctly identified and stored in a database. Each entity has attributes that describe its properties.
Example:

- ***Book*, *Member*, *Librarian* in a Library Management System.**

Attributes:

Attributes are properties or characteristics of an entity.

Types of Attributes:

1. **Simple (Atomic): Cannot be divided further.**
    - ***Example:* Name, Age**
2. **Composite: Can be divided into smaller sub-parts.**
    - ***Example:* Name → First Name, Last Name**
3. **Derived: Can be calculated from other attributes.**
    - ***Example:* Age (from Date of Birth)**
4. **Multivalued: Can have multiple values for a single entity.**
    - ***Example:* Phone Numbers**
5. **Key Attribute: Uniquely identifies an entity.**
    - ***Example:* Student_ID**

Relationship:

A Relationship is an association between two or more entities.

Types of Relationships:

1. **One-to-One (1:1):**
    a. **Each entity in A is related to only one in B, and vice versa.**
    b. ***Example:*** **A librarian manages one section.**
2. **One-to-Many (1:N):**
    a. **One entity in A is related to many in B.**
    b. ***Example:*** **A Member can borrow multiple Books.**
3. **Many-to-Many (M:N):**
    a. **Many entities in A can relate to many in B.**
    b. ***Example:*** **Books can be borrowed by many Members over time.**

2. Write ER/EER diagram notations

Ans. **ER/EER diagram notations are standard symbols used to visually represent the structure of a database. Entities are shown as rectangles, attributes as ovals, and relationships as diamonds. Key attributes are underlined, multivalued attributes use double ovals, and derived attributes are represented with dashed ovals. Lines connect attributes to entities and entities to relationships.**

In EER diagrams, specialization/generalization is shown using a triangle, indicating subclassing. Aggregation is used to model a relationship as an entity itself. Cardinality (like 1:1, 1:N, M:N) is noted near lines to show how many instances participate in relationships. These notations help in clearly defining the database structure and its components.

Component	Notation	Description
Entity	Rectangle	Represents a real-world object or concept.
Attribute	Oval	Describes properties of an entity.
Key Attribute	Oval (underlined)	Uniquely identifies an entity instance.
Relationship	Diamond	Shows the association between entities.
Line	Straight line	Connects entities to attributes and relationships.
Multivalued Attribute	Double oval	Represents attributes with multiple values.
Derived Attribute	Dashed oval	Attribute calculated from other attributes.
Composite Attribute	Oval with connected sub-ovals	Attribute with internal structure.
Cardinality	1:1, 1:N, M:N (on lines)	Indicates the number of instances involved.