Database Management System (IT615)

ERD To Relational Model, DDL

for

Sustainable Agriculture Resource Management

Group Number: 02
Aarushi Goel (202412002)
Jayesh Chauhan (202412012)

Mapping E-R Model to Relational Model

Schema

1. Farmer(**FarmerID**, Name, FarmSize, ContactInfo)
2. Crop(**CropID**, Name, Type, GrowthPeriod, Yield)
3. Technology(**TechnologyID**, Name, Type, Purpose)
4. Soil(**SoilID**, Type, NutrientContent, pHLevel, MoistureLevel)
5. Fertilizer(**FertilizerID**, Name, Type, NutrientContent, ApplicationMethod)
6. TrainingProgram(**ProgramID**, Title, Duration, TargetAudience, Content)
7. Visitor(**VisitorID**, Name, Role, ContactInfo)
8. Policy(**PolicyID**, Name, Type, ImplementationDate)
9. Aid(**AidID**, Type, Amount, Beneficiary, Date)
10. CropDisease(**DiseaseID**, Name, AffectedCrops, Symptoms, Treatment)
11. CropRotation(**RotationID**, CropSequence, Duration, Benefits)
12. Investment(**InvestmentID**, Amount, Purpose, Beneficiary, Date)
13. Weather(**WeatherID**, Date, Temperature, Precipitation, Conditions)
14. SustainablePractice(**PracticeID**, Name, Description, Benefits, ImplementationLevel)

Relational Schema

1. Farmer_Crop(**FarmerID, CropID**)
2. Farmer_Tech(**FarmerID, TechnologyID**, TimeofUsage)
3. FarmingTool(**ToolID**, Name, Type, Usage, Manufacturer, FarmerID)
4. Irrigation(**IrrigationID**, Type, CoverageArea, Efficiency, InstallationDate, FarmerID)
5. Farmer_Soil(**FarmerID, SoilID**)
6. Farmer_Fertilizer(**FarmerID, FertilizerID**, Quantity)
7. Farmer_TrainingProgram(**FarmerID, ProgramID**)

8. Crop_Fertilizer(**CropID, FertilizerID**, FertilizerQuantity)
9. Crop_Disease(**CropID, DiseaseID**)
10. Farmer_Aid(**FarmerID, AidID**)
11. Farmer_Investment(**FarmerID, InvestmentID**)
12. Irrigation_Soil(**IrrigationID, SoilID**)
13. Crop_Soil(**CropID, SoilID**)
14. Crop_Weather(**CropID, WeatherID**)
15. Farmer_SustainablePractice(**FarmerID, PracticeID**)

DDL scripts for Schema

1. Farmer

```
CREATE TABLE Farmer (
    FarmerID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    FarmSize DECIMAL(10, 2) CHECK (FarmSize > 0),
    ContactInfo VARCHAR(100)
);
```

2. Crop

```
CREATE TABLE Crop (
    CropID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Type VARCHAR(50),
    GrowthPeriod INT CHECK (GrowthPeriod >= 0),
    Yield DECIMAL(10, 2) CHECK (Yield >= 0)
);
```

3. Technology

```
CREATE TABLE Technology (
    TechnologyID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Type VARCHAR(50),
    Purpose VARCHAR(100)
);
```

4. Soil

```
CREATE TABLE Soil (
    SoilID INT PRIMARY KEY,
    Type VARCHAR(50),
    NutrientContent VARCHAR(100),
    pHLevel DECIMAL(3, 2) CHECK (pHLevel >= 0 AND pHLevel
<= 14),
    MoistureLevel DECIMAL(5, 2) CHECK (MoistureLevel >= 0)
);
```

5. Fertilizer

```
CREATE TABLE Fertilizer (
    FertilizerID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Type VARCHAR(50),
    NutrientContent VARCHAR(100),
    ApplicationMethod VARCHAR(100)
);
```

6. Training Program

```
CREATE TABLE TrainingProgram (
    ProgramID INT PRIMARY KEY,
    Title VARCHAR(100) NOT NULL,
    Duration INT CHECK (Duration > 0),
    TargetAudience VARCHAR(100),
    Content VARCHAR(100)
);
```

7. Visitor

```
CREATE TABLE Visitor (
    VisitorID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Role VARCHAR(50),
    ContactInfo VARCHAR(100)
);
```

8. Policy

```
CREATE TABLE Policy (
    PolicyID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Type VARCHAR(50),
    ImplementationDate DATE
);
```

9. Aid

```
CREATE TABLE Aid (
    AidID INT PRIMARY KEY,
    Type VARCHAR(50),
    Amount DECIMAL(10, 2) CHECK (Amount > 0),
    Beneficiary VARCHAR(100),
    Date DATE
);
```

10. Crop Disease

```
CREATE TABLE CropDisease (
    DiseaseID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    AffectedCrops VARCHAR(100),
    Symptoms VARCHAR(100),
    Treatment VARCHAR(100)
);
```

11. Crop Rotation

```
CREATE TABLE CropRotation (
    RotationID INT PRIMARY KEY,
    CropSequence VARCHAR(100)  NOT NULL,
    Duration INT CHECK (Duration > 0),
    Benefits VARCHAR(100)
);
```

12. Investment

```
CREATE TABLE Investment (
    InvestmentID INT PRIMARY KEY,
    Amount DECIMAL(10, 2) CHECK (Amount > 0),
    Purpose VARCHAR(100),
    Beneficiary VARCHAR(100),
    Date DATE
);
```

13. Weather

```
CREATE TABLE Weather (
    WeatherID INT PRIMARY KEY,
    Date DATE NOT NULL,
    Temperature DECIMAL(5, 2),
    Precipitation DECIMAL(5, 2),
    Conditions VARCHAR(100)
);
```

14. Sustainable Practice

```
CREATE TABLE SustainablePractice (
    PracticeID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Description VARCHAR(100),
    Benefits VARCHAR(100),
    ImplementationLevel VARCHAR(50)
);
```

DDL scripts for Relational Schema

1. Farmer_Crop

```
CREATE TABLE Farmer_Crop (
    FarmerID INT,
    CropID INT,
    PRIMARY KEY (FarmerID, CropID),
```

```
    FOREIGN KEY (FarmerID) REFERENCES Farmer(FarmerID)
ON DELETE CASCADE,
    FOREIGN KEY (CropID) REFERENCES Crop(CropID) ON
DELETE CASCADE
);
```

2. Farmer_Tech

```
CREATE TABLE Farmer_Tech (
    FarmerID INT,
    TechnologyID INT,
    TimeofUsage DATE,
    PRIMARY KEY (FarmerID, TechnologyID),
    FOREIGN KEY (FarmerID) REFERENCES Farmer(FarmerID)
ON DELETE CASCADE,
    FOREIGN KEY (TechnologyID) REFERENCES
Technology(TechnologyID) ON DELETE CASCADE
);
```

3. FarmingTool

```
CREATE TABLE FarmingTool (
    ToolID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Type VARCHAR(50),
    Usage TEXT,
    Manufacturer VARCHAR(100),
    FarmerID INT,
    FOREIGN KEY (FarmerID) REFERENCES Farmer(FarmerID)
ON DELETE CASCADE
);
```

4. Irrigation

```
CREATE TABLE Irrigation (
    IrrigationID INT PRIMARY KEY,
    Type VARCHAR(50),
    CoverageArea DECIMAL(10, 2) CHECK (CoverageArea > 0),
```

```
    Efficiency DECIMAL(5, 2) CHECK (Efficiency >= 0 AND
Efficiency <= 100),
    InstallationDate DATE,
    FarmerID INT,
    FOREIGN KEY (FarmerID) REFERENCES Farmer(FarmerID)
ON DELETE CASCADE
);
```

5. Farmer_Soil

```
CREATE TABLE Farmer_Soil (
    FarmerID INT,
    SoilID INT,
    PRIMARY KEY (FarmerID, SoilID),
    FOREIGN KEY (FarmerID) REFERENCES Farmer(FarmerID)
ON DELETE CASCADE,
    FOREIGN KEY (SoilID) REFERENCES Soil(SoilID) ON
DELETE CASCADE
);
```

6. Farmer_Fertilizer

```
CREATE TABLE Farmer_Fertilizer (
    FarmerID INT,
    FertilizerID INT,
    Quantity DECIMAL(10, 2) CHECK (Quantity >= 0),
    PRIMARY KEY (FarmerID, FertilizerID),
    FOREIGN KEY (FarmerID) REFERENCES Farmer(FarmerID)
ON DELETE CASCADE,
    FOREIGN KEY (FertilizerID) REFERENCES
Fertilizer(FertilizerID) ON DELETE CASCADE
);
```

7. Farmer_TrainingProgram

```
CREATE TABLE Farmer_TrainingProgram (
    FarmerID INT,
    ProgramID INT,
```

```
    PRIMARY KEY (FarmerID, ProgramID),
    FOREIGN KEY (FarmerID) REFERENCES Farmer(FarmerID)
ON DELETE CASCADE,
    FOREIGN KEY (ProgramID) REFERENCES
TrainingProgram(ProgramID) ON DELETE CASCADE
);
```

8. Crop_Fertilizer

```
CREATE TABLE Crop_Fertilizer (
    CropID INT,
    FertilizerID INT,
    FertilizerQuantity DECIMAL(10, 2) CHECK (FertilizerQuantity
>= 0),
    PRIMARY KEY (CropID, FertilizerID),
    FOREIGN KEY (CropID) REFERENCES Crop(CropID) ON
DELETE CASCADE,
    FOREIGN KEY (FertilizerID) REFERENCES
Fertilizer(FertilizerID) ON DELETE CASCADE
);
```

9. Crop_Disease

```
CREATE TABLE Crop_Disease (
    CropID INT,
    DiseaseID INT,
    PRIMARY KEY (CropID, DiseaseID),
    FOREIGN KEY (CropID) REFERENCES Crop(CropID) ON
DELETE CASCADE,
    FOREIGN KEY (DiseaseID) REFERENCES
CropDisease(DiseaseID) ON DELETE CASCADE
);
```

10. Farmer_Aid

```
CREATE TABLE Farmer_Aid (
    FarmerID INT,
    AidID INT,
```

```
    PRIMARY KEY (FarmerID, AidID),
    FOREIGN KEY (FarmerID) REFERENCES Farmer(FarmerID)
ON DELETE CASCADE,
    FOREIGN KEY (AidID) REFERENCES Aid(AidID) ON
DELETE CASCADE
);
```

11. Farmer_Investment

```
CREATE TABLE Farmer_Investment (
    FarmerID INT,
    InvestmentID INT,
    PRIMARY KEY (FarmerID, InvestmentID),
    FOREIGN KEY (FarmerID) REFERENCES Farmer(FarmerID)
ON DELETE CASCADE,
    FOREIGN KEY (InvestmentID) REFERENCES
Investment(InvestmentID) ON DELETE CASCADE
);
```

12. Irrigation_Soil

```
CREATE TABLE Irrigation_Soil (
    IrrigationID INT,
    SoilID INT,
    PRIMARY KEY (IrrigationID, SoilID),
    FOREIGN KEY (IrrigationID) REFERENCES
Irrigation(IrrigationID) ON DELETE CASCADE,
    FOREIGN KEY (SoilID) REFERENCES Soil(SoilID) ON
DELETE CASCADE
);
```

13. Crop_Soil

```
CREATE TABLE Crop_Soil (
    CropID INT,
    SoilID INT,
    PRIMARY KEY (CropID, SoilID),
```

```
    FOREIGN KEY (CropID) REFERENCES Crop(CropID) ON
DELETE CASCADE,
    FOREIGN KEY (SoilID) REFERENCES Soil(SoilID) ON
DELETE CASCADE
);
```

14. Crop_Weather

```
CREATE TABLE Crop_Weather (
    CropID INT,
    WeatherID INT,
    PRIMARY KEY (CropID, WeatherID),
    FOREIGN KEY (CropID) REFERENCES Crop(CropID) ON
DELETE CASCADE,
    FOREIGN KEY (WeatherID) REFERENCES Weather(WeatherID)
ON DELETE CASCADE
);
```

15. Farmer_SustainablePractice

```
CREATE TABLE Farmer_SustainablePractice (
    FarmerID INT,
    PracticeID INT,
    PRIMARY KEY (FarmerID, PracticeID),
    FOREIGN KEY (FarmerID) REFERENCES Farmer(FarmerID)
ON DELETE CASCADE,
    FOREIGN KEY (PracticeID) REFERENCES
SustainablePractice(PracticeID) ON DELETE CASCADE
);
```