***1.Designing (Entity Relationship)ER Diagram***

**Steps of Drawing ERD :**

**1. Identify the Entities Required**

**2. Identify the Attributes and Primary key for each Entity**

**3. Identify the Relationship needed**

**4. Identify the Cardinality Ratio and Participation**

**5. Draw the Diagram**

**Project:** **Virtual plant map management system.**

A **Virtual Plant Map Management System (VPMMS)** is an online application designed to manage and visualize the layout of industrial plants or manufacturing facilities. This system can track various components such as equipment, machinery, inventory, rooms, zones, and utilities in a plant, allowing users to interact with a digital representation of the facility.

**Key Features:**

* **Visualization of plant layout**: Create and display digital maps of plants.
* **Asset management**: Track and manage equipment and machinery.
* **Inventory management**: Keep track of spare parts and materials.
* **Maintenance tracking**: Schedule and monitor the status of equipment maintenance.
* **User roles and access management**: Define different roles like Admin, Engineer, and Manager with different access levels.
* **Interactive dashboard**: Allow users to click on areas of the plant map for detailed information and data.

**Step-1: Identify the Entities Required.**

For the **Virtual Plant Map Management System**, the primary entities involved would include:

1. **Plant**
2. **Room**
3. **Zone**
4. **Equipment**
5. **Inventory**
6. **Maintenance**
7. **User**
8. **Asset**

**Step-2: Identify the Attributes and Primary key for each Entity.**

1. **Plant:** Plant\_ID (PK), Name, Location, Area, Description.
2. **Room:** Room\_ID (PK), Room\_Name, Room\_Type, Dimensions, Plant\_ID (FK).
3. **Zone:** Zone\_ID (PK), Zone\_Name, Description, Plant\_ID (FK).
4. **Equipment:** Equipment\_ID (PK), Equipment\_Name, Equipment\_Type, Manufacturer, Status, Room\_ID (FK).
5. **Inventory:** Inventory\_ID (PK), Item\_Name, Item\_Type, Quantity, Location (FK).
6. **Maintenance:** Maintenance\_ID (PK), Equipment\_ID (FK), Maintenance\_Type, Scheduled\_Date, Completion\_Date, Status.
7. **User:** User\_ID (PK), Username, Password, Role, Full\_Name, Email, Plant\_ID (FK).
8. **Asset:** Asset\_ID (PK), Asset\_Name, Asset\_Type, Plant\_ID (FK), Equipment\_ID (FK), Value.

**Step-3: Identify the Relationship needed**

1. **Plant to Room**: A one-to-many relationship (One plant can have multiple rooms).
2. **Plant to Zone**: A one-to-many relationship (One plant can have multiple zones).
3. **Room to Equipment**: A one-to-many relationship (One room can contain many pieces of equipment).
4. **Zone to Equipment**: A many-to-many relationship (A zone can have multiple equipment, and equipment can be used across multiple zones).
5. **Equipment to Maintenance**: A one-to-many relationship (Each piece of equipment may have multiple maintenance records).
6. **Inventory to Equipment**: A one-to-many relationship (Inventory items are used to support equipment maintenance).
7. **User to Plant**: A many-to-one relationship (Users belong to a specific plant).
8. **Asset to Equipment**: A one-to-one or one-to-many relationship (An asset can be linked to one or more pieces of equipment).

**Step-4: Identify the Cardinality Ratio and Participation.**

**1.Plant to Room**: A one-to-many relationship.

Plant

Room

Have

**2.Plant to Zone**: A one-to-many relationship.

Have

Zone

Plant

**3.Room to Equipment**: A one-to-many relationship.

Equipment

Involve

Room

**4.Zone to Equipment**: A many-to-many relationship.

Used

Equipment

Zone

**5.Equipment to Maintenance**: A one-to-many relationship.

Maintenance

Management

Equipment

**6.Inventory to Equipment**: A one-to-many relationship.

Supported

Equipment

Inventory

**7.User to Plant**: A many-to-one relationship.

Plant

Used

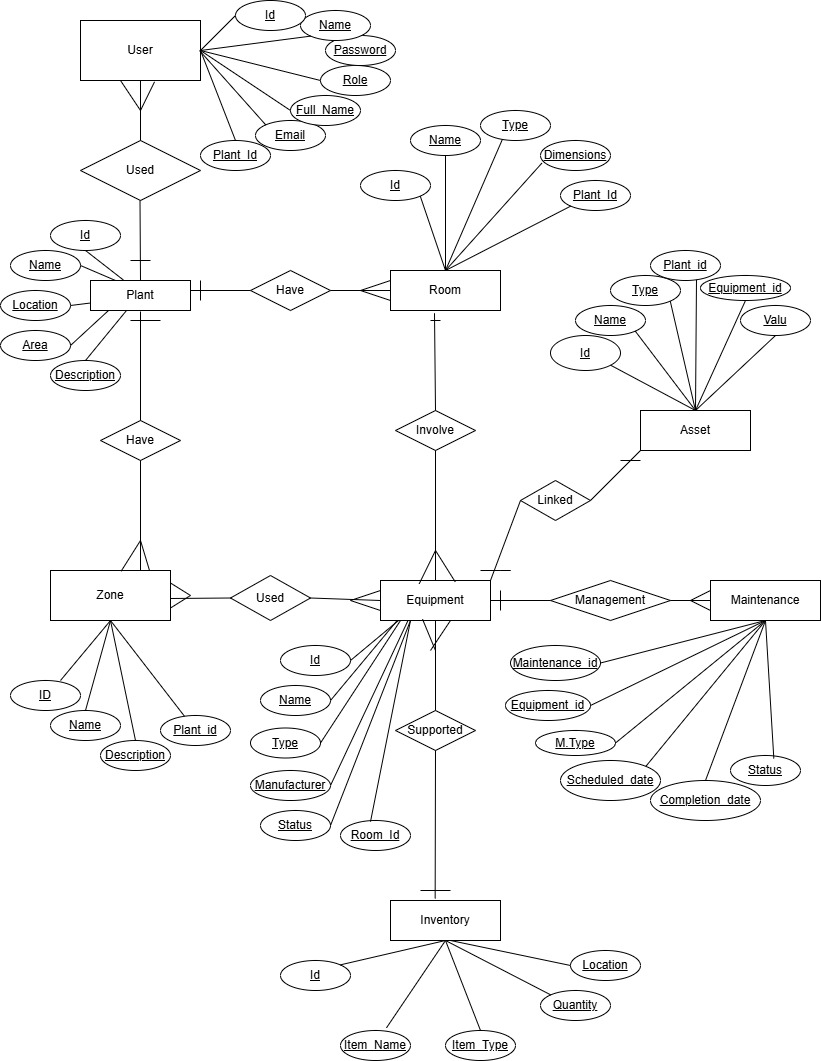
User

**8.Asset to Equipment**: A one-to-one or one-to-many relationship.

Linked

Equipment

Asset

**Step-5: Draw the Diagram.**

**2. Reduction to database schema:**

**1.Plant:** Plant\_ID (PK), Name, Location, Area, Description.

**2.Room:** Room\_ID (PK), Room\_Name, Room\_Type, Dimensions, Plant\_ID (FK).

**3.Zone:** Zone\_ID (PK), Zone\_Name, Description, Plant\_ID (FK).

**4.Equipment:** Equipment\_ID (PK), Equipment\_Name, Equipment\_Type, Manufacturer, Status, Room\_ID (FK).

**5.Inventory:** Inventory\_ID (PK), Item\_Name, Item\_Type, Quantity, Location (FK).

**6.Maintenance:** Maintenance\_ID (PK), Equipment\_ID (FK), Maintenance\_Type, Scheduled\_Date, Completion\_Date, Status.

**7.User:** User\_ID (PK), Username, Password, Role, Full\_Name, Email, Plant\_ID (FK).

**8.Asset:** Asset\_ID (PK), Asset\_Name, Asset\_Type, Plant\_ID (FK), Equipment\_ID (FK), Value.

**3.Implementing the database in MySQL:**

**All tables with sample data:**

**1.Plant:** Plant\_ID (PK), Name, Location, Area, Description.

CREATE TABLE Plant (

Plant\_ID INT PRIMARY KEY,

Name VARCHAR(255) NOT NULL,

Location VARCHAR(255),

Area DECIMAL(10, 2), -- Assuming area is in square meters or square feet

Description TEXT

);

**2.Room:** Room\_ID (PK), Room\_Name, Room\_Type, Dimensions, Plant\_ID (FK).

CREATE TABLE Room (

Room\_ID INT PRIMARY KEY,

Room\_Name VARCHAR(255) NOT NULL,

Room\_Type VARCHAR(100),

Dimensions VARCHAR(100), -- Can store dimensions like "10x20 meters"

Plant\_ID INT,

FOREIGN KEY (Plant\_ID) REFERENCES Plant(Plant\_ID) ON DELETE CASCADE

);

**3.Zone:** Zone\_ID (PK), Zone\_Name, Description, Plant\_ID (FK).

CREATE TABLE Zone (

Zone\_ID INT PRIMARY KEY,

Zone\_Name VARCHAR(255) NOT NULL,

Description TEXT,

Plant\_ID INT,

FOREIGN KEY (Plant\_ID) REFERENCES Plant(Plant\_ID) ON DELETE CASCADE

);

**4.Equipment:** Equipment\_ID (PK), Equipment\_Name, Equipment\_Type, Manufacturer, Status, Room\_ID (FK).

CREATE TABLE Equipment (

Equipment\_ID INT PRIMARY KEY,

Equipment\_Name VARCHAR(255) NOT NULL,

Equipment\_Type VARCHAR(100),

Manufacturer VARCHAR(255),

Status VARCHAR(50), -- Status like "Active", "Under Maintenance"

Room\_ID INT,

**FOREIGN KEY (Room\_ID) REFERENCES Room(Room\_ID) ON DELETE CASCADE**

**);**

**5.Inventory:** Inventory\_ID (PK), Item\_Name, Item\_Type, Quantity, Location (FK).

CREATE TABLE Inventory (

Inventory\_ID INT PRIMARY KEY,

Item\_Name VARCHAR(255) NOT NULL,

Item\_Type VARCHAR(100),

Quantity INT,

Location VARCHAR(255), -- Location could refer to a room or zone

FOREIGN KEY (Location) REFERENCES Room(Room\_ID) -- Assuming Location refers to Room\_ID for simplicity

);

**6.Maintenance:** Maintenance\_ID (PK), Equipment\_ID (FK), Maintenance\_Type, Scheduled\_Date, Completion\_Date, Status.

CREATE TABLE Maintenance (

Maintenance\_ID INT PRIMARY KEY,

Equipment\_ID INT,

Maintenance\_Type VARCHAR(100),

Scheduled\_Date DATE,

Completion\_Date DATE,

Status VARCHAR(50),

FOREIGN KEY (Equipment\_ID) REFERENCES Equipment(Equipment\_ID) ON DELETE CASCADE

);

**7.User:** User\_ID (PK), Username, Password, Role, Full\_Name, Email, Plant\_ID (FK).

CREATE TABLE User (

User\_ID INT PRIMARY KEY,

Username VARCHAR(255) NOT NULL,

Password VARCHAR(255) NOT NULL,

Role VARCHAR(50), -- e.g., Admin, Manager, Engineer

Full\_Name VARCHAR(255),

Email VARCHAR(255),

Plant\_ID INT,

FOREIGN KEY (Plant\_ID) REFERENCES Plant(Plant\_ID) ON DELETE CASCADE

);

**8.Asset:** Asset\_ID (PK), Asset\_Name, Asset\_Type, Plant\_ID (FK), Equipment\_ID (FK), Value.

CREATE TABLE Asset (

Asset\_ID INT PRIMARY KEY,

Asset\_Name VARCHAR(255) NOT NULL,

Asset\_Type VARCHAR(100),

Plant\_ID INT,

Equipment\_ID INT,

Value DECIMAL(15, 2), -- Monetary value of the asset

FOREIGN KEY (Plant\_ID) REFERENCES Plant(Plant\_ID) ON DELETE CASCADE,

FOREIGN KEY (Equipment\_ID) REFERENCES Equipment(Equipment\_ID) ON DELETE CASCADE

);