



# SISTEMAS TRANSACCIONALES

## SOLUCIÓN PROYECTO 1

Juan Jose Cediél Borrero *Código:* 202212212

*jj.cediél@uniandes.edu.co*

Este documento presenta la solución de la primera entrega del proyecto de ***Sistemas Transaccionales***.

Universidad de los Andes

Bogotá - Colombia

2 de octubre de 2023

# 1 PROYECTO ENTREGA 1

## 1.1 UML

<https://drive.google.com/file/d/1AoD45BOTV9vE7Ipk0GYz4rMerWPFEqgq/view?usp=sharing>

## 1.2 Resultados Logrados

Se ha logrado completar el diseño y la implementación de la base de datos, incluyendo relaciones y restricciones. Se realizaron pruebas exitosas para garantizar la integridad y consistencia de los datos.

## 1.3 Relaciones

El modelo de base de datos consta de 38 tablas que representan distintos aspectos del funcionamiento del hotel. Cada tabla tiene atributos que describen sus propiedades. Entre las tablas se encuentran: habitaciones, usuarios, consumos, servicios, etc.

### 1.3.1. Descripción de Tablas

1. bares	Columna	Tipo	Restricciones
	capacidad	INTEGER	NOT NULL
	estilo	VARCHAR2(50)	NOT NULL
	servicio_id	INTEGER	NOT NULL

2. bares_productos	Columna	Tipo
	bar_id	INTEGER NOT NULL
	producto_id	INTEGER NOT NULL

3. consumos	Columna	Tipo	Restricciones
	id	INTEGER	NOT NULL
	valor_total	NUMBER(12, 2)	NOT NULL
	fecha	TIMESTAMP	NOT NULL
	servicio_id	INTEGER	NOT NULL
	estancia_id	INTEGER	NOT NULL

4. dotacion_habitacion	Columna	Tipo	Restricciones
	tipo_habitacion	VARCHAR2(50 CHAR)	NOT NULL
	dotacion	VARCHAR2(50 CHAR)	NOT NULL

5. dotaciones	Columna	Tipo
	nombre	VARCHAR2(50 CHAR)
	NOT NULL	
	valor_agregado	NUMBER(12, 2)
	NOT NULL	

6. estancias	Columna
	id
	INTEGER
	NOT NULL

7. gimnasios	Columna	Tipo	Restricciones
	capacidad	INTEGER	NOT NULL
	apertura	TIMESTAMP	NOT NULL
	cierre	TIMESTAMP	NOT NULL
	valor	NUMBER(12, 2)	NOT NULL
	servicio_id	INTEGER	NOT NULL

8. gym_maqs	Columna	Tipo
	maquina_nombre	VARCHAR2(50 CHAR)
	NOT NULL	
	gimnasio_id	INTEGER
	NOT NULL	

9. habitaciones	Columna	Tipo	Restricciones
	numero	INTEGER	NOT NULL
	tipo	VARCHAR2(50 CHAR)	NOT NULL

10. indumentarias	Columna	Tipo	Restricciones
	elementos	INTEGER	NOT NULL
	valor_base	NUMBER(12, 2)	NOT NULL
	valor_elemento	NUMBER(12, 2)	NOT NULL
	servicio_id	INTEGER	NOT NULL
	id	INTEGER	NOT NULL

11. internet	Columna	Tipo	Restricciones
	capacidad	NUMBER(6, 2)	NOT NULL
	valor_por_dia	NUMBER(12, 2)	NOT NULL
	inicia	TIMESTAMP	NOT NULL
	finaliza	TIMESTAMP	NOT NULL
	servicio_id	INTEGER	NOT NULL

	Columna	Tipo	Restricciones
12. items	id	INTEGER	NOT NULL
	nombre	VARCHAR2(100 CHAR)	NOT NULL
	valor	NUMBER(12, 2)	NOT NULL

	Columna	Tipo
13. items_consumidos	consumo_id	INTEGER
	NOT NULL	
	item_id	INTEGER
	NOT NULL	

	Columna
14. maquinas	nombre
	VARCHAR2(50 CHAR)
	NOT NULL

	Columna	Tipo	Restricciones
15. ocupaciones	habitacion_numero	INTEGER	NOT NULL
	estancia_id	INTEGER	NOT NULL
	fecha	TIMESTAMP	NOT NULL

	Columna	Tipo	Restricciones
16. planes_de_consumo	nombre	VARCHAR2(100 CHAR)	NOT NULL
	descuento	NUMBER(5, 2)	
	minimas_noches	INTEGER	
	valor	NUMBER(12, 2)	
	vigencia_id	INTEGER	
	restriccion_id	INTEGER	

	Columna	Tipo	Restricciones
17. prestamos	utensilio	VARCHAR2(50 CHAR)	NOT NULL
	devuelto	VARCHAR2(5 CHAR)	DEFAULT 'false'
	condicion	VARCHAR2(30 CHAR)	NOT NULL
	servicio_id	INTEGER	NOT NULL
	id	INTEGER	NOT NULL

18. productos	Columna	Tipo	Restricciones
	origen	VARCHAR2(20)	NOT NULL
	item_id	INTEGER	NOT NULL

19. productos_incluidos	Columna	Tipo	Restricciones
	plan_de_consumo	VARCHAR2(100 CHAR)	NOT NULL
	producto_id	INTEGER	NOT NULL
	cantidad	NUMBER(12, 2)	NOT NULL

20. piscinas	Columna	Tipo	Restricciones
	capacidad	INTEGER	NOT NULL
	apertura	TIMESTAMP	NOT NULL
	cierre	TIMESTAMP	NOT NULL
	valor	NUMBER(12, 2)	NOT NULL
	profundidad	NUMBER(12, 2)	NOT NULL
	servicio_id	INTEGER	NOT NULL

21. reserva_habs	Columna	Tipo	Restricciones
	reserva_id	INTEGER	NOT NULL
	habitacion_numero	INTEGER	NOT NULL

22. reservas_estancias	Columna	Tipo	Restricciones
	id	INTEGER	NOT NULL
	inicio	TIMESTAMP	NOT NULL
	fin	TIMESTAMP	NOT NULL
	cantidad_huespedes	INTEGER	NOT NULL
	cliente	INTEGER	NOT NULL

23. reservas_servicios	Columna	Tipo	Restricciones
	fecha	TIMESTAMP	NOT NULL
	duracion	INTEGER	NOT NULL
	numero_habitacion	INTEGER	NOT NULL
	servicio_id	INTEGER	NOT NULL

24. restaurantes	Columna	Tipo	Restricciones
	capacidad	INTEGER	NOT NULL
	estilo	VARCHAR2(50)	NOT NULL
	servicio_id	INTEGER	NOT NULL

25. restaurantes_prods	Columna	Tipo	Restricciones
	restaurante_id	INTEGER	NOT NULL
	producto_id	INTEGER	NOT NULL

26. rest_prods	Columna	Tipo
	restaurante_id	INTEGER
	NOT NULL	
	producto_id	INTEGER
	NOT NULL	

27. restaurante_prods	Columna	Tipo
	restaurante_id	INTEGER
	NOT NULL	
	producto_id	INTEGER
	NOT NULL	

28. restaurantes	Columna	Tipo	Restricciones
	capacidad	INTEGER	NOT NULL
	estilo	VARCHAR2(50)	NOT NULL
	servicio_id	INTEGER	NOT NULL

29. restriccion_dias	Columna	Tipo	
	id	INTEGER	NOT NULL
	lunes	INTEGER	
	martes	INTEGER	
	miercoles	INTEGER	
	jueves	INTEGER	
	viernes	INTEGER	
	sabado	INTEGER	
	domingo	INTEGER	

30. rest_prods	Columna	Tipo
	restaurante_id	INTEGER
	NOT NULL	
	producto_id	INTEGER
	NOT NULL	

31. servicios	Columna	Tipo	Restricciones
	nombre	VARCHAR2(100 CHAR)	NOT NULL
	precio	NUMBER(12, 2)	NOT NULL
	tipo	VARCHAR2(30 CHAR)	NOT NULL

32. servicios_incluidos	Columna	Tipo	Restricciones
	plan_de_consumo	VARCHAR2(100 CHAR)	NOT NULL
	servicio_id	INTEGER	NOT NULL
	cantidad	NUMBER(12, 2)	NOT NULL

33. tipo_cliente	Columna	Tipo
	tipo	VARCHAR2(20 CHAR)
	NOT NULL	
	id	INTEGER
	NOT NULL	

34. tipo_servicio	Columna	Tipo
	servicio	VARCHAR2(30 CHAR)
	NOT NULL	
	id	INTEGER
	NOT NULL	

35. utensilios	Columna	Tipo	Restricciones
	nombre	VARCHAR2(50 CHAR)	NOT NULL
	condicion	VARCHAR2(30 CHAR)	NOT NULL
	servicio_id	INTEGER	NOT NULL
	id	INTEGER	NOT NULL

36. utensilio_prestado	Columna	Tipo
	utensilio_id	INTEGER
	NOT NULL	
	prestamo_id	INTEGER
	NOT NULL	

## 1.4 Explicacion FN

### Relación bares

A continuacion se presentan ejemplos de formas normales por cada relacion:

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que no hay dependencias parciales en ninguna clave candidata.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación bares\_productos

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria compuesta (bar\_id, producto\_id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación consumos

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación dotacion\_habitacion

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria compuesta (tipo\_habitacion, dotacion) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación dotaciones

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (nombre) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.



## Relación estancias

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación gimnasios

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (servicio\_id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación gym\_maqs

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria compuesta (maquina\_nombre, gimnasio\_id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación habitaciones

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (numero) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación indumentarias

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación internet

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (servicio\_id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación items

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación items\_consumidos

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria compuesta (consumo\_id, item\_id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación maquinas

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (nombre) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación ocupaciones

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria compuesta (habitacion\_numero, estancia\_id, fecha) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación planes\_de\_consumo

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (nombre) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación prestamos

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación productos

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (item\_id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación productos\_incluidos

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria compuesta (plan\_de\_consumo, producto\_id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación psicinas

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (servicio\_id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación reserva\_habs

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria compuesta (reserva\_id, habitacion\_numero) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación reservas\_estancias

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación reservas\_servicios

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria compuesta (numero\_habitacion, servicio\_id, fecha) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación rest\_prods

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria compuesta (restaurante\_id, producto\_id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación restaurantes

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (servicio\_id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación restricción\_dias

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria compuesta (servicio\_id, dia) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación restricción\_horas

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria compuesta (servicio\_id, hora\_inicio) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación restricción\_hora\_fin

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria compuesta (servicio\_id, hora\_fin) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación servicios

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## Relación usuarios

1. Primera Forma Normal (1FN): Sí, ya que no contiene atributos compuestos y todos los atributos son atómicos.
2. Segunda Forma Normal (2FN): Sí, ya que la relación tiene una clave primaria (id) y no hay dependencias parciales.
3. Tercera Forma Normal (3FN): Sí, ya que no hay dependencias transitivas en ninguna clave candidata.

## 1.5 Nivel de Normalización

El modelo de base de datos ha sido normalizado hasta la Forma Normal de Boyce-Codd (FN BC). Se ha verificado la cumplimiento de esta forma normal para todas las relaciones.

## 1.6 Conclusiones

Basado en la verificación de las formas normales de las relaciones en el modelo de base de datos proporcionado, se pueden obtener las siguientes conclusiones:

1. **Nivel de Normalización:** Todas las relaciones en el modelo de base de datos están en Tercera Forma Normal (3FN). Esto indica un alto nivel de normalización, lo cual es positivo para la integridad y eficiencia de la base de datos.
2. **Reducción de Redundancia:** La normalización hasta la Tercera Forma Normal (3FN) ha ayudado a reducir la redundancia de los datos. Esto significa que la información se almacena de manera eficiente y no hay duplicación innecesaria de la misma.
3. **Integridad de los Datos:** La normalización contribuye a la integridad de los datos al eliminar dependencias funcionales no deseadas. Esto garantiza que la información se mantenga precisa y coherente a lo largo del tiempo.
4. **Eficiencia en la Manipulación de Datos:** La estructura normalizada facilita la inserción, actualización y eliminación de registros en la base de datos. Esto mejora la eficiencia en la gestión y manipulación de los datos almacenados.
5. **Claridad en el Diseño:** La base de datos normalizada presenta una estructura más clara y organizada. Cada relación y sus atributos tienen un propósito específico, lo que facilita la comprensión y el mantenimiento del sistema.
6. **Escalabilidad:** La normalización adecuada puede facilitar la expansión y adaptación del sistema en el futuro. Al tener una base sólida, es más sencillo agregar nuevas funcionalidades o relaciones sin comprometer la integridad de los datos existentes.
7. **Cumplimiento de Buenas Prácticas:** La normalización hasta la Tercera Forma Normal (3FN) es una práctica recomendada en el diseño de bases de datos relacionales. Cumplir con estas normas contribuye a la robustez y calidad del sistema.

En resumen, el modelo de base de datos presentado muestra un diseño bien estructurado y normalizado, lo cual es fundamental para garantizar la integridad y eficiencia de la base de datos a lo largo del tiempo.