

Nicole Dávila Hernández

A01784217

Construcción de software y toma de decisiones (Gpo 501)

Esteban Castillo Juarez

3. Ejercicio de Modelación de Base de Datos Pokemon TCG (Justificación de tablas)

En el modelo entidad-relación de UML que acompaña este documento, he incluido las siguientes tablas:

- *users*
- *gameboard*
- *bench*
- *cards*
- *card_attributes*
- *status_effects*
- *card_status_effects*
- *active_pokemon*
- *discard_pile*
- *evolutions*
- *deck*
- *cards_deck*
- *prize_cards*

Cabe mencionar que algunas mecánicas del juego, como lo pueden ser aquellos momentos donde se requiera tirar una moneda al azar, cómo ganar / perder el juego, etc., se podrían programar directamente y sin necesidad de asignarles una tabla.

Tomando esto en cuenta, comenzamos con la tabla *users*.

Esta tabla contiene los siguientes atributos, donde '*(PK)*' denota que esta es la restricción de llave primaria de la tabla:

- *user_id (PK)*
- *username*
- *email*
- *password*

Esta tabla incluye atributos clave para identificar correctamente a los jugadores. El *user_id* será único y exclusivo para cada jugador. Me tomé la libertad de no incluir aspectos como la personalización del jugador (como lo es la vestimenta), ya que esto no es fundamental para el

juego como lo puede ser otra información del usuario (nombre de usuario, email / contraseña, etc.). Esta tabla existe simplemente porque este juego requiere de usuarios para funcionar.

La siguiente tabla que tenemos es *gameboard*.

Esta tabla principalmente funciona para denominar los jugadores, el juego en el que se encuentran y llevar cuenta del turno de cada jugador. Su llave primaria es un ID del juego y sus llaves secundarias se componen de:

- *user1_id* – en la cual se encontrará un *user_id* de quien se haya puesto como jugador 1.
- *user2_id* – en la cual se encontrará el *user_id* de quien se haya puesto como jugador 2.

Siguiendo con la tabla *bench*, esta tabla nos permite asignar una banca a cada jugador en un cierto juego, que incluya las cartas con las que el jugador pueda interactuar con el oponente. La relación entre *gameboard* y *bench* será de 1:2, puesto que por cada juego solo que existe un tablero, y por cada tablero pueden existir únicamente 2 bancas para 2 jugadores.

En las bancas, deben existir cartas, por lo que existe la tabla *cards*. Esta tabla incluye:

- *card_id* (PK)
- *card_name*
- *card_attribute*
- *card_description*
- *card_art*
- *visibility*

En esta tabla, principalmente se encuentra información acerca de la carta, como lo es su nombre, descripción, imagen, etc. Sin embargo, algunos atributos importantes de mencionar son *card_id*, *card_attribute* y *visibility*. *card_id* es el identificador único de la carta que servirá como llave foránea en muchas otras tablas.

visibility funcionará como un valor booleano que nos dirá si la carta permanece volteada (con valor 0) o si es revelada (con valor 1) dependiendo de la mecánica del juego y tomando en cuenta que este valor puede variar según las circunstancias.

Por último, *card_attribute* será lo que define a nuestra carta debido a que este atributo definirá su habilidad en el juego. Sabiendo que existen cartas tipo *trainer*, de energía y de pokémon, decidí crear una tabla extra que vincula una carta a la acción que realizará en el juego. Es decir, la tabla *cards* se relaciona directamente con otra tabla llamada *card_attributes*. *card_attributes* tendrá un nombre del atributo y un valor del atributo.

Por ejemplo, si existe una carta con un Pikachu, esta tendrá su tabla en *cards*, donde su *card_attribute* podría ser ‘pokémon’, y en la tabla *card_attributes*, *attribute_name* tendrá como valor ‘Thunder Shock’ y en *attribute_value* tendrá ‘20’. Así, sucesivamente se adapta a las cartas tipo *trainer* y de energía.

Estas cartas, también pueden ser afectadas por, vaya la redundancia, ciertos efectos que infringen algunas cartas, como los son los efectos de Confundido, Dormido, Envenenado, Paralizado y Quemado. Es por lo que existe la tabla *status_effects*, con la cual se identifican las cartas afectadas por ese efecto, y la cual se vincula con *cards_status_effects* para saber cuál es y qué hace el efecto que perjudica a la carta.

Por supuesto, las estas cartas pueden pertenecer a un mazo, por lo que tenemos la tabla *deck*, la cual incluye el ID del mazo como llave principal, el ID del usuario como llave foránea y el nombre del mazo. Esta tabla entonces se relaciona con *cards_deck*, y define las cartas que contiene el mazo así como la cantidad de cartas que existen en este.

Regresando a la banca y a cómo interactúan los jugadores, encontramos la tabla de *active_pokemon*, *discard_pile* y *prize_cards*. Como se sabe, existen varias formas de ganar el juego. Desde acabarse las cartas del mazo de cartas de premio hasta dejar al oponente sin pokémons en su banca, las tablas antes mencionadas forman una parte fundamental de la mecánica de Pokémon TCG. La tabla *active_pokemon* determinará el pokémon en juego del jugador, *discard_pile* tendrá las cartas descartadas del jugador y que podrían ser útiles después para el mismo. Por último, las tarjetas de premio que se encuentran en *prize_cards* juegan un gran rol para el jugador, ya que determinan si pierde o gana y le dan ciertas habilidades.

Como última tabla existe *evolutions*. Las evoluciones en Pokémon TCG son de alta relevancia ya que estos pueden darle al jugador una gran ventaja al jugar. Permiten evolucionar a los diferentes pokémones que pueda tener el jugador e infringir más daño a la otra carta activa del oponente. Como cada evolución es diferente, esta tabla llevará el control de la carta base y a qué evoluciona.

Con esto, se justifica la necesidad de incluir cada una de estas tablas en el modelado del juego Pokémon TCG, así sufriendo mecánicas cruciales como los jugadores, las cartas, los mazos, las habilidades y efectos de cada carta, etc.