

EVALUACION			Parcial	GRUPO		FECHA	30/11/2023
MATERIA			Bases de Datos 1				
CARRERA			Analista Programador / Analista en Tecnologías de Información				
CONDICIONES			<ul style="list-style-type: none"> - Puntos: - Duración: 2 horas - Sin material 				
Numero de Estudiante			Nombre				
Ejercicio 1:	Ejercicio 2a:	Ejercicio 2b:	Ejercicio 2c:	Ejercicio 2d:		Nota:	

Ejercicio 1

Se busca representar a través de un Modelo Entidad-Relación (MER) la estructura de datos de la plataforma de juegos en línea, Steam. La realidad de la plataforma se describe de la siguiente manera:

Steam, una reconocida plataforma de juegos en línea dispone de información relevante para su funcionamiento:

La plataforma cuenta con una variedad de usuarios registrados. Cada usuario tiene un nombre de usuario que lo identifica, dirección de correo electrónico que es única y un saldo asociado.

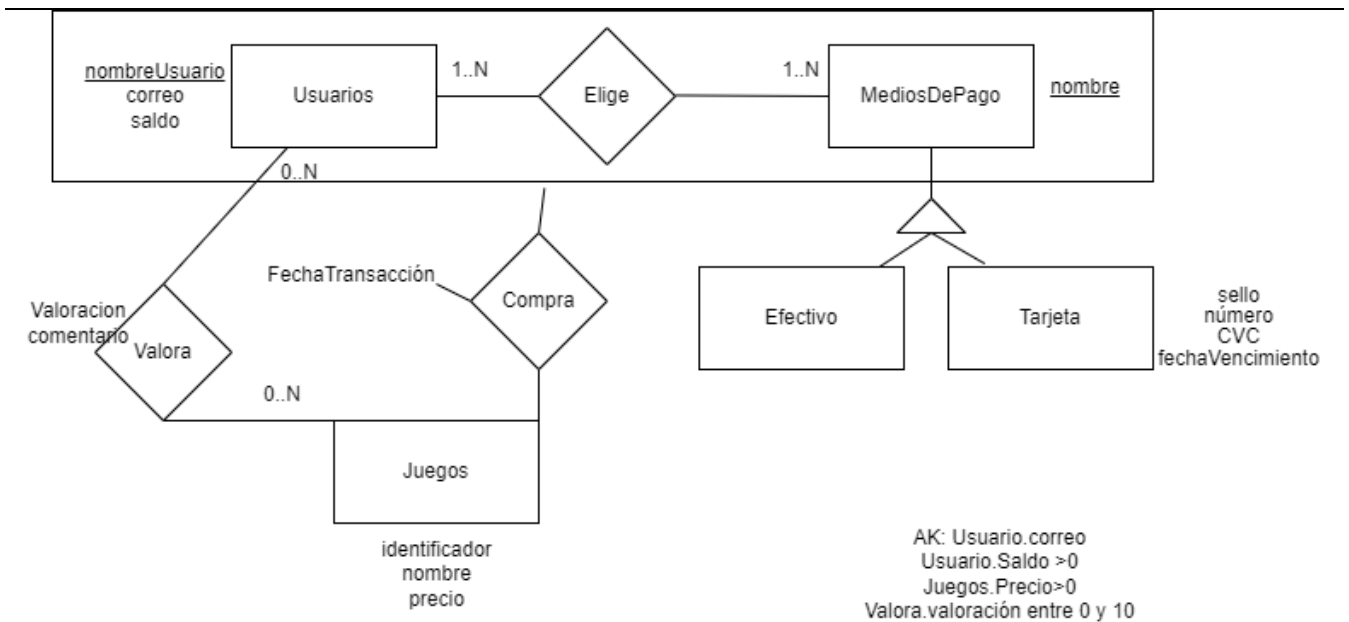
Existen medios de pago los cuales elige un usuario al menos uno al momento de crear la cuenta. Se identifican por un nombre y se distinguen, entre efectivo o tarjeta. Para el caso de Tarjeta, interesa guardar el sello, número, CVC y fecha de vencimiento de esta.

Se ofrece una amplia gama de juegos disponibles para los usuarios. Cada juego en Steam tiene un identificador único, nombre, precio.

Los usuarios de Steam pueden realizar compras de juegos disponibles seleccionando un medio de pago de los elegidos en la plataforma. Se debe registrar el usuario que realiza la compra, el medio de pago seleccionado y los juegos adquiridos, además de la fecha de la transacción.

Los usuarios pueden valorar los juegos, interesa registrar la puntuación asignada que debe ser un valor entre 0 y 10 y un comentario opcional.

Represente el modelo entidad relación (MER) que refleje la estructura de datos de la plataforma de juegos Steam, considerando las entidades, atributos, relaciones y restricciones necesarias para su correcta representación. (10 pts)



Ejercicio 2

- Dado el siguiente esquema:

Proveedores(P#, PNOMBRE, CIUDAD)

Componentes(C#, CNOMBRE, COLOR, PESO, CIUDAD)

Articulos(T#, TNOMBRE, CIUDAD)

Envios(P#, C#, T#, CANTIDAD)

Donde P#, C#, T# tienen una FK, cada una a sus respectivas tablas origen.

Con los siguientes datos a modo de ejemplo

PROVEEDORES

P#	PNOMBRE	CIUDAD
P1	CARLOS	SEVILLA
P2	JUAN	MADRID

ENVIOS

P#	C#	T#	CANTIDAD
P1	C1	T1	200

COMPONENTES

T#	TNOMBRE	CIUDAD
T1	CLASIFICADORA	MADRID
T2	PERFORADORA	MALAGA
T3	LECTORA	CACERES

ARTICULOS

T#	TNOMBRE	CIUDAD
T1	CLASIFICADORA	MADRID
T2	PERFORADORA	MALAGA
T3	LECTORA	CACERES

a) Script de creación de las tablas involucradas indicando en cada tabla PK y FK correspondientes (10 ptos)

-- Creación de la tabla Proveedores

```
CREATE TABLE Proveedores (  
    P# INT PRIMARY KEY,  
    PNOMBRE VARCHAR(100),  
    CIUDAD VARCHAR(50)  
);
```

-- Creación de la tabla Componentes

```
CREATE TABLE Componentes (  
    C# INT PRIMARY KEY,  
    CNOMBRE VARCHAR(100),  
    COLOR VARCHAR(50),  
    PESO DECIMAL(10, 2),  
    CIUDAD VARCHAR(50),  
    FOREIGN KEY (CIUDAD) REFERENCES Proveedores(CIUDAD)  
);
```

-- Creación de la tabla Artículos

```
CREATE TABLE Articulos (  
    T# INT PRIMARY KEY,  
    TNOMBRE VARCHAR(100),  
    CIUDAD VARCHAR(50)  
);
```

-- Creación de la tabla Envios

```
CREATE TABLE Envios (  
    P# INT,  
    C# INT,  
    T# INT,  
    CANTIDAD INT,  
    PRIMARY KEY (P#, C#, T#),  
    FOREIGN KEY (P#) REFERENCES Proveedores(P#),  
    FOREIGN KEY (C#) REFERENCES Componentes(C#),  
    FOREIGN KEY (T#) REFERENCES Articulos(T#)  
);
```

b) Obtener los numero de proveedores(P#) de los proveedores que abastecen el artículo T1 y sean de la misma ciudad. (5 ptos)

```
SELECT DISTINCT P.P#  
FROM Proveedores P  
JOIN Envios E ON P.P# = E.P#  
JOIN Articulos A ON E.T# = A.T#  
WHERE A.T# = 'T1' AND A.ciudad =P.ciudad
```

c) Obtener los números de componentes (C#) de los componentes suministrados a cualquier artículo de ciudades que empiecen con M. (5 pts)

```
SELECT DISTINCT C.C#  
FROM Envios E  
JOIN Articulos A ON E.T# = A.T#  
JOIN Componentes C ON E.C# = C.C#  
WHERE P.CIUDAD LIKE 'M%'
```

d) Obtener para cada proveedor el nombre, el promedio de cantidades que realiza los envíos solo para aquellos que tienen al menos 10 envíos realizados. (10 pts)

```
SELECT P.PNOMBRE AS 'Nombre del Proveedor', AVG(E.CANTIDAD) AS 'Promedio de  
Cantidades de Envío'  
FROM Proveedores P  
JOIN Envios E ON P.P# = E.P#  
GROUP BY P.PNOMBRE  
HAVING COUNT(E.P#) >= 10
```