



## Fundamentación

El diseño parte de la tabla **Player**, la cual incluye todos los campos previamente mencionados. Esta tabla se identifica por una **PK** (clave primaria) de tipo **INT autoincremental**. Todos los campos se definen como **NOT NULL**, incluso en situaciones en las que un jugador no tenga registros, como por ejemplo, cero victorias, ya que en estos casos se almacenará un valor predeterminado como el 0.

La tabla **Player** establece relaciones con varias otras entidades del sistema:

- Con **Friendship** en una relación **1 a N**, ya que un jugador puede tener muchas amistades.

- Con **Game** en una relación **N a M**, porque un juego siempre involucra dos jugadores, y un jugador puede participar en múltiples juegos.
- Con **Store** también en una relación **N a M**, dado que un jugador puede comprar en distintas tiendas y una tienda puede vender a muchos jugadores.
- Con **Deck** en una relación **1 a N**, pues un jugador puede tener varios mazos y cada mazo pertenece exclusivamente a un jugador. Aunque puedan existir mazos similares entre jugadores, la información será propia de cada uno.

La tabla **Friendship** requiere **2 claves foráneas (FK)** provenientes de **Player**, correspondientes a los jugadores involucrados en la relación. Ambos campos deben ser **NOT NULL**, ya que no puede existir una amistad sin dos participantes.

En cuanto a **Game**, esta tabla tiene una **clave primaria INT autoincremental**, además de **2 claves foráneas de Player**, ambas obligatorias, ya que una partida necesita dos jugadores. También contiene un campo para el **ganador**, el cual es otra **FK de Player** y debe ser **NOT NULL**.

La entidad **Deck** incluye una **clave primaria INT autoincremental** y una **clave foránea de Player**, ya que cada mazo le pertenece a un jugador. Adicionalmente, incorpora una variable booleana llamada “**active**” que indica si ese mazo se encuentra actualmente en uso durante una partida.

Para vincular las cartas con los mazos, se utiliza una tabla intermedia llamada **Deck\_Card**, la cual contiene **2 claves foráneas**, una que apunta a **Deck** y otra a **Card**. Esta estructura permite saber qué cartas componen cada mazo.

La tabla **Card** está compuesta por una **PK INT autoincremental**. Cada carta cuenta con un **nombre** (basado en Pokémon), una **descripción**, y un **tipo**. Los campos adicionales dependerán del tipo de carta, razón por la cual se permiten valores **DEFAULT NULL** en esos atributos.

Por otro lado, la tabla **Store** tiene una **clave primaria INT autoincremental** y un campo que define el tipo de moneda aceptada (**typeCoin**). Esta tabla se relaciona con **Player**, como se mencionó antes, y también con **Product** en una relación **1 a N**, ya que una tienda puede ofrecer múltiples productos.

Por último, la entidad **Product** posee una **clave primaria INT autoincremental** y una **clave foránea proveniente de Store**, indicando a qué tienda pertenece cada producto. Todos sus campos deben ser **NOT NULL**.