# Phase III

Logical Data Model Design: Agricultural Management System

## Tasks and Deliverables

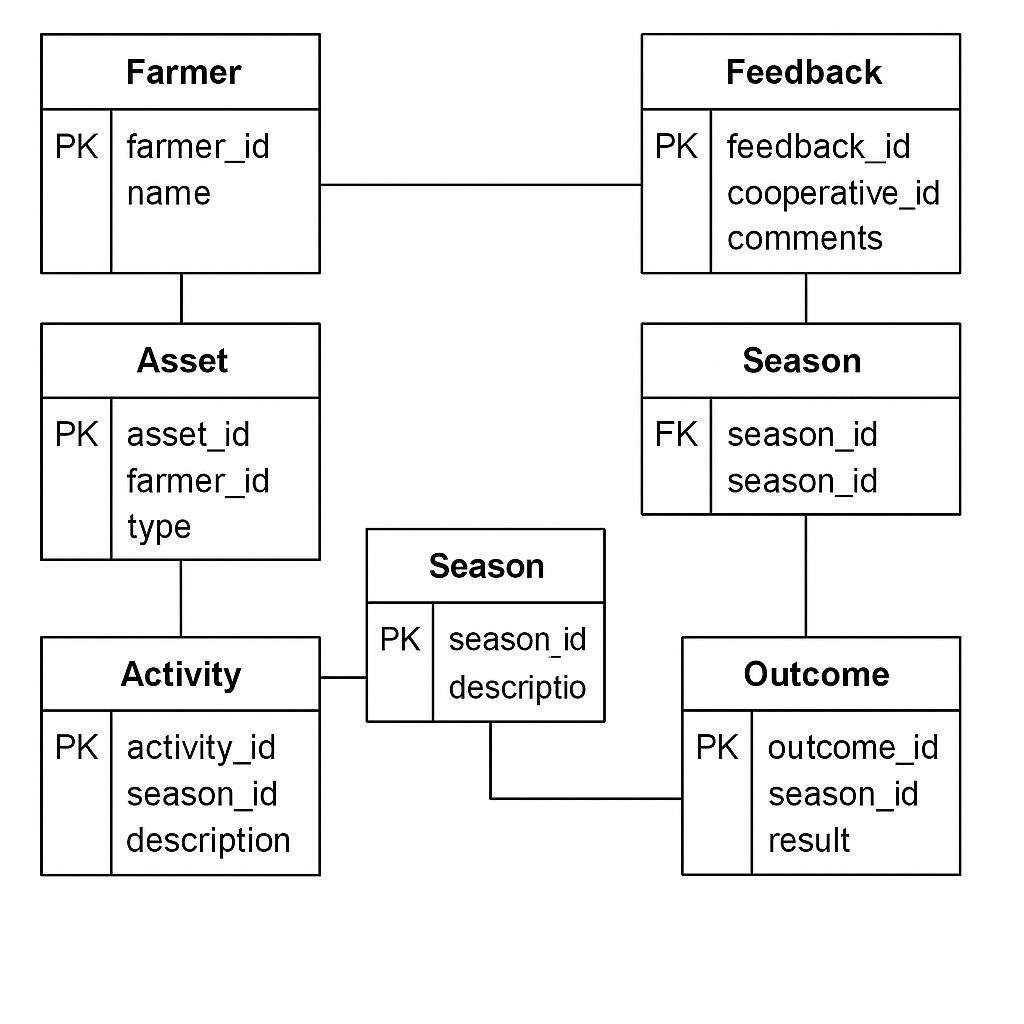
In this phase, design a detailed logical data model aligned with Phase I problem statement and Phase II process model. Your tasks are:

1. Entity-Relationship (ER) Model: identify entities, define attributes with data types, and specify PKs/FKs.
2. Relationships & Constraints: document relationships (1:1, 1:M, M:N) and apply NOT NULL, UNIQUE, CHECK, DEFAULT constraints.
3. Normalization: eliminate redundancy and ensure at least 3rd Normal Form (3NF).
4. Handling Data Scenarios: demonstrate the model supports various real-world scenarios effectively.
5. Presentation & Feedback: present the logical model for review; ensure it is well-documented and readable.

## 1. Entity-Relationship (ER) Model

Define all entities relevant to the Agricultural Management System with their attributes, data types, and keys:

ER Diagram:



### Farmer

|  |  |  |
| --- | --- | --- |
| Attribute | Data Type | Constraints |
| farmer\_id | INT | Primary Key, NOT NULL |
| name | VARCHAR(100) | NOT NULL |
| contact | VARCHAR(100) | NULL |

### Cooperative

|  |  |  |
| --- | --- | --- |
| Attribute | Data Type | Constraints |
| coop\_id | INT | Primary Key, NOT NULL |
| name | VARCHAR(100) | NOT NULL |
| location | VARCHAR(100) | NULL |

### Asset

|  |  |  |
| --- | --- | --- |
| Attribute | Data Type | Constraints |
| asset\_id | INT | Primary Key, NOT NULL |
| farmer\_id | INT | FK → Farmer(farmer\_id), NOT NULL |
| type | VARCHAR(20) | NOT NULL |
| subtype | VARCHAR(50) | NOT NULL |

### Season

|  |  |  |
| --- | --- | --- |
| Attribute | Data Type | Constraints |
| season\_id | INT | Primary Key, NOT NULL |
| asset\_id | INT | FK → Asset(asset\_id), NOT NULL |
| start\_date | DATE | NOT NULL |
| expected\_yield | DECIMAL(10,2) | DEFAULT 0 |

### Activity

|  |  |  |
| --- | --- | --- |
| Attribute | Data Type | Constraints |
| activity\_id | INT | Primary Key, NOT NULL |
| season\_id | INT | FK → Season(season\_id), NOT NULL |
| activity\_type | VARCHAR(50) | NOT NULL |
| amount | DECIMAL(10,2) | NOT NULL |
| activity\_date | DATE | NOT NULL |

### Outcome

|  |  |  |
| --- | --- | --- |
| Attribute | Data Type | Constraints |
| outcome\_id | INT | Primary Key, NOT NULL |
| season\_id | INT | FK → Season(season\_id), UNIQUE, NOT NULL |
| actual\_yield | DECIMAL(10,2) | NOT NULL |
| harvest\_date | DATE | NOT NULL |

### Feedback

|  |  |  |
| --- | --- | --- |
| Attribute | Data Type | Constraints |
| feedback\_id | INT | Primary Key, NOT NULL |
| season\_id | INT | FK → Season(season\_id), NOT NULL |
| coop\_id | INT | FK → Cooperative(coop\_id), NOT NULL |
| feedback\_text | TEXT | NOT NULL |
| feedback\_date | DATE | NOT NULL |

## 2. Relationships & Constraints

* Farmer → Asset: One-to-Many
* Asset → Season: One-to-Many
* Season → Activity: One-to-Many
* Season → Outcome: One-to-One
* Season → Feedback: One-to-Many
* Cooperative → Feedback: One-to-Many
* Constraints:
* NOT NULL on all primary key and essential foreign key columns
* UNIQUE constraint on Outcome.season\_id
* CHECK(activity\_type IN ('Planting','Fertilization','Pest Control','Vaccination','Disease Control','Feeding','Health Check'))
* DEFAULT 0 for Season.expected\_yield

## 3. Normalization

* Ensure the model conforms to Third Normal Form (3NF):

1. 1NF: Atomic attributes; primary keys defined.
2. 2NF: Non-key attributes fully dependent on the primary key.
3. 3NF: No transitive dependencies; separate lookup entities.

## 4. Handling Data Scenarios

* Demonstrate handling of real-world scenarios:

1. Multiple assets per farmer: Handled via Asset table referencing Farmer
2. Concurrent seasons for different assets: Season table tied to each Asset
3. Diverse activity types: Enforced by CHECK constraint on Activity.activity\_type
4. Single outcome per season: Unique constraint on Outcome.season\_id
5. Multiple feedback entries per season: Feedback table allows multiple entries per season

## 5. Presentation & Feedback

The logical model will be presented for stakeholder review using the ER diagrams above. Feedback will be collected and incorporated before the physical design phase.