**Pokemon Catch Database**

Lu Zhang – CS340 Project

**Outline**

The database is a model for Pokemon Go, a mobile game by Nintendo. In particular this Pokemon Catch Database will be used to track players’s catching activities, including Pokemons they caught, time/location of the catching happens.

One possible goal of this game is to have players (“trainers”) catch all the Pokemon monsters and pit them through different locations. The Pokemon universe is filled with various locations where Pokemons can be caught by trainers. In addition, trainers are scattered throughout different locations in the universe.

**Database Outline**

Our database has five main entities: Pokemon, Pokemon\_Type, Catch, Player, and Location. The Pokemon table holds rows of Pokemon, which have a Pokemon\_id, pokemon\_name. Each Pokemon must have a Pokemon\_id and pokemon\_name. The primary key is pokemon\_id, and it is auto\_incrementing. Pokemon\_id and pokemon\_name are required. An optional description may be included. Each ID must be assigned to a pokemon name. Each pokemon name is unique.

Pokemon\_Type table contains information about pokemon type, including 18 types: Poison, water, normal, bug, ghost, fairy, fighting, electric, dragon, grass, ice, ground, flying, psychic, dark, fire, rock, steel. Each pokemon name must have and only have one type associated with when they are caught by trainers. Pokemon\_Type table has pokemon\_type, pokemon\_name.

Player table contains basic player information, player\_id, first\_name, last\_name. Player\_id is primary key, and auto-incrementing. Combination of first\_name and last\_name must be unique. Player\_id, first\_name, last\_name are required.

Location table contains location info where pokemon are caught by players. Each location has an ID that is auto-incrementing, and a location name which must be unique. The primary key is the location\_id. The location\_id and location\_name cannot be null.

Catch table tracks event record when players catch a pokemon at some location. Each catch row contains Catch\_time, Player\_id, Location\_id, Pokemon\_id. Player\_id, location\_id and pokemon\_id are foreign keys referencing the primary keys in player table, location table and pokemon table.

So one relationship is the one-to-one relationship between pokemon and pokemon\_type, where each pokemon name must and only has one type associated with.

Another relationship is the many-to-many relationship between player and location. Each player can catch different pokemons at various locations. Each location where pokemons can be caught by various players. This relationship is represented by a separate table named Catch where each row has a player\_id, location\_id, catch\_time, pokemon\_id; the combination of player\_id and location\_id makes up the primary key. Catch table works as a ternary relationship link Pokemon table, location table and player table together.

One-to-many relationship bewteen pokemon and player, where one pokemon can be only caught by one player, one player can catch multiple pokemons.

One-to-many relationship between pokemon and location, where one pokemon can be only caught on one location. But at one location player can encounter many pokemons.

**ER Diagram**

**Database Schema**