

Jackson Schlosser

Professor McKee

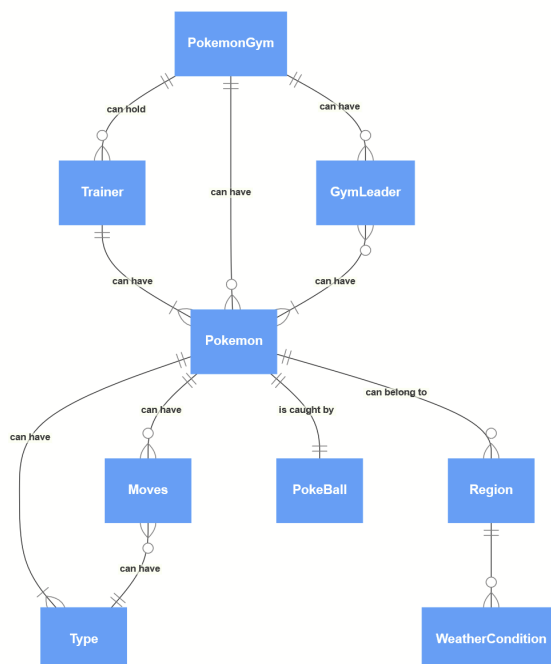
CMPT 308N

4/21/23

Project Evolution

Like many real-world projects, building this database was a slow process with lots of trial and error. At first, I chose to build a database in MySQL about an ice cream shop. I soon realized that the scope of that functional database would be too challenging for my first attempt at creating something like this. I then chose something I'm very familiar with, Pokemon. A Pokemon database would have everything I need to show a fully functional database with about ten entities. At first, I chose Pokemon, PokemonType, PokemonMove, Trainer, PokemonTrainer, PokemonRegion, GymLeader, PokemonGym, WeatherCondition, and Pokeball for my entities. Figure 1 shows my first attempt at creating an Entity-Relationship Diagram (ERD) which hardly resembles an ERD.

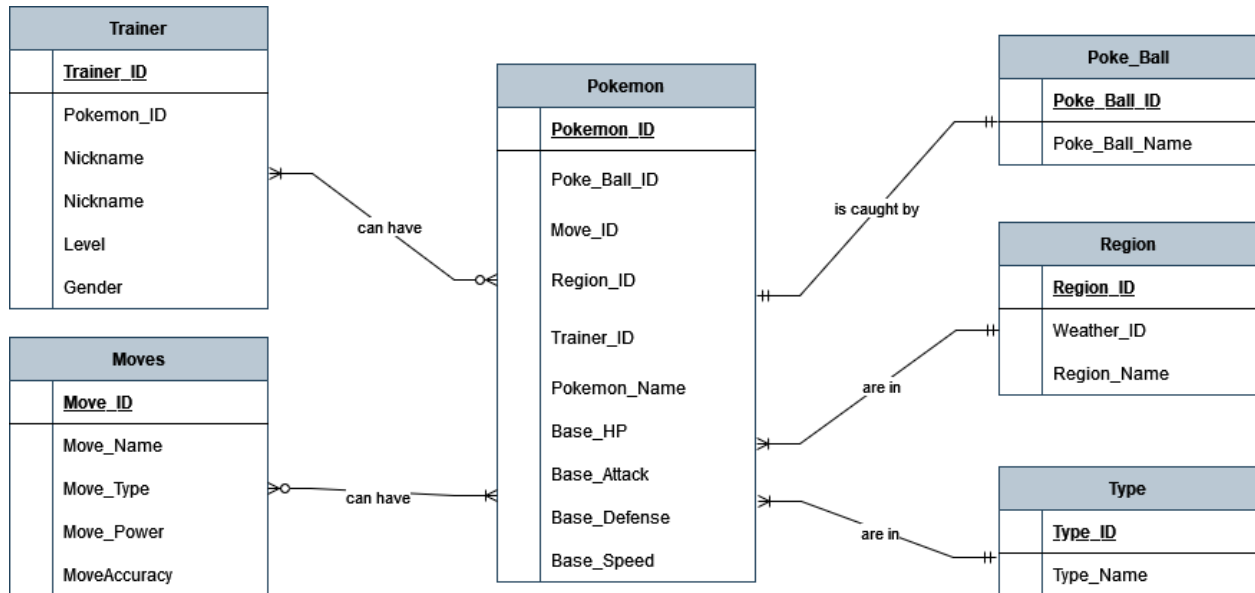
Figure 1



At this point I was still struggling with relationships, primarily many-to-many relationships.

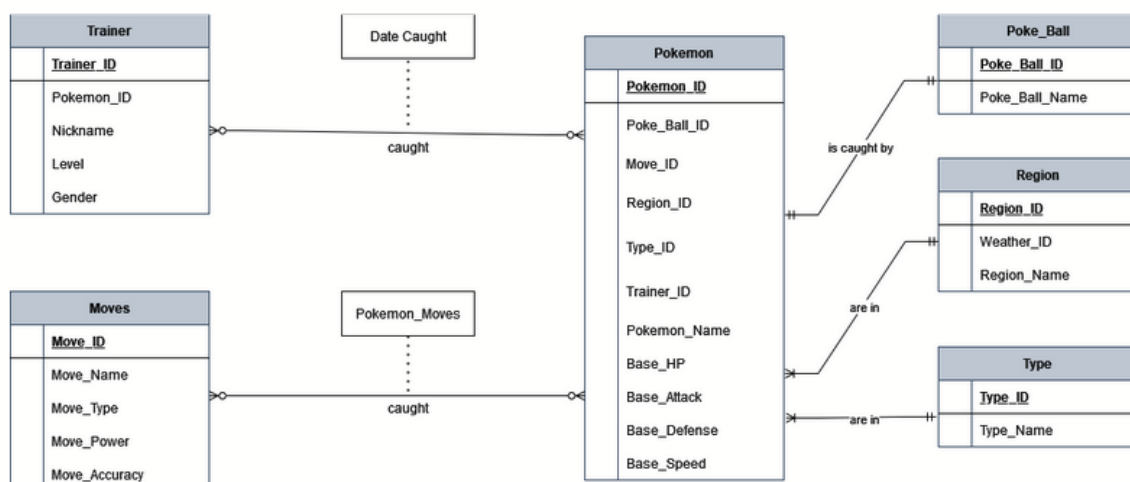
After some trial and error, I was able to create a better diagram with less entities, and an overall simpler design. See figure 2.

Figure 2



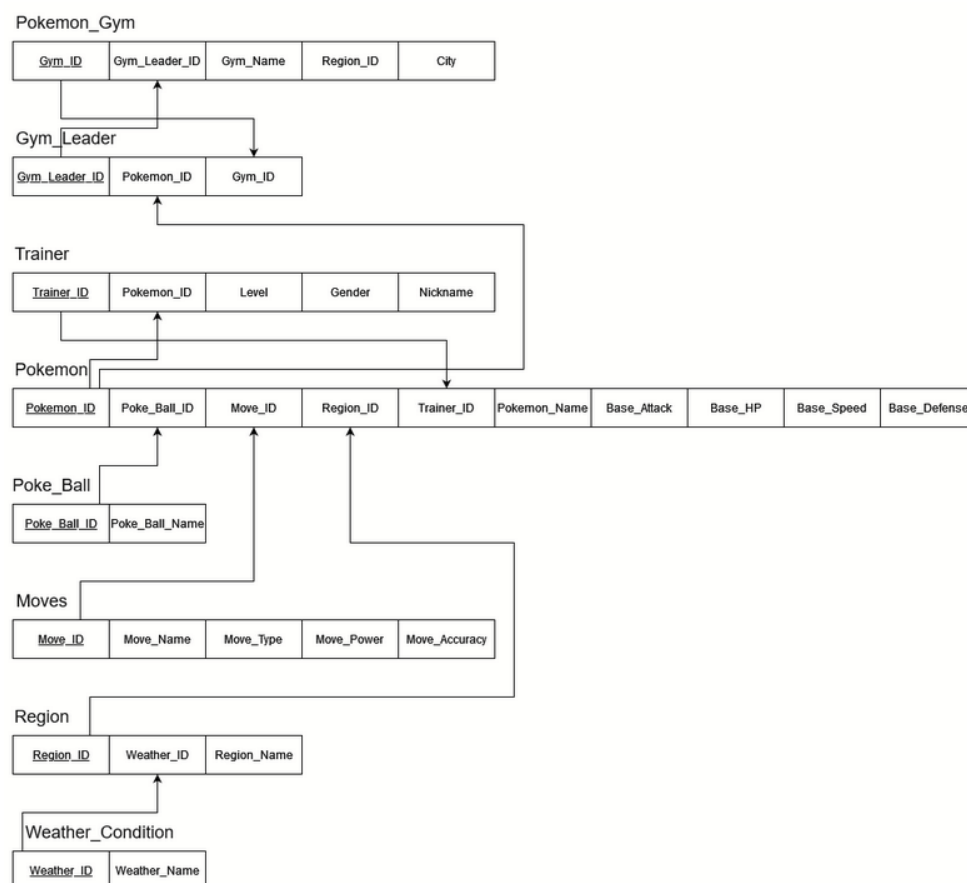
I was headed in the right direction, but I was still missing join tables for my many-to-many relationships. Once I was able to implement join tables for these relations, I came up with my final Entity-Relation Diagram. See Figure 3.

Figure 3



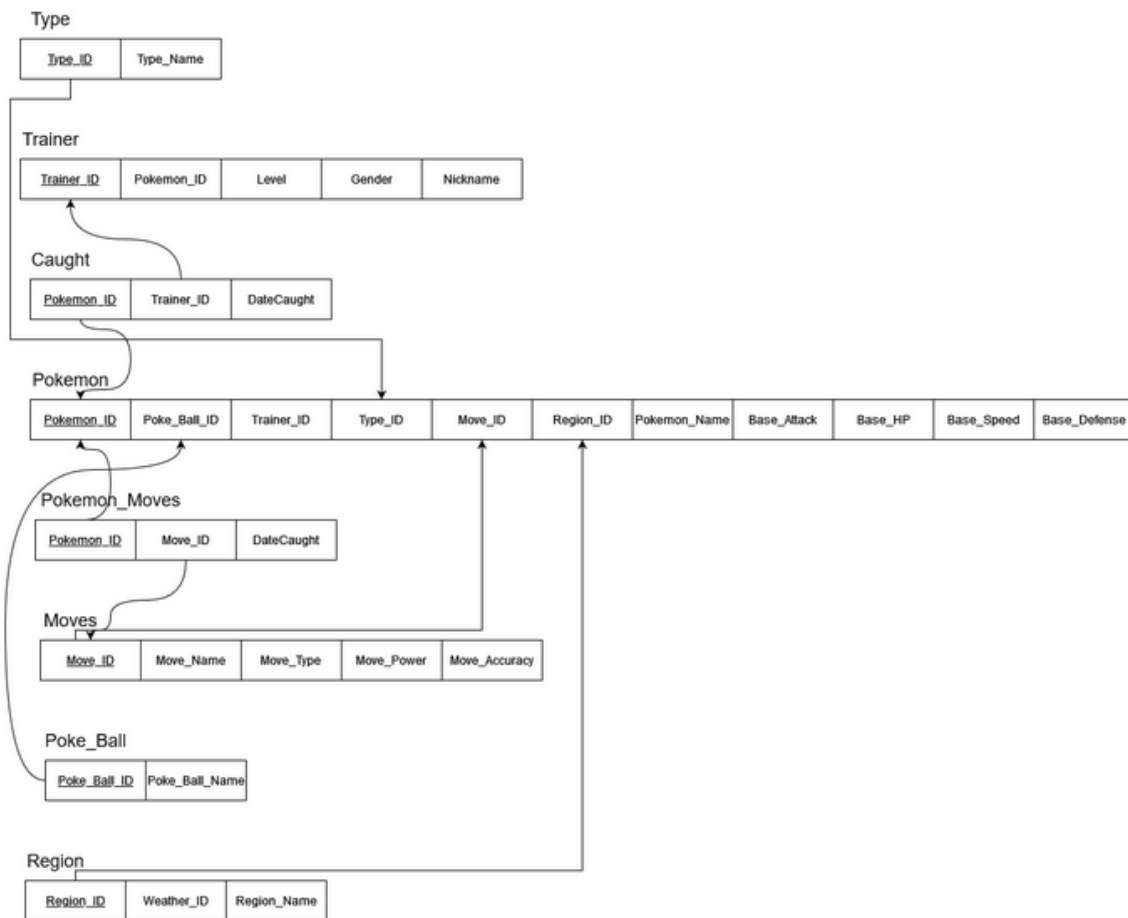
The ERD process was by far the most challenging part of this project. At first, it was difficult to understand how everything was related with so many confusing entities and attributes. The next step was to create a relational schema showing referential integrity with all relations in the third normal form. My first attempt wasn't terrible, but it was cluttered and a little confusing. See figure 4.

Figure 4



This diagram successfully shows referential integrity and all relations are in the third normal form, but I decided to cut a few of the entities to make it simpler. This led me to my last relational schema, see figure 5.

Figure 5



After going through the challenging process of creating the two diagrams, I had a good understanding of the database as a whole. This enlightenment made it fairly easy to write the SQL code. First, I started with CREATE TABLE statements to define a table for each of my entities. I then added INSERT INTO statements with at least four entries. Lastly, I added SQL queries such as INNER JOIN, OUTER JOIN, subquery, and correlated subquery to perform specific actions with my database. Surprisingly, the SQL code was the easiest part of this project for me, maybe it's because I understood the assignment by this point. In conclusion, the development of this functional and efficient database required a significant amount of time and

effort, and I faced numerous challenges along the way. Nonetheless, I ultimately produced a reliable and functional database that met the requirements of the project.