



Tecnológico de Monterrey

Modelación de base de datos

Amilka Daniela Lopez Aguilar - A01029277

Jin Sik Yoon - A01026630

Emiliano Deyta Illescas - A01785881

Campus Santa Fe

Construcción de software y toma de decisiones

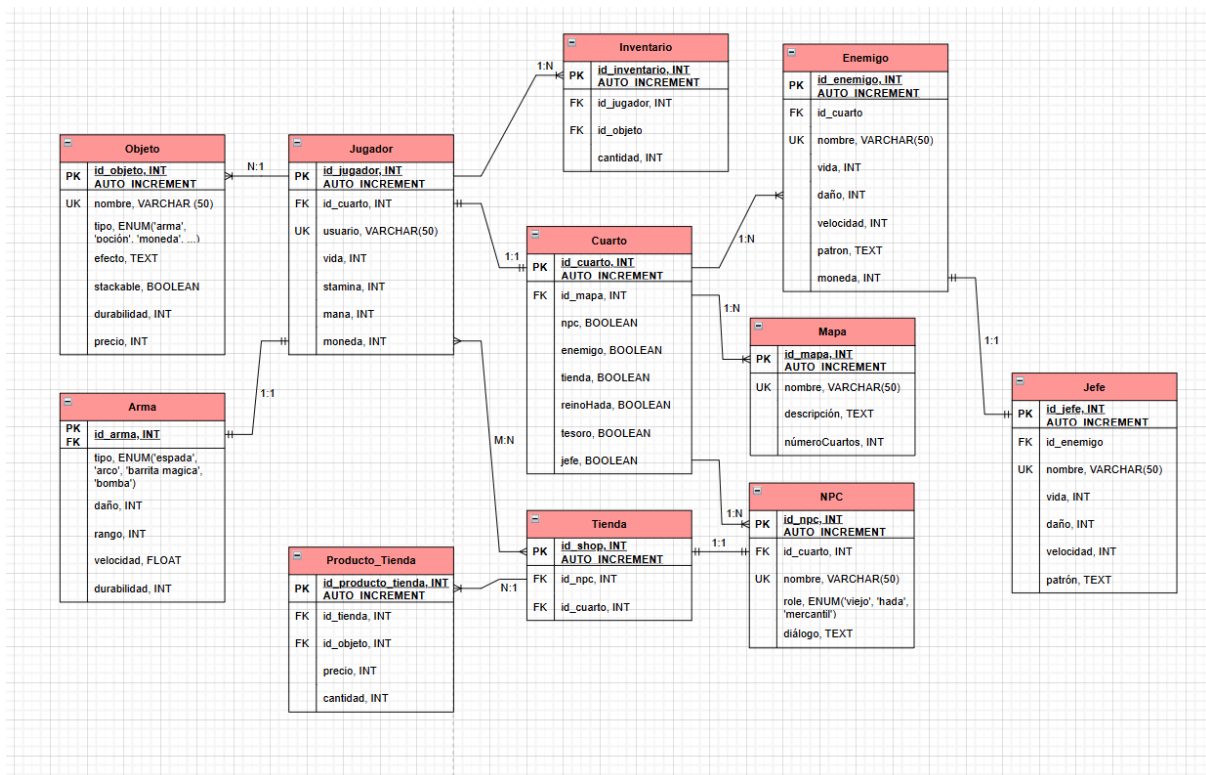
Octavio Navarro Hinojosa

Esteban Castillo Juarez

Gilberto Echeverría Furió

Viernes 14 de marzo de 2025

Diagrama ER



Justificación

Se comienza con la tabla principal de **Jugador**, la cual tiene como llave primaria un ID tipo entero que incrementa de 1 en 1 de acuerdo con los perfiles admitidos dentro del juego. Dicha tabla tiene una llave foránea tipo entero que permite conocer el cuarto o nivel en el que se encuentra el jugador. Indispensable para conocer y guardar su progreso. Jugador tiene otros atributos tipo entero como vida, stamina, mana y moneda. Las 4 son métricas dentro del juego que permiten darle un reto y repercusiones de recepción de ataque, completar un nivel y atacar. Las 4 se ven afectadas por cierto porcentaje durante un combate.

Jugador se relaciona con cardinalidad N-1 a **Objeto**, donde un jugador puede portar varios artefactos y varios artefactos le pertenecen a un jugador, esto porque la partida se reduce a un jugador dentro del mapa. No hay un modo de CO-OP. La tabla objeto igualmente funciona como un número entero que incrementa en ID. Objeto cuenta con una llave primaria tipo

entero que se incrementa conforme existan entradas del mismo. Tiene un nombre tipo varchar de 50 caracteres no repetible. guarda tipo enumerados de armas, pociones y monedas, lo cual permite llevar mejor control de lo que existe dentro del inventario del jugador. una descripción para el jugador de en qué consiste el efecto del objeto y si se ha acumulado con los “runs” del juego, lo que permite construir un jugador más fuerte conforme pierde. Aquellos no acumulables cuentan con una duración tipo entero y un precio tipo entero a reflejarse en la tienda y en las monedas como una compra.

Objeto difiere de tienda en la portabilidad del jugador. Si el artefacto se encuentra en tienda, el jugador aún no lo adquiere.

Jugador se relaciona con la tabla **Arma** a través de una llave foránea tipo entero de ID_arma. La relación es 1:1. Incluso si el jugador cambia a las armas especiales, dicha arma será la única que puede portar durante un tiempo predeterminado, por lo que el jugador debe ser crítico al pensar en el mejor momento para utilizar ciertas armas. Arma guarda tipos enumerados para diferenciar entre ellos. Cada arma cuenta con una cantidad entera de daño y rango, así como una velocidad de ataque tipo flotante y una duración de portación tipo entero.

El Jugador cuenta con una relación 1 a N con la tabla **Inventario**, pues un jugador contará con muchos objetos dentro de su inventario (considerando que siempre inicia con un arma sencilla por defecto). Su llave primaria es un ID tipo entero que se incrementa conforme recibe artefactos y monedas. Inventario accede mediante llaves foráneas al ID de jugador para atribuirle su inventario, así como a objetos para atribuirle sus artefactos acumulados y monedas. Cada uno cuenta con una cantidad tipo entero para atribuirle efectos acumulados y disponibilidad.

Siguiendo la lógica del videojuego, Jugador se relaciona con **Cuarto**, Mapa, Enemigo y Jefe Final para definir puntualmente el progreso del jugador. Jugador se relaciona con cuarto con una cardinalidad de 1:1, pues solamente puede estar en un nivel de los 10 que existen a la vez. Lo hace a través del ID de cuarto tipo entero que se incrementa. Cuarto tiene atributos de id_mapa (llave foránea a la tabla), si existe o no un npc dentro del cuarto, lo que permite diferenciar entre niveles de tienda/reto de los niveles tradicionales de combate, por lo que recibe un booleano para verificar si existen enemigos, tienda, el reino de las hadas, tesoros o un jefe final dentro del cuarto.

Cada cuarto puede contar con varios mapas 1:N dependiendo de la implementación. La tabla **Mapa** tiene una llave primaria tipo entero que se autoincrementa y guarda una llave única de nombre tipo VARCHAR para evitar niveles repetidos, una descripción tipo texto para el jugador y el número de cuartos tipo entero a implementar.

Cuarto accede al enemigo a través de la llave foránea que le atribuye a cada enemigo un ID de cuarto. 1 cuarto puede contar con varios enemigos N. **Enemigo** tiene una llave primaria ID tipo entero que se incrementa. Guarda el nombre VARCHAR único para evitar que se repita entre el diseño de los enemigos. Cada enemigo tiene atributos tipo entero de vida, daño, velocidad y monedas que pueden soltar. Cuentan con una descripción para que el jugador entienda sus patrones de ataque.

Para cada enemigo, y por ende, cada cuarto, existe 1 **Jefe** 1:1. Se le atribuye una llave foránea que indica su ID y respeta que debe tener un nombre único tipo VARCHAR. Cuenta con el resto de los atributos mencionados para Enemigo.

Para evaluar la aparición de un NPC dentro del cuarto, se indica que 1 cuarto puede tener múltiples NPC N. La tabla **NPC** guarda su llave primaria ID tipo entero que se incrementa. Se le atribuye a un cuarto según su idea especificado en la llave foránea. Contiene un nombre único tipo VARCHAR, un rol enumerado para conocer el tipo de NPC, y su diálogo predeterminado tipo texto.

Tienda es una tabla intermedia que permite relacionar NPC con tiendas y productos dentro de ella. Solo 1 NPC se encarga de 1 tienda, la cual tiene una llave primaria tipo ID entero que se incrementa. Las llaves foráneas tipo ID de cuarto y NPC enteros permiten atribuirle la tienda a cierto cuarto y NPC, que en este caso, aparecen cada 3 cuartos y solamente por parte del anciano.

Tienda se relaciona en 1 a varios N productos. La tabla **Producto_Tienda** tiene una llave primaria tipo ID entero que se incrementa según la cantidad de productos que se le añadan. Dicha tabla cuenta con llaves foráneas que permiten atribuirle ciertos objetos y la tienda (nivel) al que pertenece. Cuenta con atributos tipo entero de precio y cantidad disponible, para evitar que el jugador compre sin los recursos necesarios o más de lo necesario con el fin de mantener un juego retador.