



Universidad Tecnológica Nacional  
Facultad Regional Buenos Aires

Bases De Datos 2025

Curso: K3522

Turno: Mañana

## Trabajo Practico Bases De Datos GD2C2025

GRUPO N°6 - DROP_DATABASE	
Nombre y Apellido	Legajo
Alvaro Federico Gianola Otamendi	209.536-1
Mariano Amir Luna	209.061-2
Oscar Mercado Ruiz Díaz	208.301-2
Eliseo Gerard Paniagua Muriel	202.748-3

FECHA DE PRESENTACIÓN:	31/10/2025
FECHA DE DEVOLUCIÓN:	
CALIFICACIÓN:	
FIRMA PROFESOR:	



UTN.BA  
INGENIERIA  
EN SISTEMAS

Índice

Estrategia General	3
<hr/>	
Decisiones de Diseño del DER	3
• Curso y días de cursado	3
• Módulos y cursos	3
• Evaluaciones y módulos	3
• Encuestas	3
• Factura y detalle de factura	3
<hr/>	
Constraints	4
• PK	4
• FK	4
• Check constraints	4
<hr/>	
Triggers	4
• trg_validarRangoFechasCurso:	4
• trg_fecha_cambio_estado_inscripcion:	4
• trg_asignarInstanciaParcial:	4
• trg_unicaInscripcion:	4
<hr/>	
Índices	4
<hr/>	
Valores Predeterminados	4
<hr/>	
Consideraciones de Integridad y Rendimiento	5
<hr/>	
Conclusión	5

Bases De Datos	
TPGD2C2025	Grupo: DROP_DATABASE

# Estrategias

## Estrategia General

El modelo de datos fue diseñado con el objetivo de garantizar integridad, coherencia y eficiencia. Se implementó en un esquema propio (DROP\_DATABASE) para aislar las tablas del resto de los trabajos y evitar conflictos. Se aplicaron principios de normalización hasta 3FN, reduciendo la redundancia y asegurando una estructura fácilmente mantenible.

## Decisiones de Diseño del DER

Durante el paso del modelo conceptual al relacional, se tomaron varias decisiones para mejorar la representación de los datos y ajustarse a los requerimientos funcionales del enunciado:

- Curso y días de cursado**  
 En el DER inicial, la entidad Día\_Semana estaba relacionada directamente con Curso. Esto impedía que un curso tuviera más de un día de cursado. Para resolverlo, se incorporó una entidad intermedia Día\_Cursado, que permite la relación 1 curso → N días, respetando la multiplicidad real del negocio.
- Módulos y cursos**  
 Se estableció una relación muchos a muchos entre Curso y Módulo mediante la tabla intermedia Modulo\_x\_Curso, dado que un módulo puede formar parte de varios cursos y un curso puede tener múltiples módulos.
- Evaluaciones y módulos**  
 Para poder registrar qué módulos fueron evaluados en una instancia específica, se creó la tabla Modulo\_de\_curso\_tomado\_en\_evaluacion, que relaciona Evaluacion con Modulo\_x\_Curso. Esto permite mayor trazabilidad sobre los contenidos evaluados y facilita consultas posteriores.
- Encuestas**  
 Las encuestas se modelaron separando las entidades Encuesta, Pregunta, Encuesta\_Respondida y Detalle\_Encuesta\_Respondida, de modo que:
  - cada curso tenga su encuesta,
  - cada encuesta contenga 4 preguntas numeradas,
  - y cada respuesta se registre individualmente.
 Esta estructura permite almacenar los resultados históricos sin perder consistencia ni duplicar información.
- Factura y detalle de factura**  
 Se decidió implementar una tabla Periodo independiente para reflejar el año y mes del ciclo facturado, evitando el uso de fechas duplicadas y facilitando agrupamientos y reportes por período.

Estas decisiones fueron tomadas priorizando la representación fiel del dominio y la facilidad para migrar y consultar los datos provenientes de la tabla Maestra.

Bases De Datos	
TPGD2C2025	Grupo: DROP_DATABASE

## Constraints

Se definieron Primary Keys, Foreign Keys y Check Constraints en todas las tablas:

- **PK:** garantizan unicidad de cada registro.
- **FK:** aseguran integridad referencial (por ejemplo, un Curso no puede existir sin una Sede, y una **Inscripcion\_Curso** debe tener un Alumno válido).
- **Check constraints:** controlan reglas de negocio simples, como:
  - nota BETWEEN 1 AND 10
  - precioMensual >= 0
  - nroPregunta BETWEEN 1 AND 4

Esto evita inconsistencias y asegura que los datos cumplan con las restricciones del dominio.

## Triggers

Se implementaron triggers automáticos para reforzar reglas de negocio y mantener la coherencia:

- **trg\_validarRangoFechasCurso:** evita que la fecha de inicio sea posterior a la de fin.
- **trg\_fecha\_cambio\_estado\_inscripcion:** actualiza automáticamente la fecha de respuesta al modificarse el estado.
- **trg\_asignarInstanciaParcial:** calcula automáticamente el número de intento de un alumno según sus evaluaciones previas.
- **trg\_unicaInscripcion:** impide que un alumno se inscriba dos veces al mismo curso.

De esta forma, el sistema mantiene coherencia temporal y lógica sin intervención manual.

## Índices

Se crearon índices en columnas con alta frecuencia de búsqueda o de unión (JOIN):

- IX\_Inscripcion\_Curso\_Alumno (por legajoAlumno)
- IX\_Inscripcion\_Curso\_Curso (por codigoCurso)
- IX\_TP\_Alumno\_Curso (por curso)
- IX\_Factura\_Alumno (por legajoAlumno)
- IX\_Pago\_Factura (por id)

Estos índices optimizan las consultas en reportes, inscripciones, pagos y facturación.

## Valores Predeterminados

Se asignaron valores por defecto a campos que se completan automáticamente:

- fechaInscripcion, fechaRegistro → SYSDATETIME()

Bases De Datos	
TPGD2C2025	Grupo: DROP_DATABASE

- estado de inscripción → 'pendiente'
- periodoAnio, periodoMes → año y mes actuales del sistema

Esto evita nulos innecesarios y garantiza consistencia temporal.

---

## Consideraciones de Integridad y Rendimiento

- Se evitó el borrado en cascada, priorizando la trazabilidad histórica.
- Los cálculos derivados, como la duración del curso, se definieron como columnas calculadas persistentes, optimizando rendimiento sin comprometer integridad.
- Todas las relaciones críticas están respaldadas por constraints e índices, lo que permite mantener equilibrio entre seguridad de datos y eficiencia en las consultas.

---

## Conclusión

El modelo combina reglas de negocio implementadas en el servidor (triggers y constraints) con un diseño relacional normalizado e indexado.

Las decisiones tomadas en el DER y su traducción al modelo físico garantizan una base consistente, escalable y alineada con los requerimientos funcionales del enunciado.