

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
FACULTAD DE CIENCIAS, 2025-II  
FUNDAMENTOS DE BASES DE DATOS



---

## TAREA 02:

### *Modelo Entidad – Relación*

---

NOMBRE DEL EQUIPO:  
NAMEisNULL

INTEGRANTES:

Flores Mata Ricardo - 422127808

Matute Cantón Sara Lorena - 319331622

Sánchez Cruz Norma Selene - 320198508

Suárez Ortiz Joshua Daniel - 320151260

Villegas Martínez Vania Victoria - 418003114

## 1. Conceptos del Modelo Entidad - Relación

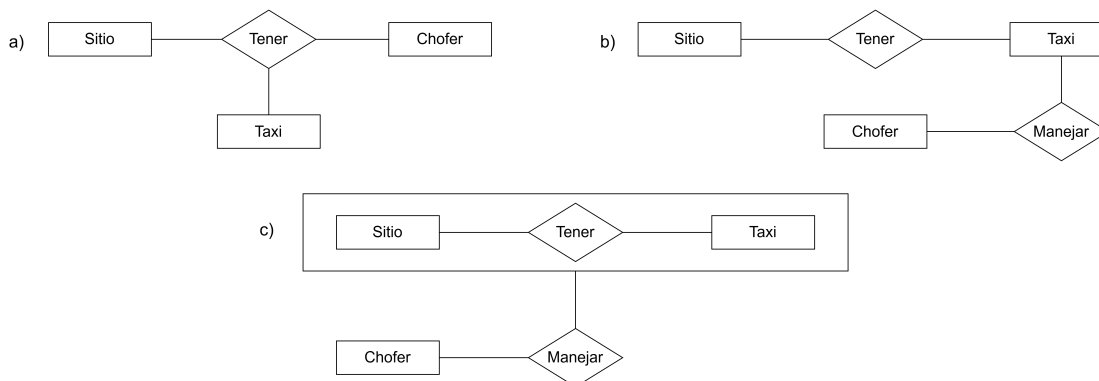
- a. ¿Qué es un **tipo de relación**? Explica las diferencias con respecto a una **instancia de relación**.  
Un **tipo de relación** en el modelo Entidad Relación es la asociación entre dos o más entidades, tal que se especifica como se relacionan estas entre si, en el contexto de nuestra base de datos. En cambio **instancia de relación** es un ejemplar específico de un tipo de relación. Donde podemos observar como se relacionan las instancias específicas de las entidades involucradas.
- b. ¿Bajo qué condiciones se puede **migrar un atributo** de algún **tipo de entidad** que participa en un **tipo de relación binaria** y convertirse en un **atributo** del tipo de relación? ¿Cuál sería en el efecto?  
Un atributo puede migrar de algún tipo de entidad que participa en un tipo de relación binaria y convertirse en un atributo del tipo de relación cuando dicho atributo no describe únicamente a su entidad origen, sino que define o describe un atributo de la interacción de las dos entidades que participan en la relación.  
El efecto de esta migración sería que el atributo deja de ser dependiente de una única entidad para ahora describir la relación entre ambas entidades.
- c. ¿Cuál es el significado de un **tipo de relación recursiva**? Proporciona un **par de ejemplos** de este tipo de relación.  
Un tipo de relación recursiva es cuando una entidad se relaciona consigo misma.
- Un ejemplo de esto podría ser la entidad usuario en una base de datos que modele una red social. Un usuario puede seguir a otro usuario, tal que la entidad usuario se relaciona consigo misma mediante la relación "seguir".
  - Otro ejemplo sería una base de datos que modele los vuelos posibles de un avión la cual tiene una entidad llamada ciudad. De una ciudad puedes ir a otra, tal que la entidad ciudad este relacionada consigo misma mediante la relación "ir".
- d. Responde a las **siguientes cuestiones**, deberás indicar **si son posibles o no, justificando** tu respuesta. Cuando **no sea posible** deberás **indicar alguna recomendación** al respecto:
- ¿Un **atributo compuesto** puede ser **llave**?  
**Si.** En particular existen las llaves compuestas, las cuales toman diversos atributos para garantizar que la llave sea única.
  - ¿Un **atributo multivaluado** puede ser **llave**?  
**No.** Esto debido a que por definición la llave es única para cada entidad y no puede contener múltiples valores para una misma instancia de entidad. Observemos que al ser un atributo multivaluado una instancia de entidad puede tener múltiples valores por lo cual no puede ser una llave.  
En este caso lo más sencillo sería usar una llave artificial o buscar otros atributos con los cuales hacer una llave compuesta.
  - ¿Un **atributo derivado** puede ser **llave**?  
**No.** Esto debido a que los atributos derivados son calculados a partir de otras atributos y carece de un valor almacenado. De igual manera al ser calculados puede variar el valor tal que se pueda dar en algún momento que comparta valor con algún otra instancia tal que ya no sea único.  
Dependiendo de donde vengan los atributos necesarios para calcular el atributo derivado; Se podría usar una llave compuesta o debilitar la clase y usar las llaves los otras entidades.
  - ¿Un **atributo multivaluado** puede ser **compuesto**? **Si.** Esto debido a que un atributo multivaluado puede tener mas de un valor por instancia de entidad y estos valores a su vez se pueden dividir en atributos simples.
  - ¿Un **atributo multivaluado** puede ser **derivado**?  
**Si.** El ser multivaluado significa que puede haber diversos valores para una misma instancia, por lo cual puede que esos diversos valores sean calculados de otros atributos los cuales a su vez pueden o no ser multivaluados.
  - ¿Qué implicaría la existencia de una **entidad** cuyos atributos sean **todos derivados**? Esto implica que no tiene algún valor almacenado directamente, sino que se obtienen y/o calculan de otros atributos. Por lo cual sus valores no son independientes y dependen de otros atributos los cuales pueden ser de una o de diversas entidades.
- e. Explica el concepto de **categorías (herencia múltiple)** en el **modelo E-R** y proporciona **dos ejemplos** de la **vida real** en donde se aplique este concepto.

En el modelo de entidad relación las categorías se refieren a la posibilidad de que una entidad herede o pertenezca a dos o mas superentidades.

- En la base de datos de digamos, la facultad de ciencias. Se puede tener que un estudiante igual sea un empleado. Por ejemplo, un ayudante es simultáneamente un estudiante y un empleado de la facultad.
- Supongamos que en la base de datos de un juguetería se tiene la superentidad peluche y una superentidad electrónico. Puede haber juguetes que sean tanto peluches como electrónicos.

## 2. Entendiendo el Modelo Entidad - Relación

- i. A continuación, se muestran tres representaciones posibles referidas a las relaciones entre Sitios de taxis, Choferes y Taxis. Analiza las ventajas y desventajas de cada propuesta, contestando las preguntas que se presentan a continuación:



- Indica qué **diagramas representan** la información requerida por las siguientes **solicitudes de información**:  
¿Qué **taxis** maneja el chofer **Raúl López** en el sitio **Santa Fe**?

El diagrama a responde directamente esta pregunta porque aquí solamente se registran las 3 entidades al mismo tiempo. El diagrama b también responde esta pregunta aunque es necesario primero consultar los taxis manejados por el chofer Raúl López y después consultar cuáles son los que pertenecen al sitio Santa Fe. El inciso c, por su parte, también responde la pregunta porque podemos consultar primero los taxis que tiene el sitio Santa Fe y de esos, cuáles son los que maneja el chofer Raúl López.

¿A qué **sitios** está afiliado el chofer **Carlos Reyes**?

El diagrama a responde esta pregunta pero con todos los taxis que maneja el chofer Carlos Reyes, lo que no es lo ideal. El diagrama b, por su parte, responde de una forma parecida dado que no hay una relación directa entre la entidad chofer y la entidad sitio. Finalmente, el diagrama c, también respondería con todos los taxis que maneja el chofer Carlos Reyes. Por lo que, mi conclusión es que ningún diagrama responde esta pregunta eficientemente.

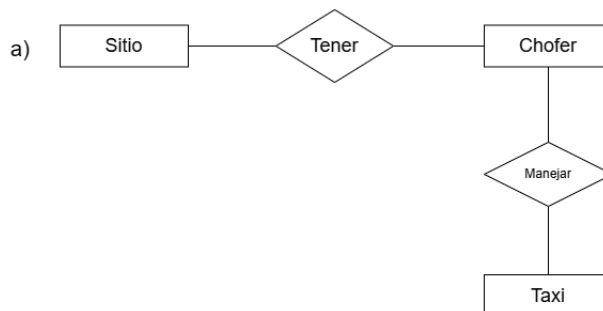
¿Qué **taxis** están asociados al sitio **Universidad**?

El diagrama a responde a esta pregunta pero con todos los choferes que están asociados a algún taxi del sitio universidad y no considera los taxis que no tienen chofer. El diagrama b responde esto directamente al igual que el diagrama c.

- ¿Qué **modificación harías** en el diagrama de la **figura a)**, sin perder información, para que se puedan conocer **qué taxis que maneja cada chofer**?

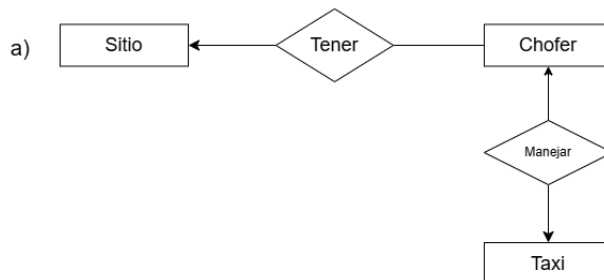
Quitaría la relación tener de la entidad Taxi y pondría otra relación *Manejar* entre las entidades Taxi y Chofer para poder tener la información más ordenada en tablas distintas.

Modificación para la pregunta sobre los taxis que maneja cada chofer

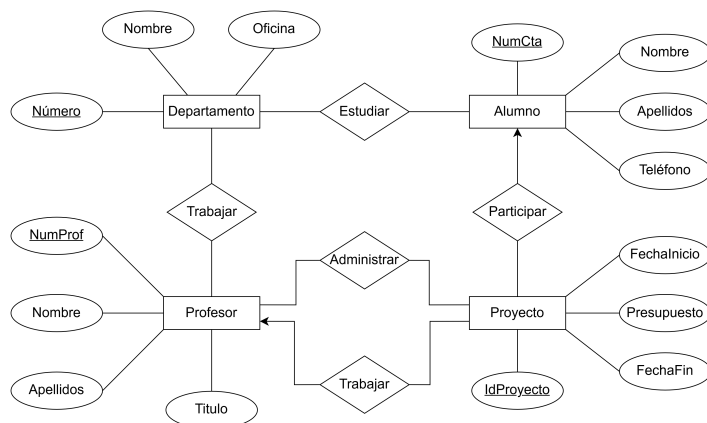


- ¿Qué **diferencia** existe entre los diagramas de las figuras **a)** y **c)**? Que el diagrama **a)** guarda todo a la vez, no permite registrar choferes asociados a sitios sin taxis, taxis asociados a choferes sin sitio y sitios sin choferes o taxis asociados. Mientras que el diagrama **c)** puede considerar taxis asociados a sitios sin que estrictamente se necesite un chofer para registrarlos.
- Cómo **modificarías** el diagrama de la **figura a)** para representar las **siguientes restricciones**:  
Un **chofer** no puede manejar más de un **taxi** en el mismo **sitio**. Un **taxi** no puede ser manejado por más de un **chofer** en el mismo **sitio**.

Modificación para el último inciso del ejercicio 2i

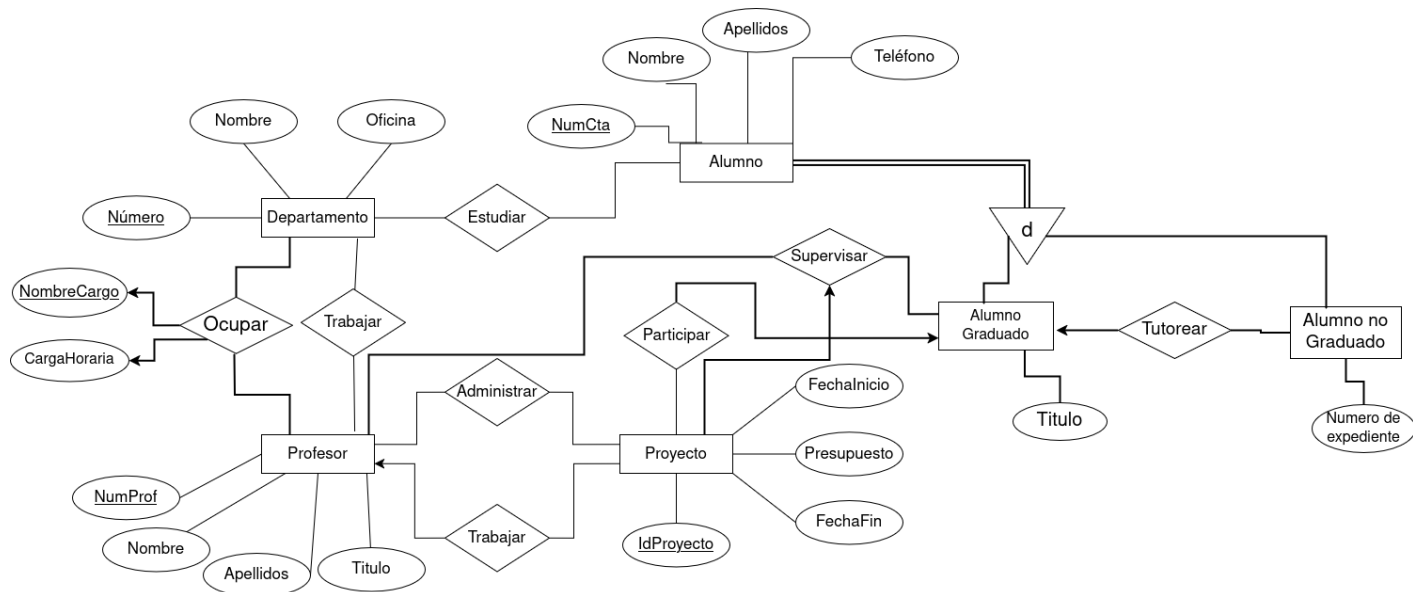


- ii. El siguiente modelo E-R corresponde a una base de datos de una universidad. Luego de unos años de funcionamiento, se han detectado una serie de deficiencias en el sistema de mantenimiento de datos y se quieren realizar las siguientes modificaciones:



- Dado que **solo los alumnos graduados** pueden **participar** en **proyectos**, se desea distinguir entre **alumnos graduados** y **no graduados**. Además de la información almacenada para un alumno, para los **alumnos graduados** se desea almacenar el **título que posee** y para los **alumnos no graduados** su **número de expediente**.
- Un **alumno graduado** puede ser tutor de varios **alumnos no graduados** y cada alumno **no graduado** tendrá **solamente un tutor**.
- Se desea almacenar, para **cada profesor**, el **nombre del cargo** que ocupa en cada **departamento** (el cual es único dentro del departamento) y la **carga horaria** asociada. Un **mismo cargo** puede tener **diferente carga horaria** dependiendo del departamento y/o del profesor. Dentro de **un departamento** no pueden existir **profesores** con el **mismo cargo**. Un **profesor** podrá tener el **mismo cargo** en **varios departamentos**.
- Cuando un **alumno graduado** participa en un **proyecto** y un **profesor** debe **supervisar** su trabajo en ese proyecto. Cada **alumno graduado** podrá trabajar en **múltiples proyectos**, en los cuales **podrá ser supervisado** por **diferentes profesores**.

Obtén un nuevo **modelo E-R** modificando el **modelo presentado**, para incorporar los **cambios deseados**. Identifica las restricciones de **cardinalidad**, **participación** e **identidad** en el nuevo modelo propuesto.



### 3. Mini – mundo, planteamiento a partir del modelo Entidad – Relación

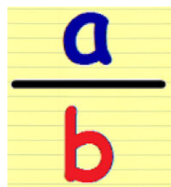
#### a. Números racionales

Diseña un **modelo E/R** para representar **números racionales**, bajo las siguientes consideraciones:

- Cada número racional está determinado por **dos números enteros**: **numerador** y **denominador**.
- Algunos de estos números son **racionales reducidos** de otros números racionales, por ejemplo,  $1/4$  es un **racional reducido** de  $6/24$ . Se cumple que **todo número racional tiene un único racional reducido** (considera que solo se denomina **racional reducido** a aquel que ya está **totalmente simplificado**, es decir,  $3/12$  no sería un **racional reducido**).
- Además de conocer el **racional reducido** asociado a cada fracción, se desea saber el **factor de reducción** asociado (en el caso anterior el **factor de reducción** sería 6).
- Dos **racionales** se consideran diferentes si tienen el **numerador** o **denominador** distintos, aunque correspondan a la misma fracción reducida.

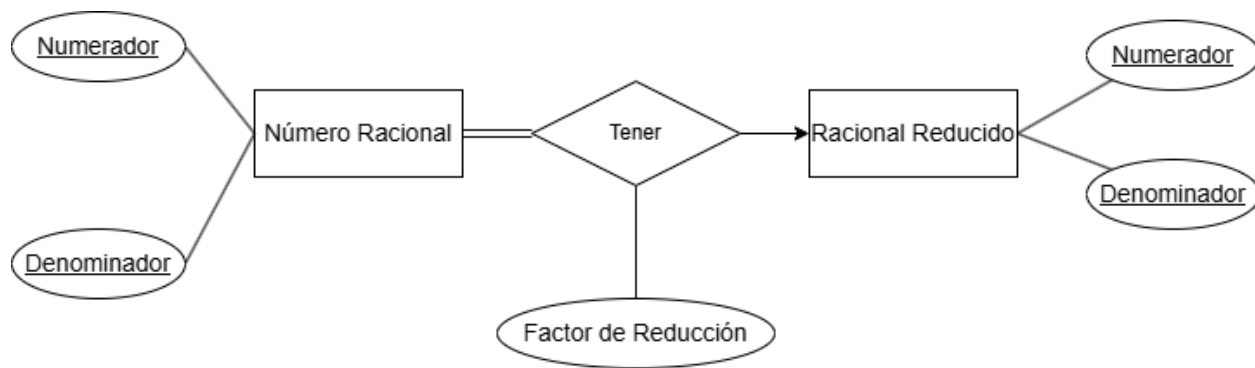
Primero los casos de uso subrayados:

Diseña un **modelo E/R** para representar **números racionales**, bajo las siguientes consideraciones:



- Cada **número racional** está determinado por **dos números enteros**: **numerador** y **denominador**.
- Algunos de estos números son **racionales reducidos** de otros números racionales, por ejemplo,  $1/4$  es un **racional reducido** de  $6/24$ . Se cumple que **todo número racional tiene un único racional reducido** (considera que solo se denomina **racional reducido** a aquel que ya está **totalmente simplificado**, es decir,  $3/12$  no sería un **racional reducido**).
- Además de conocer el **racional reducido** asociado a cada fracción, se desea saber el **factor de reducción asociado** (en el caso anterior el **factor de reducción** sería 6).
- Dos **racionales** se consideran diferentes si tienen el **numerador** o **denominador** distintos, aunque correspondan a la misma fracción reducida.

Aquí está el diagrama:



## b. Juego de computadora

### Código de colores:

- Entidades
- Atributos
- Relaciones
- Restricciones

Se desea construir una base de datos para modelar un juego de computadora. En el juego se utilizan conceptos como: **personajes**, **objetos** y **escenarios**. Las especificaciones del diseño son:

- Cada **personaje** tiene un **nombre único** y lleva asociados **5 valores** que indican: **vida**, **nivel de fuerza**, **inteligencia**, **habilidad** y **rapidez**. Interesa saber cada **momento** en qué **escenario** se **encuentra**. Cada **personaje** tiene un **rol** en el juego (p.e. enemigos, aliados, etc.) y **cada personaje puede dominar a otros personajes** (a cero o más), pero **sólo** puede ser dominado por **otro**, o por ninguno.
- Los **personajes** **tienen** en **cada momento** algunos **objetos** en su poder. Los **objetos** pueden incrementar alguna de sus características (p.e. pocimas para aumentar la fuerza) o bien puede ser armas. Cada **objeto** tiene un **nombre** y un **tipo**, además de un valor numérico que indica el **porcentaje** en que puede incrementar una característica o bien el daño en el nivel de vida que puede reducir.
- Además de saber qué objetos tiene cada **personaje** hay que guardar información acerca de en qué momento (**hora**, **minuto** y **segundo**) tomó el **personaje** el objeto. Puede haber **objetos** que no estén en posesión de ningún **personaje** (no los ha encontrado nadie aún), pero **ninguno puede pertenecer a dos personajes al mismo tiempo**.
- Los **objetos** se deben poder distinguir entre sí y están determinados por el **escenario** en que se encuentran situados al comienzo del juego. **Cada escenario puede contener 0 o más objetos**.
- Cada **escenario** es **único** y tiene 2 factores: **riesgo** y **tiempo**, mismos que indican características particulares del **escenario** en que el **jugador** se **ubique**.
- Se deben almacenar todas las estadísticas del juego para cada **jugador**, como, por ejemplo, **nombre del jugador**, **puntos alcanzados**, **nivel de avance**, **tiempo de juego**, y algún aspecto adicional que se

considere pertinente.

Podemos iniciar descartando a juego como entidad pues no estamos trabajando más de un juego por lo que, en este caso, no será una entidad a modelar. Por lo estaremos poniendo las siguientes entidades:

- Personaje
- Objeto
- Escenario
- Jugador
- Momento

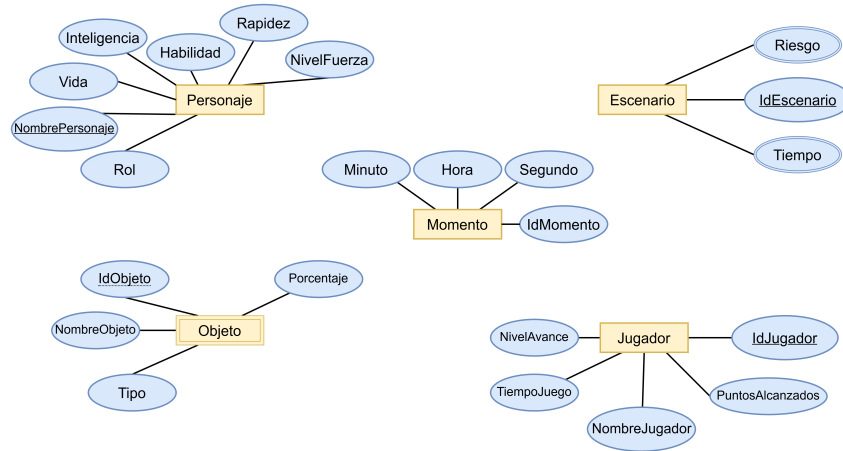
Inicialmente se contemplo a **Momento** como un atributo de la relación pero dado que es tanto para la de el personaje se encuentra en que escenario y la de los personajes poseen algunos momentos se clasifico como entidad.



Nos indica que **Personaje** tiene un **nombre único** por lo que si puede ser una llave y así la consideramos, como otros atributos podremos **vida, nivel de fuerza, inteligencia, habilidad y rapidez**, estos se quedaran como atributos simples. Por otro lado, tenemos que **Momento** tendra como atributos simples **Hora, minuto, segundo** como es solicitado pero aparte tendrá una llave a la que le pondremos **IdMomento** esto porque es una entidad. Para la entidad **Escenario** debido a que es único le pondremos una llave **IdEscenarios** y como otros atributos tendra **riesgo y tiempo** pero dado a que nos dicen que estos indican características particulares del escenario los podremos como multivaluados. Pasando a **Objeto** comencemos considerando que como nos dicen que estos están determinados por el **Escenario** en el que se encuentran al comenzar el juego nos habla de una entidad debil que depende de una entidad fuerte **Escenario**, dado a que los objetos se deben distinguir entre si podremos un discriminante **IdObjeto** pero debido a lo la entidad debil y como demás atributos su **nombre**, su **tipo** y dado a que indica que **porcentaje** es un valor numerico también lo dejaremos como un atributo simple. Y por último, **Jugador** no podemos poner **nombre** como llave pues no nos dicen que sea único y pueden existir problemas de que dos jugadores tengan el mismo nombre por lo que pondremo otra llave sustituta **IdJugador**, hablando de **nombre** se contemplo problemas en dejarlo como atributo simple pues no podemos asegurar que sea atomico pues puede haber casos donde si sea un solo nombre pero también pueden existir casos donde sea un nombre compuesto, y por lo mismo tambien se contemplo como un atributo compuesto pero no creo que sea tan util en este contexto por lo que al final se llevo a la conclusión de que fuera multivaludo, pero el ayudante indico que mejor se dejara como simple y solo se pudiera representar como una sola cadena, ya que multivaluado significaria que puede tener varios nombres a la vez y no se quiere eso, por lo cual nuestro razonamiento anterior estaba equivocado respecto a esto y ya como último tenemos a los tributos simples que serán **tiempo juego, nivel de avance, puntos alcanzados**. Todas estas consideraciones se plasman de la siguiente manera:

Teniendo lo anterior podemos pasar a las relaciones y restricciones, empecemos con que un personaje se **encuentra** en un escenario, aqui podemos ver varios personajes se pueden encuentran en un mismo escenario pero que un personaje se encuentre en varios escenarios en principio no tiene sentido porque un mismo personaje no puede estar en varios escenarios al mismo tiempo por lo que será de cardinalidad muchos a uno (m:1) y la participación será parcial. Ahora tenemos que un personaje puede **dominar** a otros personajes pero sólo puede ser dominado por otro, esto nos habla de que solo de un lado se cumple tanto de uno o muchos pero del otro solo se cumple con uno por lo que podemos decir que es de cardinalidad uno a muchos (1:m) y es tanto de participación total de un lado como parcial del otro. Dado a que los personajes tienen en cada momento algunos objetos en su poder podemos hacer esta relación ternaria y además como ningun objeto puede **pertenecer** a dos personajes a la vez será una cardinalidad de uno a muchos (1:m) pues un personaje si puede tener uno o más objetos pero un objeto no puede pertenecer a más de un personaje. Enfoquemonos en las relaciones faltantes de la entidad **Escenario** podemos verlo como

- Tenemos que un objeto solo puede estar en un escenario y varios escenarios pueden estar en un mismo escenario, pero un objeto no puede estar en varios escenarios, por lo que es una relacion de mucho a uno (m:1).



- Ocurre lo mismo con la relación entre **Escenario** y **Jugador** pues un jugador puede estar solo en un escenario y varios jugadores pueden estar en un solo escenario a la vez pero un jugador no puede estar en varios escenarios a la vez.

Todo lo anterior nos dio como resultado el siguiente modelo entidad - relación:

