

Project Phase-3

Arnav Sharma:2023111033

Hardik Chadha:2023111031

Hiten Garg:2023101116

Virat Garg:2023101081

- **Relational Schema:**

The relational schema design follows the guidelines detailed in **Chapter 9**, which provides a systematic approach to entity-relationship mapping. Below is the methodology used to design the schema:

- 1. Entity Creation**

Each simple entity is created individually, and their respective attributes are documented in a tabular format for clarity.

- 2. Handling Relationships Based on Cardinality Ratios**

- a. Many-to-Many (M) Relationships:**

For M relationships, a separate table is created. This table includes the primary keys of the connecting entities as its attributes. For example:

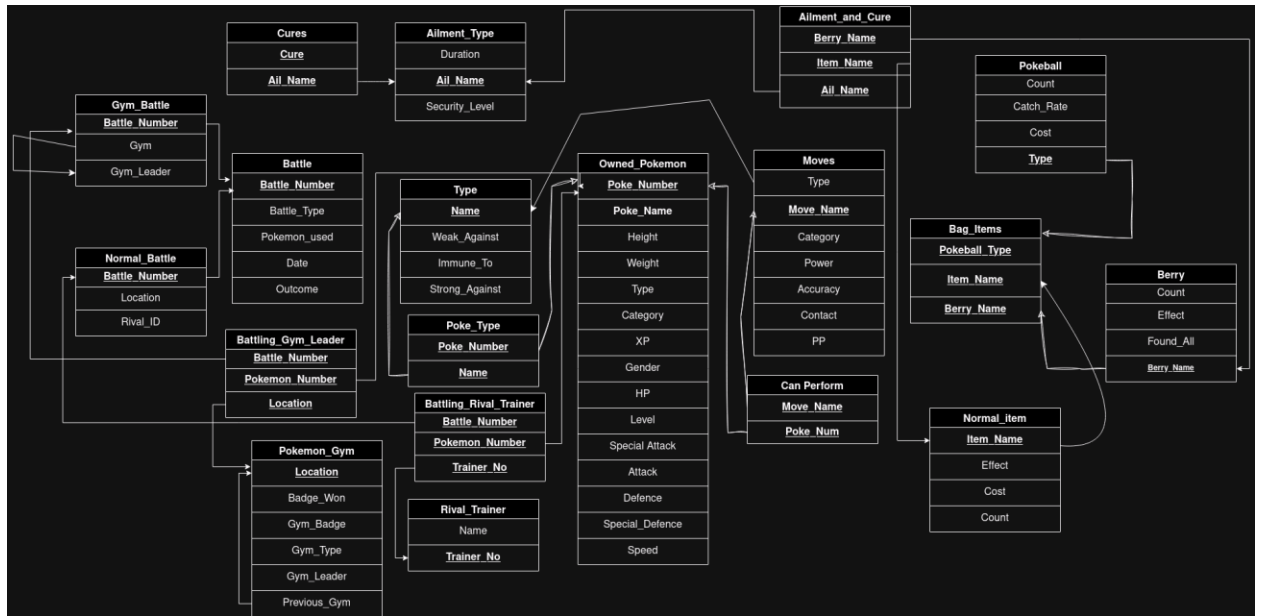
- i. Ailment_and_Cure
- ii. Battling_Rival_Trainer
- iii. Battling_Gym_Leader

These tables act as bridges to maintain the relationships between the corresponding entities.

- b. One-to-Many (1) Relationships:**

For 1 relationships, the foreign key from the "one" side is added as an attribute to the table on the "many" side. For instance:

For each multivalued attribute, a separate relation is created. This new relation includes the primary key of the original entity as a foreign key and the multivalued attribute as its own primary key. For example, in our case, the Cure attribute of the Ailment_Type entity is a multivalued attribute. Therefore, a separate relation named Cures is created to handle it appropriately.



- **2NF:**

For each relation, we examine its primary keys, specifically focusing on composite primary keys and any non-prime attributes. Since no partial dependency exists in my code, the structure in 2NF is identical to that in 1NF.

- **3NF:**

In 3NF, every non-prime attribute must depend solely on the primary key and nothing else. Since the Gym can be uniquely determined based on the Gym_Leader (assuming each Gym has exactly one fixed Gym_Leader), we created a separate relation named **Gym_Info** to represent this dependency. All other attributes appear to depend exclusively on their respective primary keys.

