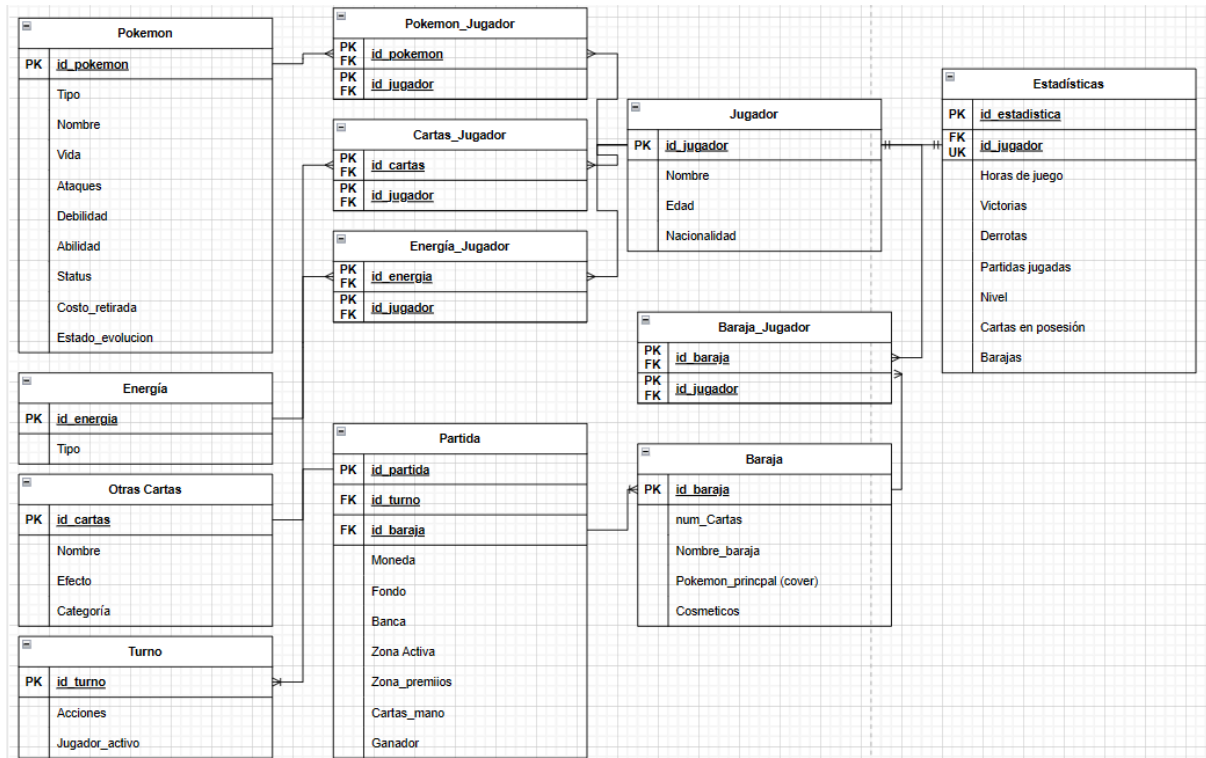


## Modelo Entidad Relación de Pokemon TCG



### Jugador:

- **Atributos:** Los atributos de esta clase están hechos para saber información básica del jugador, el nombre (para poder mostrarlo en las partidas), la edad y la nacionalidad (que puede ayudar con el idioma).
- **Restricciones de integridad:** Tiene su llave principal, en la llave principal se pondrá el auto-incremento, y nombre tendrá la restricción de not null, para que a fuerza cada jugador tenga un nombre.
- **Relaciones:** Tiene relaciones con las tablas Pokemon, Otras Cartas y Baraja, con todas hay una tabla intermedia ya que se trata de una relación de muchos a muchos.
- **Uso:** Todo videojuego requiere de jugadores, por lo que esta tabla es la base de este modelo entidad-relación.

### Pokemon:

- **Atributos:** Estos atributos ayudan a tener la información de cada carta que tiene un pokémon, para así poder usarlas en las partidas.
- **Restricciones de integridad:** Tiene su llave principal para poder identificar cada carta, la llave principal tendrá el auto-incremento y tipo, vía, nombre, costo\_retirada y estado\_evolucion tendrán el not null, ya que cada pokémon debe tener mínimo esa información.
- **Relaciones:** Tiene una relación con la tabla Jugador, pero ya que es de muchos a muchos, se hace una tabla intermedia llamada Pokemon\_Jugador (Ya que muchos jugadores pueden tener varios pokemon).

- **Uso:** La principal forma de atacar y no perder es usando pokémon, ya sea para hacer daño o para aguantar un golpe, por esta razones esta tabla es imprescindible para el modelo ER.

#### Otras Cartas:

- **Atributos:** Estos atributos están hechos para los otros tipos de cartas (herramientas, escenarios, soportes/apoyos/entrenadores). Con estas cartas solo se necesita saber su nombre, su categoría, y su efecto en una partida.
- **Restricciones de integridad:** Tiene su llave primaria para identificar cada carta, esa llave tendrá un auto-incremento.
- **Relaciones:** Tiene una relación con la tabla Jugador, pero ya que es de muchos a muchos, se hace una tabla intermedia llamada Cartas\_Jugador (Ya que muchos jugadores pueden tener muchas cartas que no sean pokemon).
- **Uso:** Hay muchas cartas que se comportan diferente que los pokémon, por lo que tener esta tabla es necesario para poder representar las partidas de este videojuego de la manera adecuada.

#### Energía:

- **Atributos:** El único atributo que necesita es el tipo, ya que dependiendo del tipo de energía es la cantidad de pokemons que pueden afectar.
- **Restricciones de integridad:** Tiene su llave primaria para identificar cada carta de energía. Esta llave tendrá un auto-incremento.
- **Relaciones:** Tiene una relación con la tabla Jugador, pero ya que es de muchos a muchos, se hace una tabla intermedia llamada Energía\_Jugador (Ya que muchos jugadores pueden tener muchas cartas de energía).
- **Uso:** La mayoría de pokémon requieren de energía para atacar, por lo que tener esta tabla es necesario para poder representar las partidas de este videojuego de la manera adecuada.

#### Turno:

- **Atributos:** Estos atributos están hechos para marcar el orden de una partida, por eso se tiene el jugador que va a estar activo en el turno y las acciones que hará en ese turno.
- **Restricciones de integridad:** Tiene su llave primaria para poder identificar cada turno que tendrá un auto-incremento y jugador activo tendrá la propiedad de not null porque siempre tiene que ser el turno de algún jugador.
- **Relaciones:** Tiene una relación con la tabla Partida, ya que una partida puede tener varios turnos.
- **Uso:** Una mecánica clave de cualquier juego pokemon son los turnos, por lo que esta tabla también es muy importante para este modelo ER.

#### Partida:

- **Atributos:** Estos atributos son para saber los elementos de una partida entre dos jugadores, como las mecánicas base (la moneda, el área de premios, la banca, la zona activa, etc), el objetivo (tener un ganador).
- **Restricciones de integridad:** Tiene su llave primaria para identificar cada partida, aparte de eso tiene dos llaves foráneas con turno y baraja ya que tiene relaciones con estas. La llave primaria tendrá un auto-incremento.

- **Relaciones:** Esta tabla tiene una relación con la tabla Baraja y Turno, ya que la baraja de 2 jugadores conforman una partida y una partida debe tener más de 1 turno.
- **Uso:** Para poder usar las cartas y no solo coleccionarlas, es necesario tener partidas, por lo que esta tabla también es necesaria para el modelo ER.

#### **Estadísticas:**

- **Atributos:** Estos atributos son para guardar información del jugador y su experiencia en el juego como el tiempo jugado, su historial de partidas, su nivel de cuenta, o incluso la cantidad de cartas que tiene en posesión.
- **Restricciones de integridad:** Tiene su llave primaria para identificar las estadísticas de cada jugador, y tiene una llave foránea para poder relacionarse con el jugador. La llave primaria tendrá un auto-incremento.
- **Relaciones:** Tiene una relación con la tabla Jugador que es de 1 a 1 (Un jugador tiene un set de estadísticas).
- **Uso:** El uso de estadísticas puede ayudar mucho a los desarrolladores del juego para ver el estado de su videojuego, aparte que un jugador siempre aprecia poder ver su experiencia en el juego, por lo que esta tabla también es muy importante para el modelo ER.