|  |  |  |
| --- | --- | --- |
| USER | **ERD** | DEVICE |
| UserID(PK)  Name  Role  Preferences | DeviceID(PK)  DeviceType  Status  Settings  RoomId(FK) |

|  |
| --- |
| ROOM |
| RoomId(PK)  RoomName |

**LOGICAL DATA MODEL**

The **Logical Data Model (LDM)** for a **Smart Home Automation System** describes how the data is organized logically within the database, without focusing on the physical storage or implementation details. It outlines the structure of entities, their attributes.

-USER: The user entity represents individuals relating with the smart home system. A user can control devices and have preferences for automation.

.Attributes:

UserID (primary key): A unique identifier for each user.

Name: The name of the user.

Role:The role of the user (e.g: admin, guest), which determines the level of access they have.

Preferences: The user’s settings or preferences (e.g: preferred temperature, light intensity).

-Device: The device entity represents all the smart devices in the system, such as lights, sensors, cameras, and speakers.

.Attributes:

DeviceID (Primary Key): A unique identifier for each device.

DeviceType: Type of device (e.g: light, camera).

Status: Current state of the device (e.g: on or off).

Settings: Arrangement or setting details for the device.

RoomId (Foreign Key): A location to the room the device is located in.

-Room: The Room entity represents the physical rooms or areas within the home where devices are located.

.Attributes:

RoomId (Primary Key): A unique identifier for each room.

RoomName: Name of the room (e.g: living room, kitchen, bedroom).

**PHYSICAL DATA MODEL**

The Physical data model is the next step after the Logical Data Model (LDM). It takes the entities, attributes, and relationships defined in the LDM and translates them into a structure that can be implemented in a specific database management system (DBMS).

Tables: The physical tables are the concrete implementations of the entities from the logical model (e.g: User, Device, Room). Each table will be implemented in the DBMS with specific column data types, constraints, and other physical considerations.

Indexes: To improve the speed of data retrieval, especially for frequently queried fields, indexes are created on columns that are often searched or joined, such as UserID, DeviceID, and RoomID.

Foreign Keys: Foreign key relationships are defined to enforce referential integrity between tables. For example, the Device table will have a foreign key linking it to the Room table, and the User\_Device table will have foreign keys linking to both the User and Device tables.

Data Types: In the PDM, the data types of each column are specified (e.g: INT, VARCHAR, DATE).

Constraints: Constraints like NOT NULL, UNIQUE, CHECK values are set to enforce data integrity.

**DATA DICTIONARY**

**Data Dictionary**: provides a detailed description of each element in the database schema. It defines the tables, columns, data types, constraints, and relationships, ensuring that everyone understands the structure and purpose of each data element.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table  Name | Column Name | Data Type | Description | Constraints | Indexes |
| User | UserID | int | Unique identifier of  Each user | Primary key  Auto\_increment | Index on name (optional) |
|  | Name | Varchar(100) | Full name of the user | Not null |  |
|  | Role | Varchar(50) | The user’s role(e.g: Admin, Quest) | Not null |  |
|  | Preferences | text | User preferences for smart home automation | nullable |  |
| Device | DeviceID | int | Unique identifier of  Each device | Primary key  Auto\_increment | Index on name (optional) |
|  | DeviceType | Varchar(50) | Type of device (e.g:light,camera) | Not null | Index on device type |
|  | Status | boolean | Current status of the device (on/off) | Not null | Index on status (optional) |
|  | Settings | text | Settings for the device (e.g., temperature, brightness) | nullable |  |
|  | RoomID | int | Foreign key referencing the room the device is located in | Foreign Key (Room.RoomID) |  |
| Room | RoomID | int | Unique identifier for each room | Primary Key, Auto-increment | Index on RoomName (optional) |
|  | RoomName | Varchar(50) | Name of the room (e.g., Living Room, Kitchen, Bedroom) | Not null |  |
| User\_Device | UserID | int | Foreign key referencing the user | Foreign Key (User.UserID) | Index on UserID, Index on DeviceID |
|  | DeviceID | (UserID, DeviceID) | Composite key to ensure uniqueness in user-device relationship | Primary Key |  |