



# Tecnológico de Monterrey

**Campus Santa Fe**

Ejercicio de creación de una base de datos relacional en MySQL para el reto

Construcción de software y toma de decisiones

Grupo 501

Sylvia Fernanda Colomo Fuente A01781983

Samantha Covarrubias Figueroa A01026174

Cristina Alessandra González A01025667

Valeria Martínez Martínez A01782413

Mauricio Tumalan Castillo A01369288

Profesores:

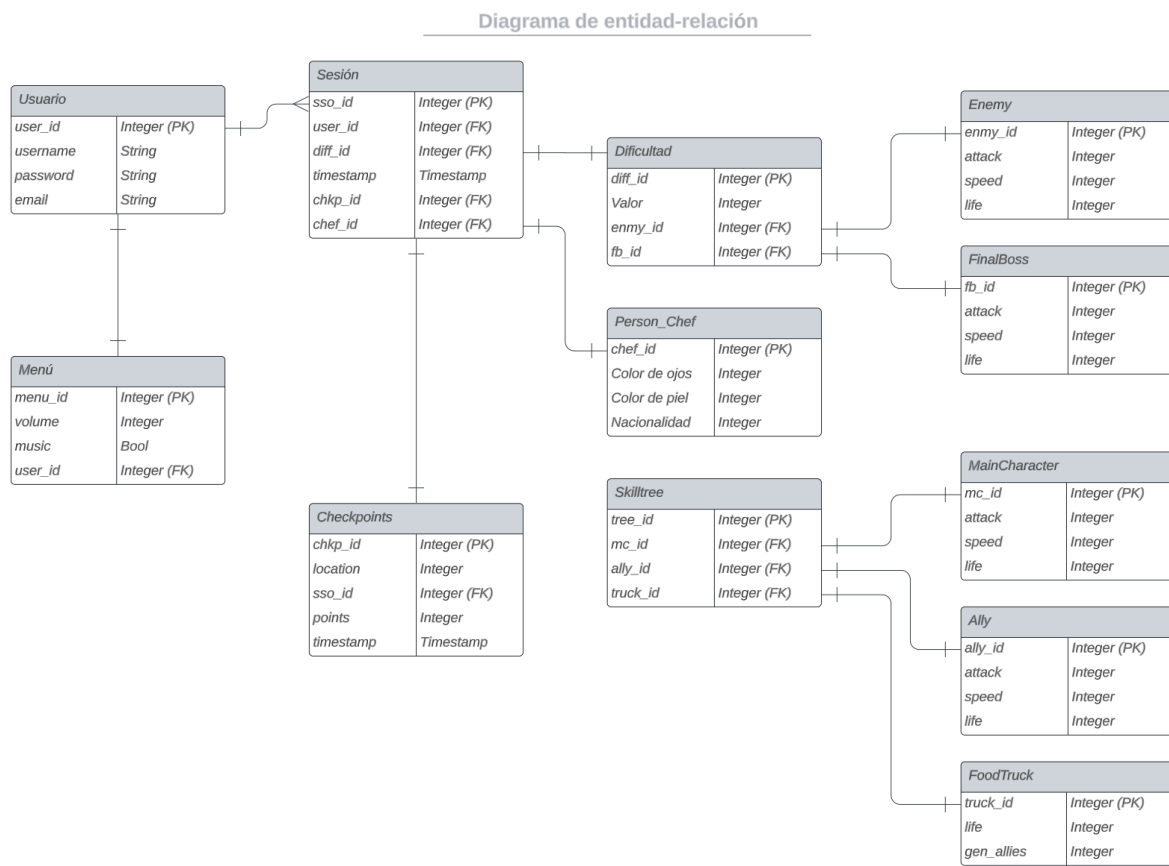
Esteban Castillo Juárez

Gilberto Echeverría Furió

Octavio Navarro Hinojosa

8 de junio de 2023

## Esquema conceptual



### Relaciones (nombre, cardinalidad en ambas direcciones, restricciones de integridad)

Un usuario puede tener múltiples sesiones, múltiples sesiones pueden ser de un mismo usuario.

- Cardinalidad: (Usuario) 1:M (Sesión)
- Restricciones: La llave primaria de usuario es user\_id y la llave primaria de Sesión es sso\_id. Las llaves secundarias de sesión son user\_id, diff\_id, chkp\_id y chef\_id.

Un usuario tiene un sólo menú asociado, un solo menú está asociado a un solo usuario.

- Cardinalidad: (Usuario) 1:1 (Menú)
- Restricciones: La llave primaria de Menú es menu\_id y la llave foránea es user\_id.

Una sola sesión está asociada a una sola dificultad, es decir, solo puede existir una dificultad por sesión.

- Cardinalidad: (Sesión) 1:1 (Dificultad)
- Restricciones: La llave primaria de Dificultad es diff\_id, las llaves foráneas son enemy\_id y fb\_id.

Una sola sesión está asociada a un solo valor de checkpoints, un solo valor (progreso) de checkpoints está asociado a cada sesión..

- Cardinalidad: (Sesión) 1:1 (Checkpoints)
- Restricciones: La llave primaria de Checkpoints es chkp\_id, la llave foránea de sso\_id,

Una sola sesión está asociada a una única personalización del chef.

- Cardinalidad: (Sesión) 1:1 (Person\_Chef)
- Restricciones: La llave primaria de Person\_Chef es chef\_id.

Un nivel de dificultad está asociado a un valor (multiplier) para los atributos en Enemy

- Cardinalidad: (Dificultad) 1:1 (Enemy)
- Restricciones: La llave primaria de Enemy es enemy\_id.

Un nivel de dificultad está asociado a un valor (multiplier) para los atributos en FinalBoss

- Cardinalidad: (Dificultad) 1:1 (FinalBoss)
- Restricciones: La llave primaria de FinalBoss es fb\_id.

Un skilltree está asociado a un único valor (multiplier) para los atributos en MainCharacter.

- Cardinalidad: (Skilltree) 1:1 (MainCharacter)
- Restricciones: La llave primaria de Skilltree es tree\_id, las llaves foráneas son mc\_id, ally\_id, truck\_id. La llave primaria de MainCharacter es mc\_id.

Un skilltree está asociado a un único valor (multiplier) para los atributos en Ally.

- Cardinalidad: (Skilltree) 1:1 (Ally)
- Restricciones: La llave primaria de Ally es ally\_id.

Un skilltree está asociado a un único valor (multiplier) para los atributos en FoodTruck.

- Cardinalidad: (Skilltree) 1:1 (FoodTruck)
- Restricciones: La llave primaria de FoodTruck es truck\_id.

Para todas estas relaciones, una restricción general es que sus campos sean NOT NULL, es decir, que contengan valores, además para user\_id se considera un valor autoincrementado.

## **Justificación del modelo en la tercera forma normal**

Para que un modelo pueda estar en la tercera forma normal, es menester que a su vez se encuentre en primera y segunda forma normal. Para que un modelo se encuentre en la 1FN debe tener las siguientes características:

1. Todos los atributos son atómicos.

2. Tiene una llave primaria única.
3. La llave primaria no contiene atributos nulos.
4. No existe variación en el número de columnas.
5. Los campos simples se identifican por la llave.
6. Todos los atributos o valores son indivisibles.
7. No existen grupos repetidos.

El modelo anteriormente descrito cumple con estas condiciones, tanto por la naturaleza de las tablas propuestas como por el uso de las restricciones, que se encargan de asegurarse que se tenga una llave primaria única para cada entidad y que no se tengan atributos nulos.

Para que el modelo se encuentre en 2FN tiene que cumplir además con lo siguiente:

1. No existen dependencias funcionales parciales.

El modelo cumple con esta condición ya que todas las entidades tienen campos que dependen directamente de la llave primaria y no se tienen atributos sobrantes o innecesarios. Por ejemplo, para la entidad Usuario, tanto el username, el correo y la contraseña asociados dependen completamente del user\_id y todos son necesarios para el login en el juego y registro de estadísticas entre otros datos. Asimismo, en la entidad de Person\_Chef también se tiene que todos los atributos dependen completamente de la llave primaria que es chef\_id y todos son parte de la personalización que se le da al chef en el juego, no hay alguno que sobre o que falte en ese rubro.

Por último, para que el modelo se encuentre en la 3FN debe cumplir con:

1. No existen dependencias transitivas entre las columnas.

En el modelo propuesto no existen dependencias funcionales transitivas, es decir, que ciertas columnas dependan de columnas que no son llave primaria. En el modelo esto no ocurre ya que en cualquiera de las entidades, sus atributos dependen exclusivamente de la columna identificada como la llave primaria. Por ejemplo, si en la entidad Sesión se eliminará el atributo de diff\_id (que es una llave secundaria que hace referencia a la llave primaria de la entidad Dificultad), no afectaría a ningún otro atributo de esta entidad. De igual forma, si se modificara o eliminara algún atributo dentro de la entidad Skilltree, las demás no se verían afectadas ya que no dependen de algo que no sea la llave primaria, modificar algo para el main character (el chef) no afectaría las cualidades de ataque, velocidad y vida de los aliados o del carro de comida.