

Obligatorio Base de Datos 2 - 2025

Enunciado del Problema

Se solicita la creación de una base de datos para una plataforma de e-learning, con el fin de gestionar:

1. Docentes (quienes crean y administran los cursos).
2. Cursos, cada uno asociado a un docente, una nota mínima de aprobación y un conjunto de módulos.
3. Módulos, cada uno perteneciente a un curso y con un valor en puntos.
4. Clases, pertenecientes a un módulo y también con un valor en puntos.
5. Estudiantes, que se inscriben en cursos, registran su progreso, obtienen calificaciones y pueden recibir certificaciones.
6. Bitácora de auditoría, para las operaciones (INSERT, UPDATE, DELETE) realizadas sobre las tablas principales (cursos, módulos, clases, estudiantes, inscripciones, etc.).

Requerimientos Funcionales Detallados

1. Docentes
 - Un docente puede crear múltiples cursos.
 - Cada docente tiene información básica (nombre, apellido, correo, fecha de registro).
 - No se manejarán roles en esta versión, pero los docentes existen como una entidad aparte para vincularlos a los cursos.
2. Cursos

- Cada curso es creado por un docente (no se manejarán co-docentes ni colaboraciones en esta versión).
- Cada curso tiene un título, descripción, duración estimada y estado (activo o inactivo).
- Nota mínima de aprobación (por ejemplo, un campo MinGrade o NotaMinima) para recibir el certificado.
- Un curso se compone de uno o varios módulos.

3. Módulos

- Cada módulo pertenece a un curso.
- Cada módulo almacena información como: título, descripción (opcional), orden dentro del curso y valor en puntos para determinar su peso o contribución a la calificación final.
- Cada módulo se compone de una o varias clases.

4. Clases

- Cada clase pertenece a un módulo.
- Almacena un título, el contenido o descripción, y valor en puntos (opcional o según la estrategia de calificación).
- Las clases contribuyen al puntaje acumulado del módulo o directamente a la calificación final del curso, de acuerdo con la lógica que se defina.

5. Estudiantes

- Almacenan datos personales básicos (nombre, apellido, correo electrónico, fecha de registro).
- Pueden inscribirse en uno o varios cursos.

6. Inscripciones, Progreso y Calificación

- Un estudiante se inscribe en un curso, lo cual genera un registro de inscripción.
- Se registra el porcentaje de avance (0% a 100%) en cada curso.
- Se lleva una calificación final para cada estudiante en cada curso (puede ser numérica, en escala 0 a 100).
- Para determinar la calificación, se pueden tomar en cuenta los puntos obtenidos en los módulos y/o clases (por ejemplo, sumatoria o promedio ponderado).
- Si la calificación final es mayor o igual a la nota mínima del curso, el estudiante obtiene un certificado.

7. Certificación

- Al aprobar un curso, se genera un registro de certificación con:
 - Identificador único (o número de certificado).
 - Fecha de emisión.
 - Referencia al estudiante y al curso (o a su inscripción).

8. Auditoría de Cambios

- Se requieren triggers que registren las modificaciones (INSERT, UPDATE, DELETE) en las tablas clave (docentes, cursos, módulos, clases, estudiantes, inscripciones, calificaciones, certificaciones, etc.).
- La información mínima a registrar en la bitácora:
 - Usuario (o identificador simulado) que realiza la operación.
 - Fecha y hora del cambio.
 - Tipo de operación (INSERT, UPDATE, DELETE).

- Datos relevantes antes y/o después de la operación (según corresponda).

9. Motor de Base de Datos

- Implementación obligatoria en SQL Server Express, utilizando T-SQL y sus particularidades (uso de IDENTITY, etc.).

Entregable 1: Análisis y Diseño de la Base de Datos

Para la primera entrega, se solicita:

1. Análisis del Problema

- Explicar la estructura y la lógica para registrar docentes, cursos, módulos, clases, estudiantes, inscripciones y calificaciones.
- Describir la manera en que se utilizarán los puntos en módulos/clases para calcular la calificación final (de forma general, no se requiere un algoritmo complejo, pero sí coherente).
- Especificar cómo se manejará la nota mínima (por curso) para determinar la aprobación.

2. Diagrama Relacional

- Mostrar claramente las tablas, con llaves primarias y foráneas, y las relaciones entre ellas..
- Incluir la tabla (o tablas) necesaria(s) para registrar el progreso, calificaciones y certificaciones.

3. Diccionario de Datos

- Describir cada tabla: nombre, campos, tipos de datos, restricciones (NOT NULL, UNIQUE, etc.), llaves primarias (PK) y foráneas (FK).

4. Script de Creación de Tablas

- Suministrar el código T-SQL que crea todas las tablas y restricciones.

5. Inserción de Datos de Prueba

- Proveer un script con sentencias INSERT para poblar las tablas con datos de ejemplo:
 - Varios docentes (al menos 2).
 - Varios cursos (al menos 2), cada uno creado por un docente distinto.
 - Módulos y clases correspondientes.
 - Varios estudiantes inscritos en los cursos, con algo de progreso y calificaciones de prueba.

Entregable final: Implementación Final

Para la segunda entrega, se requiere:

1. Documentación Final de la Solución

- Describir concretamente cómo se realiza el cálculo de la calificación final y la verificación de la nota mínima.
- Explicar el proceso para generar la certificación cuando un estudiante cumple con la calificación mínima.

- Incluir documentación solicitada en la primera entrega con las modificaciones solicitadas (si se requiere).

2. Scripts de Creación de Tablas (Versión Definitiva)

- Incluir las modificaciones o correcciones realizadas con base en el feedback de la primera entrega.

3. Stored Procedures (SPs)

- Registrar un nuevo docente.
- Registrar un nuevo estudiante.
- Crear un nuevo curso asociado a un docente.
- Inscribir a un estudiante en un curso.
- Actualizar la calificación final de un estudiante para un curso, tomando en cuenta los puntos de módulos/clases y la nota mínima para aprobar.
- Generar la certificación, si la calificación final es mayor o igual a la nota mínima.

4. Funciones (Functions)

- Al menos una función que facilite alguna operación de cálculo o verificación, por ejemplo, el cálculo de la calificación final o la determinación de si un estudiante aprueba un curso.

5. Vistas (Views)

- Al menos una vista que consolide información de interés:
 - Vista de "Cursos y su docente" con la nota mínima y el estado del curso.
 - Vista de "Estudiantes, progreso y calificación en cada curso".

6. Triggers para Auditoría

- Implementar triggers para las tablas: docentes, cursos, módulos, clases, estudiantes, inscripciones, calificaciones, certificaciones y cualquier otra tabla que considere crítica.
- Registrar en tablas de auditoría:
 - Operación realizada (INSERT, UPDATE, DELETE).
 - Usuario o identificador simulado de quién hizo el cambio.
 - Fecha y hora del cambio.
 - Datos relevantes (antes y después, si aplica).

7. Evidencia de Funcionamiento

- Scripts de prueba que muestren la inserción de nuevos cursos, la inscripción de estudiantes, la actualización de calificaciones y la generación de certificaciones.
- Mostrar que los triggers almacenan la información requerida en la bitácora de auditoría.