Case Study: ER – Diagram

Consider a “**Hostel Management System**” which helps the student to book a room for a period of one year and make the payment in the same portal. There are several types of rooms available in the database like “Standard” or “Deluxe” which has fixed number of occupancy. After booking a room, the student will receive an invoice for the payment.

A "**Hostel Management System**" is a portal where once the admin accepts the signup request ofStudent/Staff then the student gets a privilege to book the room according his/her convenience.For Example: "Standard" or "Deluxe" . The students have to pay the fee according to the type of the room. The rooms also have the occupancies accordingly which a student can choose before booking. And once the booking is done, the student will receive an invoice for the payment.

Consider a **“Mega Mart”** Retail Store application where the customer will be purchasing the items from the retail outlet. The supplier will place the quotation to the retail store from whom items are bought. The order will be placed by the retailer. The retail outlet will be maintaining the employee, customer and the items bought by the customer.

All the details pertaining to **Hostel Management** application are stored in the form of relations.

Few of the relations mentioned below,

* item: details of the items being sold in the retail store.
* supplier: details of the suppliers from whom items are bought to the retail store.
* quotation: details of the quotations placed by the suppliers.
* supplierorder: details of the orders placed for suppliers.
* retailoutlet: details of the various retail outlets.
* employee: details of employees working for the retail outlet.
* customer: details of the customers who buy items from retail outlets.
* address: details of the address.
* USERMASTER: details of the users who registered in **Hostel Management** application
* STAFF: details of the staffs of the hostel are stored
* STUDENT: details of the students are stored
* ROOM: details of the rooms are stored
* ALLOTMENT: it contains the details of room allocation
* INVOICE: it contains the payment details of the students

**Problem Statement 1: Enlist all the Entities and Attributes.**

|  |  |
| --- | --- |
| **Entities** | **Attributes** |
| **USERMASTER** | **userName(PKey), password,FirstName, LastName, dateOfBirth, emailAddress,address, mobile, role, status, photo** |
| **STAFF** | **staffId(PKey Autogenerated),userName(FKey), dateOfJoing, designation, salary** |
| **STUDENT** | **studentId(PKey Autogenerated), username(FKey), gurdianFirstName, gurdianLastName, gurdianMobile** |
| **ROOM** | **roomId(PKey Autogenerated),roomType, roomOccupancy, roomCapacity fee,bookingStatus(Default column="Available")** |
| **ALLOTMENT** | **allocationId(PKey Autogenerated), roomId(FKey), studentId(FKey), status, allocationDate** |
| **INVOICE** | **invoiceId(PKey Autogenerated) , allocationId(FKey), invoiceDate** |

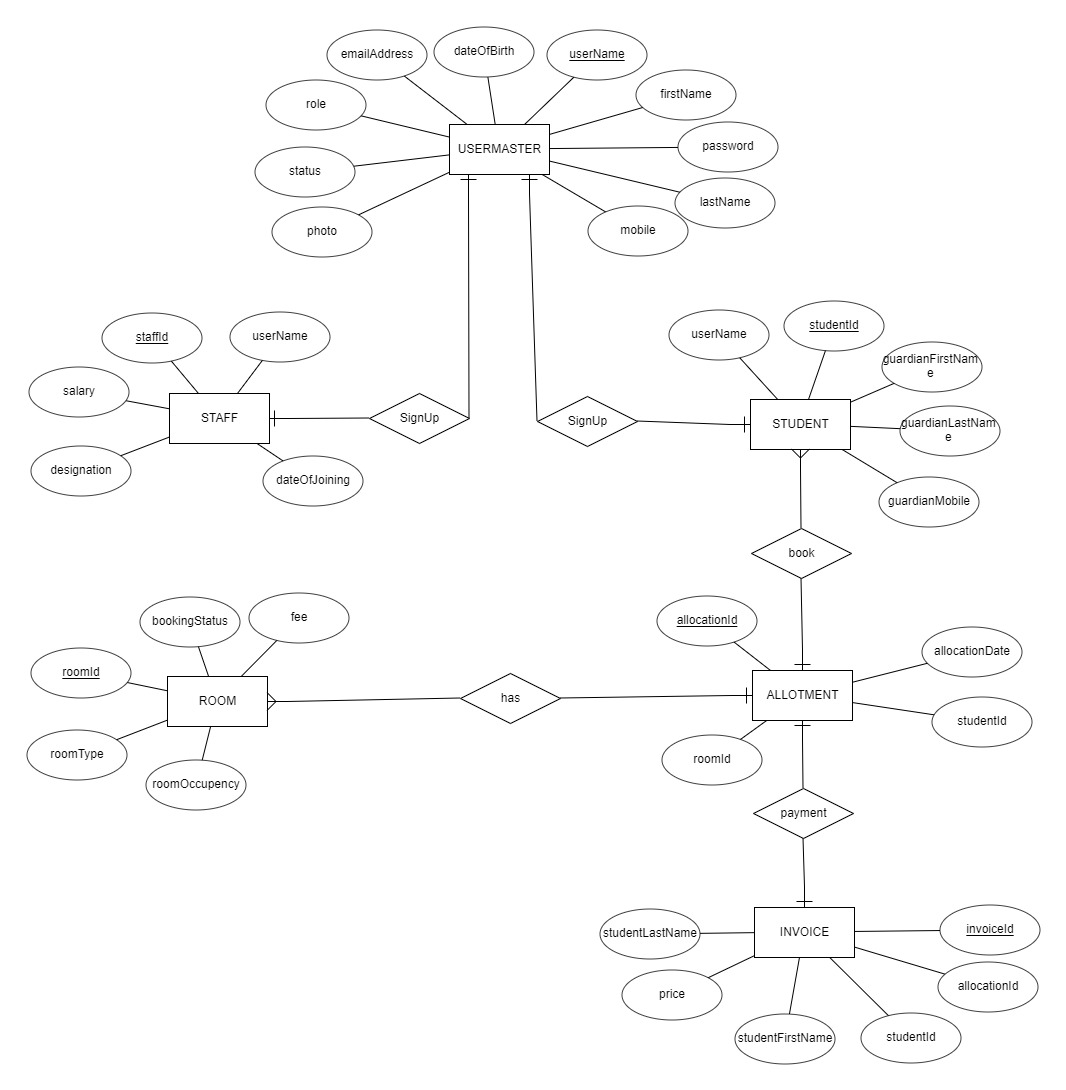
**Problem Statement 2: Identify the candidate key, primary key and foreign keys for the above relation.**

|  |  |  |
| --- | --- | --- |
| **ENTITY** | **PRIMARY KEY** | **FOREIGN KEY** |
| **USERMASTER** | **userName** |  |
| **STAFF** | **staffId** | **userName** |
| **STUDENT** | **studentId** | **userName** |
| **ROOM** | **roomId** |  |
| **ALLOTMENT** | **allocationId** | **studentId, roomId** |
| **INVOICE** | **invoiceId** | **allocationId** |

**Problem Statement 3: Draw the ER model for the above relation.**

**Following are the Steps we should follow**,

1. Identify the Entities
2. Identify the Relationships between Entities
   * One to One
   * One to Many
   * Many to Many
3. Identify the key attributes
4. Draw The ER-Diagram

****

**Problem Statement 4: Identify the Degree of Relationships and Cardinality of all identified relationships.**

Consider the above ER-Diagram pertaining to Retail application scenario and answer the questions below:

* 1. USERMASTER-STAFF (one-to-one),
  2. USERMASTER-STUDENT (one-to-one),
  3. STUDENT-ALLOTMENT (one-to-many),
  4. ROOM-ALLOTMENT (one-to-many),
  5. ALLOTMENT-INVOICE (one-to-one),