

Evaluación de portafolio[Actividad Evaluada]

Francisco Valencia A.

Requerimientos Funcionales Mínimos Esperados

1) Distinguir las características, rol y elementos fundamentales de una base de datos relacional para la gestión de la información en una organización.

Las bases de datos relacionales, es un tipo de base de datos que organiza la información en estructuras de tablas, filas, columnas, registros, claves primarias y claves foráneas. Estas nos permiten generar relaciones entre las diversas tablas e información ingresada que a su vez nos facilita el proceso de consultas a dicha base de datos. Por medio de este tipo de base de datos podemos establecer relaciones entre las tablas con la finalidad de mostrar de manera lógica y eficiente los datos de manera organizada.

Tabla: Estructura central de una base de datos relacional. Aquí se almacenan los datos en filas y columnas.

Filas: Representan instancias individuales de datos dentro de una tabla. Cada fila cuenta con un identificador único conocido como la llave primaria.

Columnas: Aquí se definen las características de los datos almacenados en la tabla. Cada columna tiene un tipo de dato específico.

Registro: Es una fila en una tabla que contiene la información completa sobre una entidad.

Campo: Es un atributo específico de cada entidad, como nombre, dirección. El campo estructura los registros, cada registro tiene un valor único para cada uno de los campos de una tabla.

Entidad: Representan objetos o conceptos sobre los que se quiere guardar información. Por ejemplo Libro, Cliente, Usuario.

Atributos: Son las características o propiedades de cada entidad, por ejemplo la entidad libro tiene atributos como título, autor, número de páginas, entre otras.

Restricciones: Estas son reglas que se aplican a las columnas de una tabla, como las claves primaria y foránea. Nos ayudan a prevenir errores y mantener consistencia en los datos.

Clave Primaria: Es un identificador único para cada fila en una tabla, permite la identificación, y la relación con otras tablas.

Clave Foránea: Columnas en una tabla que hacen referencia a la clave primaria de otra tabla, permitiendo establecer relaciones entre dichas tablas.

Relaciones: Son vínculos entre tablas, establecidas mediante claves externas, que nos permiten relacionar datos entre diversas tablas y cruzar su información.

Restricciones: Son reglas que se aplican a los datos que nos permiten mantener la integridad y la coherencia.

Vistas: Es una tabla virtual basada en el resultado de una consulta SQL, no almacena datos en sí misma, sino que se define sobre una o más tablas existentes. Se usan para simplificar consultas complejas, mejorando la seguridad por medio de restringir el acceso a los datos o presentarlos de manera personalizada sin exponer información innecesariamente. Las visas pueden ser solo de lectura o permitir operaciones que modifiquen la base de datos.

Índices: Es una estructura de datos que nos ayuda a buscar más rápido los datos en una tabla. Al crear índices sobre una columna específica nos permite que las consultas en dicha columna se puedan ejecutar más rápido, ya que con el índice logramos que no se busque en toda la tabla sino que vaya directamente a lo solicitado y establecido.

Procedimientos almacenados: Es un conjunto de instrucciones SQL que se almacenan en la base de datos y que se pueden ejecutar repetidamente. En un procedimiento almacenado se puede recibir parámetros de entrada y devolver resultados, ayudan a mejorar el rendimiento al reducir la cantidad de datos que se transfieren entre una aplicación y la base de datos, ya que permite ejecutar consultas complejas con una sola llamada.

Transacciones: Consiste en una unidad de trabajo que consta de una o más operaciones sobre nuestra base de datos. Permite garantizar que dichas operaciones dentro de una transacción se ejecuten con éxito o manejar que no se realice dicha operación, asegurando con ello que nuestros datos se mantengan íntegros y no se modifiquen ante posibles errores y contemos con una forma de evitar la pérdida o desorganización de información.

Funciones: Consiste en un bloque de código SQL que nos retorna un valor, son útiles para encapsular cálculos que se repiten frecuentemente en diferentes consultas.

Triggers: Este corresponde a un procedimiento almacenado que se ejecuta automáticamente cuando un evento específico de la base de datos ocurre, por ejemplo actualización, eliminación de datos en una tabla, inserción de datos. Con los triggers podemos auditar modificaciones en la base de datos, mejorar la integridad de los mismos, o generar acciones o respuestas relacionadas a eventos de los datos.

Consultas SQL: Corresponde al lenguaje estándar que se utiliza para consultar y manipular datos en una base de datos relacional, por medio de esta podemos recuperar, insertar, modificar y eliminar datos.

Gestión de datos en tablas.

En una base de datos relacional, los datos se organizan en tablas. Estas son la estructura fundamental de una base de datos relacional, ya que cada tabla contiene datos relacionados a una entidad específica, como sería el caso de una entidad cliente, otro producto por ejemplo. Cada una de esas tablas representa una entidad y cada fila de dicha tabla representa un registro único de esta entidad, así es como en una tabla cliente cada fila correspondería a un cliente específico que a su vez cuenta con detalles sobre dicho cliente. Así cada columna de dicha tabla, nos muestra o representa un atributo, con características específicas de la entidad que estructura nuestra tabla. Siguiendo el mismo ejemplo, una tabla clientes encontramos columnas como nombre, dirección, teléfono, entre otras.

Las relaciones entre tablas se establecen mediante claves primarias y foráneas, que permiten conectar información de diferentes tablas, dándonos acceso a información de la tabla. La clave primaria es un atributo único que identifica cada registro de la tabla. La clave foránea corresponde a una columna de una tabla que referencia a una clave primaria de otra tabla, con ella podemos relacionar dichas tablas entre sí y ampliando la información que podemos acceder o conocer por ejemplo de un cliente. Así mismo las relaciones entre tablas pueden ser de varios tipos; uno a uno, que es cuando un registro en una tabla se relaciona con un único registro en otra. Uno a varios, un registro se puede relacionar con varios registros en otra tabla. Varios a varios, varios registros se pueden relacionar con varios registros de otra tabla.

Los beneficios de las relaciones entre tablas es que nos permiten tener integridad de datos, evita la redundancia, es flexible ya que nos permite realizar consultas, y combinar tablas para obtener información, con ello mejoramos la organización de la información sobretodo cuando son grandes volúmenes de datos los que se manejan.

En este caso particular, quiero comenzar a estructurar una plataforma para jugadores de cartas intercambiables (de ahora en adelante TCG), donde los usuarios puedan registrarse y crear un perfil, conectarse con otros jugadores, acordar duelos presenciales y ganar puntos para acceder a beneficios, relacionados a su participación y número de victorias.

En dicho ejemplo usare las siguientes entidades; Jugador, Duelos, ResultadosDuelos, Ubicaciones, Beneficios, HistorialBeneficios.

Ejemplo:

Archivo adjunto, clavef.sql.

2) Utilizar Lenguaje Estructurado de Consultas (SQL) para la obtención de información que satisface los requerimientos planteados a partir de un modelo de datos dado.

Las relaciones entre las tablas que se mostraron anteriormente se pueden describir de la siguiente forma:

Un jugador puede participar en muchos duelos.

Un duelo tiene un resultado (ganador,perderdor, empate).

Un duelo se asocia a una ubicación.

Un jugador puede acumular puntos y canjear beneficios.

Se registra el historial de canjes en HistorialBeneficios.

5) Elaborar un modelo de datos de acuerdo a los estándares de modelamiento para resolver un problema de baja complejidad.

Ver imagen cardinalidad.png.

Este modelo se considera una solución de baja complejidad inicial pero diseñada para ser escalable. Algunas mejoras previstas en futuras versiones incluyen:

- Agregar entidad mazos para registrar los mazos de cada jugador.
- Incluir catálogo de cartas (cartas) con información de nombre, rareza, tipo, etc.
- Registrar historial de partidas con estadísticas adicionales (turnos, duración, modalidad).
- Normalización avanzada a 3FN para aislar atributos derivados o calculables.
- Optimización mediante índices en campos de búsqueda frecuente como nombre_usuario o id_jugador.