DB Task 2 - Smart Home Automation System DB

Why this Domain

I chose the Smart Home Automation domain for database design due to its increasing relevance and impact on modern living. Smart Home Automation systems integrate technology to enhance convenience, security, and energy efficiency within households. The complexity of managing various devices, user roles, and diverse functionalities necessitates a well-structured and normalized database design. This domain presents an opportunity to create a comprehensive system that not only supports the seamless control of devices but also enables efficient user management, device grouping, and scheduling. The database design aims to provide a flexible foundation for accommodating diverse devices, attributes, and user preferences while promoting scalability and maintainability in response to the evolving landscape of smart home technologies.

Entities

User Table:

Stores information about users who interact with the system.

Attributes:

1. **UserID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **Email**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL
3. **Password**

- Datatype: VARCHAR(255)
- Constraints: NOT NULL

UserDetail Table:

Stores additional details about users.

Attributes:

1. **UserDetailID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **FirstName**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL
3. **LastName**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL

UserUserDetailMappings Table:

Associate users with their additional details.

Attributes:

1. **MappingID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **UserID**
    - Datatype: INT (Foreign Key referencing User)
    - Constraints: NOT NULL
3. **UserDetailID**

- Datatype: INT (Foreign Key referencing UserDetail)
- Constraints: NOT NULL

UserRoles Table:

The table defines different roles associated with the system. Here only admin will be the role used.

Attributes:

1. **RoleID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **RoleName**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL

UserRoleMappings Table:

Associate users with their roles.

Attributes:

1. **MappingID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **UserID**
    - Datatype: INT (Foreign Key referencing User)
    - Constraints: NOT NULL
3. **RoleID**
    - Datatype: INT (Foreign Key referencing UserRoles)
    - Constraints: NOT NULL

Houses Table:

Represents physical houses or living spaces.

Attributes:

1. **HouseID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **HouseName**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL
3. **Address**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL

HouseUserMappings Table:

Associates users with their houses.

Attributes:

1. **HouseUserID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **UserID**
    - Datatype: INT (Foreign Key referencing User)
    - Constraints: NOT NULL
3. **HouseID**
    - Datatype: INT (Foreign Key referencing Houses)
    - Constraints: NOT NULL

Rooms Table:

Describes the rooms within each house.

Attributes:

1. **RoomID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **RoomName**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL
3. **Description**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL

RoomHouseMappings Table:

Associates houses with their rooms.

Attributes:

1. **RoomHouseID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **HouseUserID**
    - Datatype: INT (Foreign Key referencing HouseUserMappings)
    - Constraints: NOT NULL
3. **RoomID**
    - Datatype: INT (Foreign Key referencing Rooms)
    - Constraints: NOT NULL

Devices Table:

Represents devices within each room, such as thermostats, lighting, or security cameras.

Attributes:

1. **DeviceID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **DeviceName**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL
3. **DeviceType**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL

Attributes Table:

Describes various attributes that devices may have (e.g., temperature sensor, humidity sensor).

Attributes:

1. **AttributeID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **AttributeType**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL
3. **AttributeValue**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL

DeviceGroups Table:

Groups devices together for convenient management.

Attributes:

1. **GroupID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **GroupName**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL
3. **Description**
    - Datatype: VARCHAR(255)

DeviceRoomAttributeGroupMappings Table:

Groups devices together and associates them with rooms, attributes, and groups.

Attributes:

1. **DeviceMappingID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **DeviceID**
    - Datatype: INT (Foreign Key referencing Devices)
    - Constraints: NOT NULL
3. **RoomHouseID**
    - Datatype: INT (Foreign Key referencing RoomHouseMappings)
    - Constraints: NOT NULL
4. **AttributeID**

- Datatype: INT (Foreign Key referencing Attributes)
- Constraints: NOT NULL

5. **GroupID**
    - Datatype: INT (Foreign Key referencing DeviceGroups)
    - Constraints: NOT NULL

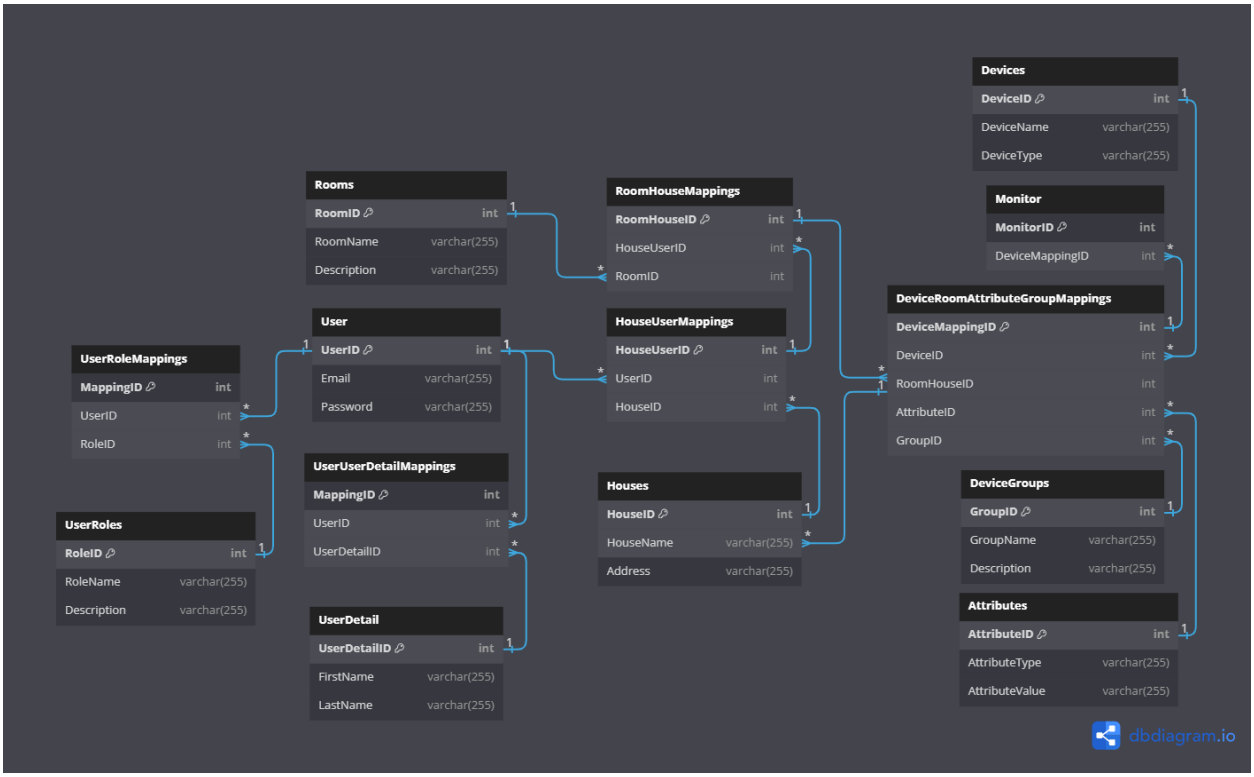
Monitor Table:

Stores monitor information associated with device mappings.

Attributes:

1. **MonitorID**
    - Datatype: INT
    - Constraints: PRIMARY KEY, NOT NULL
2. **DeviceMappingID**
    - Datatype: INT (Foreign Key referencing DeviceRoomAttributeGroupMappings)
    - Constraints: NOT NULL
3. **Status**
    - Datatype: VARCHAR(255)
    - Constraints: NOT NULL

Entity-Relationship Diagram



Normalization & Final Tables

To determine if the database is in 3NF (Third Normal Form), we need to ensure that:

1. It is already in 1NF and 2NF.
2. There are no transitive dependencies.

It is in 3NF since there are no transitive dependencies within each table.

Entities in a table format:

User		
UserID (PK, INT)	Email	Password

UserDetail		
UserDetailID (PK, INT)	FirstName	LastName

UserUserDetailMappings		

MappingID (PK, INT)	UserID (FK)	UserDetailID (FK)

UserRoles		
RoleID (PK, INT)	RoleName	Description

UserRoleMappings		
MappingID (PK, INT)	UserID (FK)	RoleID (FK)

Houses		
HouseID (PK, INT)	HouseName	Address

HouseUserMappings		
HouseUserID (PK, INT)	UserID (FK)	HouseID (FK)

Rooms		
RoomID (PK, INT)	RoomName	Description

RoomHouseMappings		
RoomHouseID (PK, INT)	HouseUserID (FK)	RoomID (FK)

Devices		
DeviceID (PK, INT)	DeviceName	DeviceType

Attributes		
AttributeID (PK, INT)	AttributeType	AttributeValue

DeviceGroups		
GroupID (PK, INT)	GroupName	Description

DeviceRoomAttributeGroupMappings		
DeviceMappingID (PK, INT)	DeviceID (FK)	RoomHouseID (FK)

Monitor		
MonitorID (PK, INT)	DeviceMappingID (FK)	Status

Code

```
-- Create the database
CREATE DATABASE IF NOT EXISTS SmartHomeAutomationDB;

-- Use the created database
USE SmartHomeAutomationDB;

-- Create User table
CREATE TABLE User (
    UserID INT PRIMARY KEY,
    Email VARCHAR(255),
    Password VARCHAR(255)
);

-- Create UserDetail table
CREATE TABLE UserDetail (
    UserDetailID INT PRIMARY KEY,
    FirstName VARCHAR(255),
    LastName VARCHAR(255)
);

-- Create UserUserDetailMappings table
CREATE TABLE UserUserDetailMappings (
    MappingID INT PRIMARY KEY,
    UserID INT,
    UserDetailID INT,
    FOREIGN KEY (UserID) REFERENCES User(UserID),
    FOREIGN KEY (UserDetailID) REFERENCES UserDetail(UserDetail
);

-- Create UserRoles table
CREATE TABLE UserRoles (
    RoleID INT PRIMARY KEY,
    RoleName VARCHAR(255),
```

```
    Description VARCHAR(255)
);

-- Create UserRoleMappings table
CREATE TABLE UserRoleMappings (
    MappingID INT PRIMARY KEY,
    UserID INT,
    RoleID INT,
    FOREIGN KEY (UserID) REFERENCES User(UserID),
    FOREIGN KEY (RoleID) REFERENCES UserRoles(RoleID)
);

-- Create Houses table
CREATE TABLE Houses (
    HouseID INT PRIMARY KEY,
    HouseName VARCHAR(255),
    Address VARCHAR(255)
);

-- Create HouseUserMappings table
CREATE TABLE HouseUserMappings (
    HouseUserID INT PRIMARY KEY,
    UserID INT,
    HouseID INT,
    FOREIGN KEY (UserID) REFERENCES User(UserID),
    FOREIGN KEY (HouseID) REFERENCES Houses(HouseID)
);

-- Create Rooms table
CREATE TABLE Rooms (
    RoomID INT PRIMARY KEY,
    RoomName VARCHAR(255),
    Description VARCHAR(255)
);

-- Create RoomHouseMappings table
```

```
CREATE TABLE RoomHouseMappings (
    RoomHouseID INT PRIMARY KEY,
    HouseUserID INT,
    RoomID INT,
    FOREIGN KEY (HouseUserID) REFERENCES HouseUserMappings(House
    FOREIGN KEY (RoomID) REFERENCES Rooms(RoomID)
);

-- Create Devices table
CREATE TABLE Devices (
    DeviceID INT PRIMARY KEY,
    DeviceName VARCHAR(255),
    DeviceType VARCHAR(255)
);

-- Create Attributes table
CREATE TABLE Attributes (
    AttributeID INT PRIMARY KEY,
    AttributeType VARCHAR(255),
    AttributeValue VARCHAR(255)
);

-- Create DeviceGroups table
CREATE TABLE DeviceGroups (
    GroupID INT PRIMARY KEY,
    GroupName VARCHAR(255),
    Description VARCHAR(255)
);

-- Create DeviceRoomAttributeGroupMappings table
CREATE TABLE DeviceRoomAttributeGroupMappings (
    DeviceMappingID INT PRIMARY KEY,
    DeviceID INT,
    RoomHouseID INT,
    AttributeID INT,
    GroupID INT,
```

```
    FOREIGN KEY (DeviceID) REFERENCES Devices(DeviceID),
    FOREIGN KEY (RoomHouseID) REFERENCES RoomHouseMappings(RoomH
    FOREIGN KEY (AttributeID) REFERENCES Attributes(AttributeID)
    FOREIGN KEY (GroupID) REFERENCES DeviceGroups(GroupID)
);

-- Create Monitor table
CREATE TABLE Monitor (
    MonitorID INT PRIMARY KEY,
    DeviceMappingID INT,
    Status VARCHAR(255),
    FOREIGN KEY (DeviceMappingID) REFERENCES DeviceRoomAttribute
);
```