

Christian Chavarro Espejo  
Juan Sanmiguel Mateus

201613724  
201617603

## 1. Análisis

Si lo requiere, ajuste el modelo del mundo (modelo conceptual: diagrama de clases UML) propuesto en la iteración 2 que representó el caso de estudio. Indique cuáles clases del modelo del mundo fueron actualizadas o creadas en esta iteración.

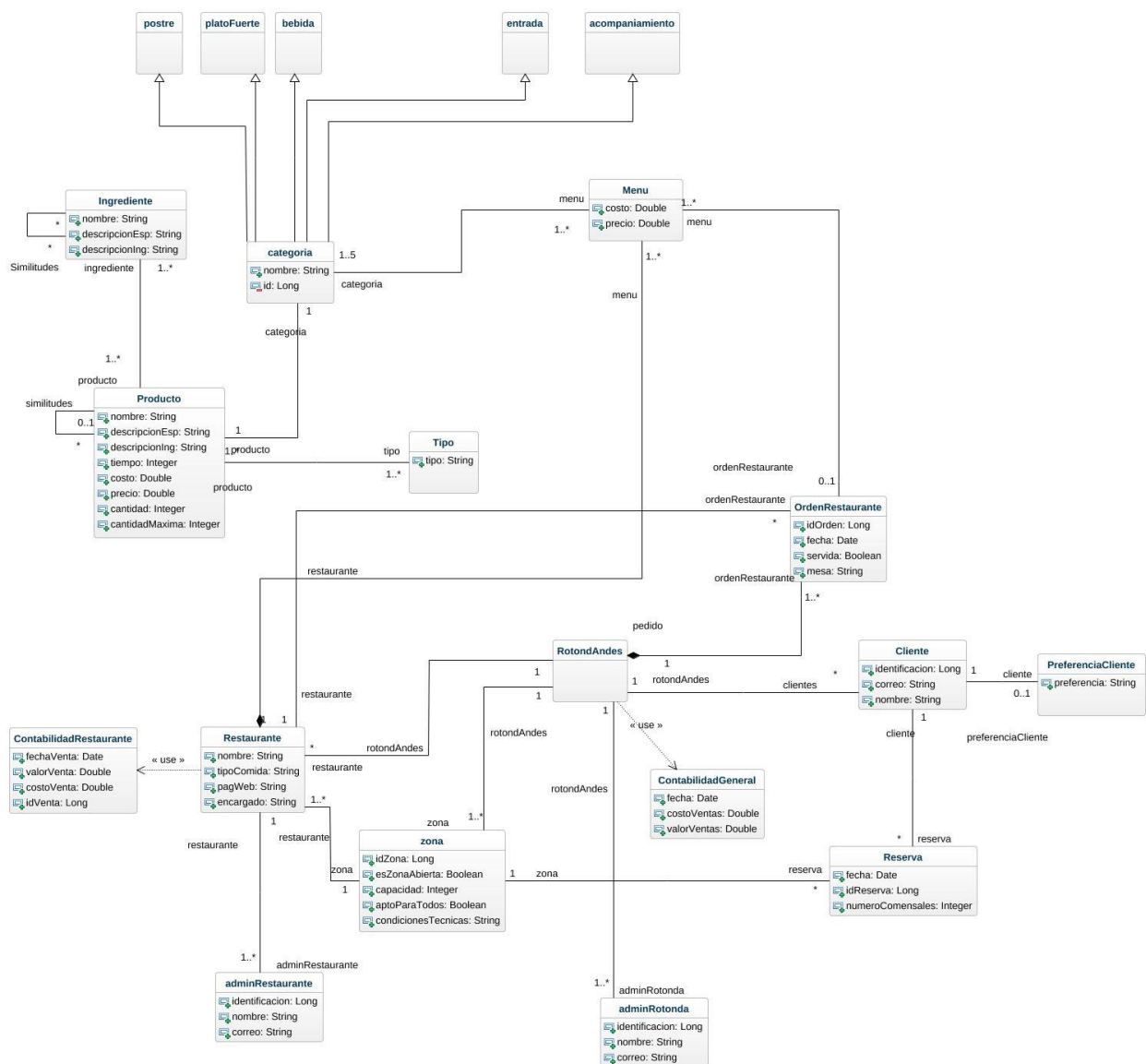


Figura 1. Modelo conceptual planteado para RotondAndes en este caso de estudio.

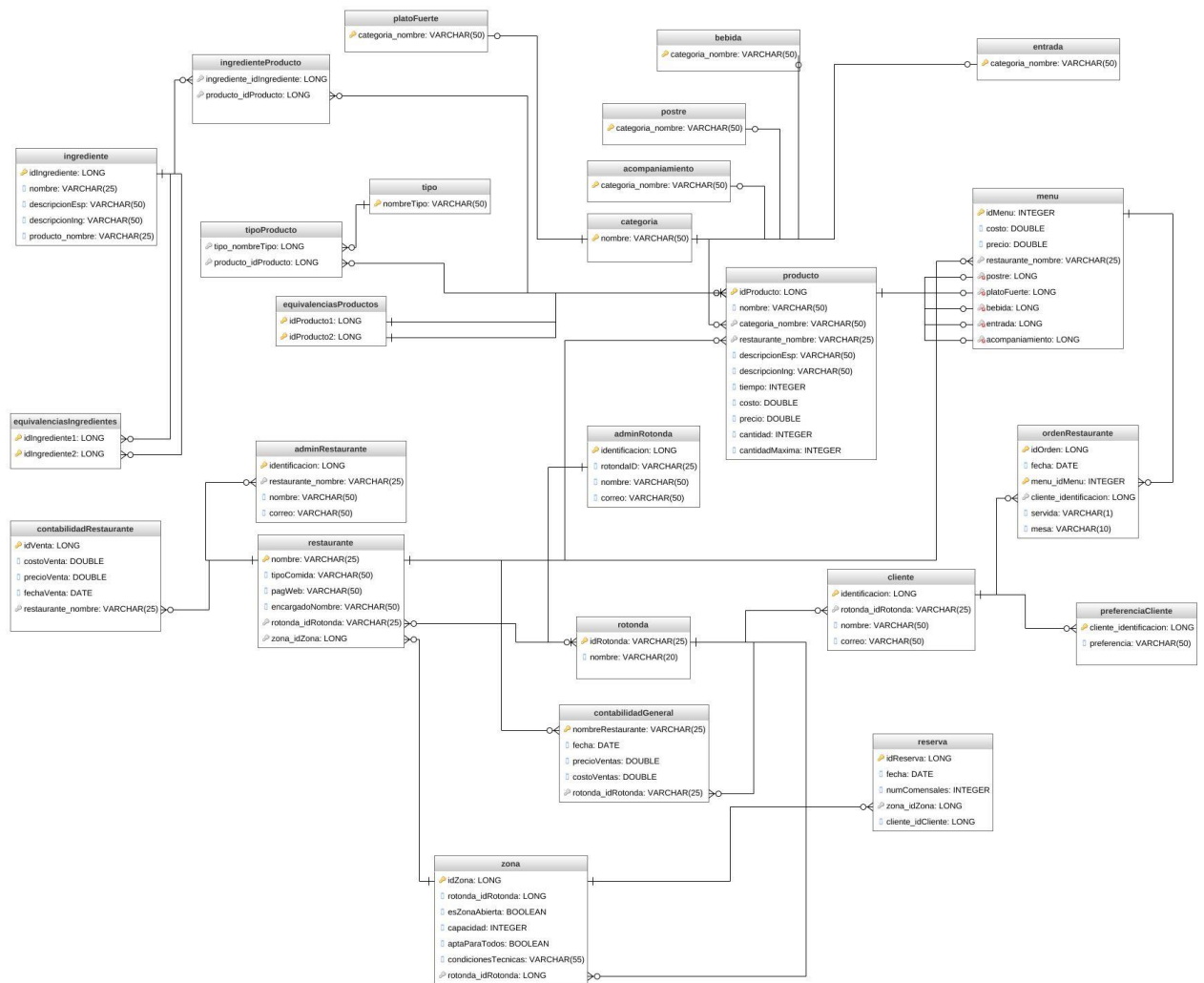


Figura 2. Modelo lógico para RotondAndes, diseñado a partir del modelo conceptual anterior. Para el caso de estudio 3. Basado en el caso de estudio anterior.

Respecto a las nuevas necesidades propuestas en la iteración 3 y contrastando el modelo del mundo del pasado caso de estudio, estos son los cambios. El modelo del mundo correspondiente a la iteración anterior, se encuentra en la subcarpeta diagramas en la carpeta docs del proyecto.

Clases actualizadas o modificadas:

- OrdenRestaurante: El atributo mesa fue agregado, esto, en función de los pedidos dirigidos a varios comensales en mesas.
- Producto: Con el objetivo de surtir de manera eficaz los restaurantes, el atributo cantidadMaxima fue agregado a esta clase.

Asociaciones agregadas:

- Ingrediente-Ingrediente y Producto-Producto: Para registrar las similitudes entre ingredientes o productos, se añadió la relación hacia ingrediente desde la clase ingrediente, igualmente con producto.

Las descripciones anteriores, representan los cambios realizados en el modelo del mundo respecto al caso de estudio anterior.

## 2. Diseño de la aplicación

- A partir del diseño existente, analice el impacto que representa la introducción de los nuevos requerimientos y restricciones a nivel del modelo conceptual.

Partiendo del diseño conceptual realizado en la iteración II y las agregaciones especificadas en el párrafo anterior, los requerimientos introducidos para este nuevo caso de estudio, se pueden realizar de manera escalable en la aplicación. Es decir, no es necesario realizar mayores cambios en lo preexistente para garantizar el desempeño de la Rotonda.

- Realice los cambios necesarios en su modelo relacional para respetar las reglas de negocio y asegurar la calidad del mismo.

Tal como fue descrito anteriormente, los cambios realizados en el modelo relacional fueron los siguientes.

Tablas agregadas:

La primera tabla que se agregó, es la de equivalencias entre productos.

Nombre	EquivalenciasProductos		
Descripción	Registra las equivalencias entre productos.		
Llave	IdProducto1, IdProducto2		
Atributo	Tipo	Descripción	Características/Restricciones
IdProducto1	Number	Id primer producto.	NN, FK Tproducto.id
IdProducto2	Number	Id segundo producto.	NN, FK Tproducto.id

Tabla 1: descripción tabla EquivalenciasProductos.

La segunda tabla en agregarse, fue la correspondiente a las equivalencias entre ingredientes.

Nombre	EquivalenciasIngredientes		
Descripción	Registra las equivalencias entre ingredientes.		
Llave	IdIngrediente1, IdIngrediente2		
Atributo	Tipo	Descripción	Características/Restricciones
IdIngrediente1	Number	Id primer ingrediente.	NN, FK Tingrediente.id
IdIngrediente2	Number	Id segundo ingrediente.	NN, FK Tingrediente.id

Tabla 2: descripción tabla EquivalenciasIngredientes.

Atributos, columnas, agregados a tablas preexistentes:

- Cantidad Máxima: en la tabla producto, con el fin de surtir los productos de un restaurante.
- Mesa: en la tabla correspondiente a las órdenes. Para así asegurar que los pedidos están relacionados a una mesa y varios clientes.

Las descripciones anteriores, representan los cambios realizados entre las dos versiones del caso de estudio de la rotonda.

- Documente el diseño y las decisiones tomadas para crear los elementos de la base de datos que da el respaldo de persistencia a la aplicación, a partir del modelo conceptual.

A la hora de persistir objetos de la rotonda en la DB, se tiene en cuenta que:

- Sea un objeto esencial del mundo, un producto, una zona, una similitud de ingredientes.
- En algún punto se relacionará con otro objeto, los restaurantes de una zona, los administradoresRestaurante y el restaurante.

Notas sobre el diseño:

- Para el diseño conceptual actual de RotondAndes, todas las clases pasan a la base de datos. En el caso de Rotonda, solo se persiste un id y nombre, esto, esperando que la aplicación sea escalable a más rotondas.
- En el sistema de Rotondas, no se tienen en cuenta las mesas como objeto, estas mesas, solo se usarán como atributo en la tabla de orden y servirán como un filtro. Así, cualquier orden se puede entregar a cualquier mesa y evitar conflictos como: capacidad de la mesa, la mesa está en otra zona.

5. Incluya un listado con las tablas generadas en la base de datos, utilizando los estándares establecidos, disponibles en la wiki del curso (sección tutoriales).

Este listado, resultado de una consulta SQL, debe incluir el nombre de la tabla, el nombre y el tipo de dato de sus campos, así como los nombres de restricciones de llaves primarias, llaves foráneas y de chequeo.

A la hora de realizar las consultas en las tablas, se realizó la siguiente consulta para cada una de ellas:

```
DESCRIBE {NOMBRETABLA}  
SELECT CONSTRAINT_NAME, CONSTRAINT_TYPE,  
SEARCH_CONDITION, R_CONSTRAINT_NAME  
FROM ALL_CONSTRAINTS  
WHERE TABLE_NAME ='{NOMBREDELATABLA} ';
```

Ahora, se muestra el listado de tablas de acuerdo al esquema presentado en el curso y el resultado de esta consulta para cada tabla.

## AdministradorRestaurante

Nombre	AdministradorRestaurante.		
Descripción	Guarda los administradores de los restaurantes.		
Llave	Cedula		
Atributo	Tipo	Descripción	Características/Restricciones
Cedula	Number	Id admin.	NN, PK
Nombre	Varchar 20	N. admin.	NN
Correo	Varchar 20	Correo	NN
NombreRestaurante	Varchar 20	N. rest.	NN, FK Trestaurante.nombre

Tabla 3: Descripción de la tabla AdministradorRestaurante.

The screenshot shows two windows of the SQL Rotonda application. The left window displays the results of a query: `select CONSTRAINT_NAME, CONSTRAINT_TYPE, SEARCH_CONDITION, R_CONSTRAINT_NAME from all_constraints where TABLE_NAME = 'ADMINISTRADORRESTAURANTE';`. The results are as follows:

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_CONSTRAINT_NAME
1 TABLE1_FK	P	(null)	(null)
2 ADMINISTRADORRESTAURANTE_FK1	R	(null)	RESTAURANTE_FK
3 SYS_C00150468	C	"NOMBRE_RESTAURANTE" IS NOT NULL	(null)
4 SYS_C00150467	C	"CORREO" IS NOT NULL	(null)
5 SYS_C00150466	C	"NOMBRE" IS NOT NULL	(null)
6 SYS_C00150463	C	"CEDULA" IS NOT NULL	(null)

The right window shows the output of the `describe ADMINISTRADORRESTAURANTE` command:

Nombre	¿Nulo?	Tipo
CEDULA	NOT NULL	NUMBER
NOMBRE	NOT NULL	VARCHAR2 (30)
CORREO	NOT NULL	VARCHAR2 (50)
NOMBRE_RESTAURANTE	NOT NULL	VARCHAR2 (50)

Imagen 1: Resultados de la consulta para AdministradorRestaurante.

## AdministradorRotonda

Nombre	AdministradorRotonda.		
Descripción	Guarda los administradores de la rotonda.		
Llave	Cedula		
Atributo	Tipo	Descripción	Características/Restricciones
Cedula	Number	Id admin.	NN, PK
Nombre	Varchar 20	Nom.admin.	NN
Correo	Varchar 20	Correo	NN
IdRotonda	Number	Id rot.	NN, FK Trotondandes.id

Tabla 4: Descripción de la tabla AdministradorRotonda.

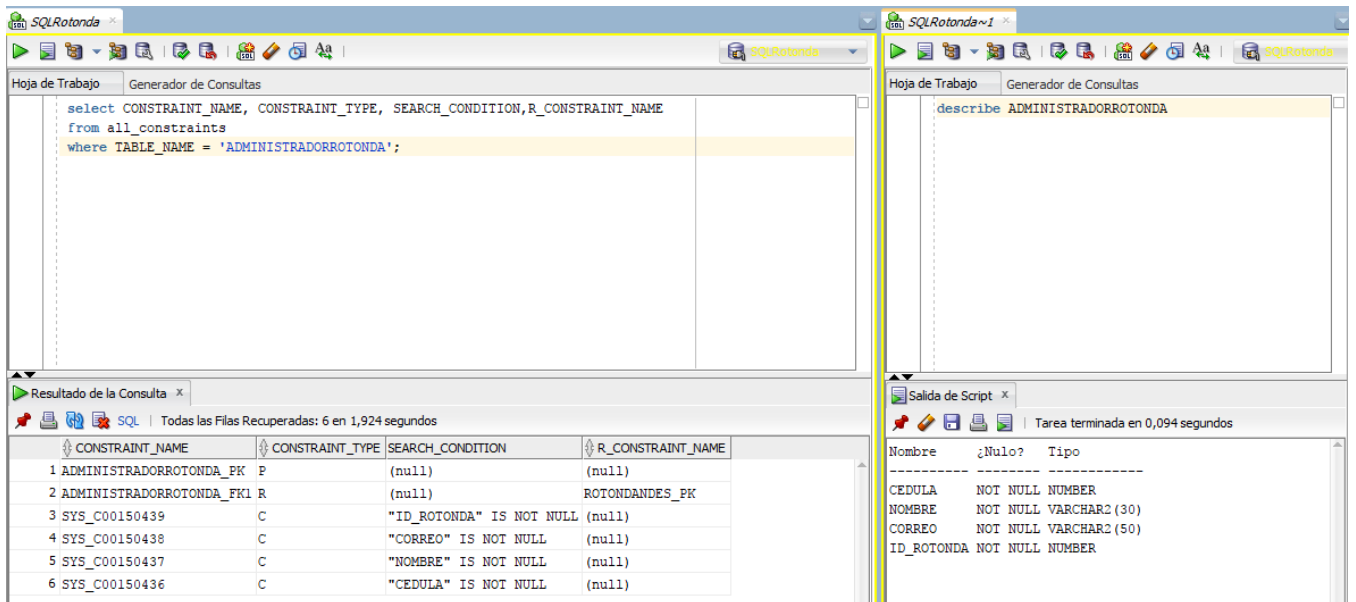


Imagen 2: Resultados de la consulta para AdministracionRotonda.

## Categoría

Nombre	Categoría		
Descripción	Registra las categorías		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
id	Number	id cat.	NN, PK
Categoría	Varchar 20	Nombre	NN

Tabla 5: Descripción de la tabla Categoría.

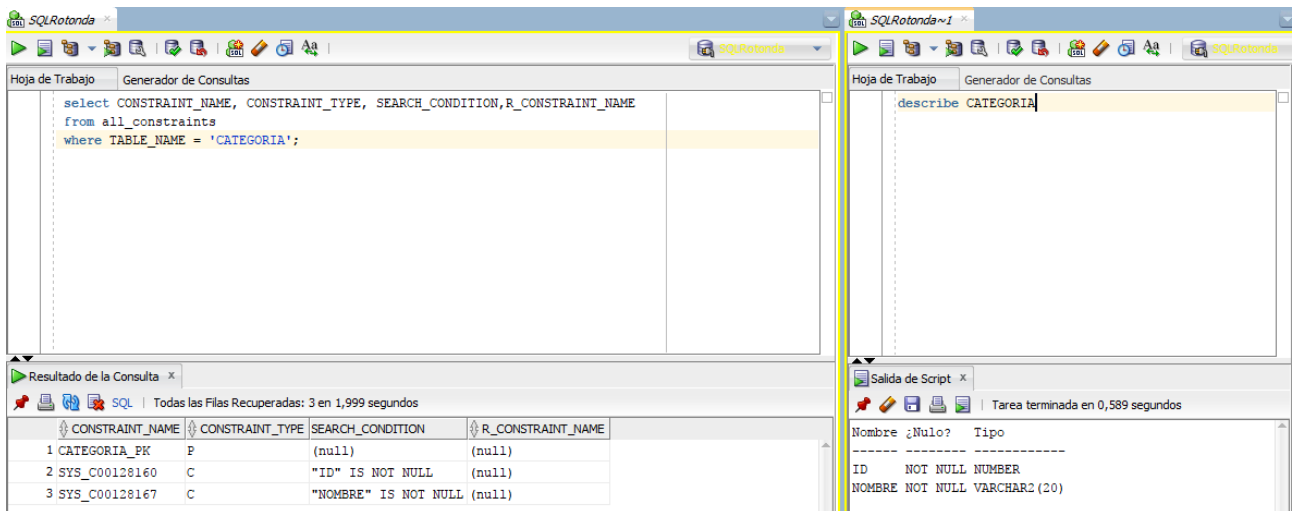


Imagen 3: Resultados de la consulta para Categoría.

## Cliente

Nombre	Cliente		
Descripción	Registra los clientes de la rotonda.		
Llave	Cedula		
Atributo	Tipo	Descripción	Características/Restricciones
Cedula	Number	Id cliente.	NN, PK
Nombre	Varchar 20	Nombre.	NN
Correo	Varchar 20	Correo cli.	NN
IdRotonda	Number	Id rotonda	NN, FK Trotondandes.id

Tabla 8: Descripción de la tabla Cliente.

The screenshot shows two windows of the SQLRotonda application. The left window displays the results of a query: `select CONSTRAINT_NAME, CONSTRAINT_TYPE, SEARCH_CONDITION, R_CONSTRAINT_NAME from all_constraints where TABLE_NAME = 'CLIENTE';`. The results are shown in a table with 6 rows. The right window displays the output of a `describe CLIENTE` command, showing the table's structure with columns: Nombre (Varchar2(50)), CEDULA (Number), CORREO (Varchar2(50)), and ID\_ROTONDA (Number).

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_CONSTRAINT_NAME
1 SYS_C00128165	C	"ID_ROTONDA" IS NOT NULL	(null)
2 SYS_C00128164	C	"CORREO" IS NOT NULL	(null)
3 SYS_C00128163	C	"NOMBRE" IS NOT NULL	(null)
4 SYS_C00128162	C	"CEDULA" IS NOT NULL	(null)
5 CLIENTE_FK1	R	(null)	ROTONDANDES_FK
6 CLIENTE_PK	P	(null)	(null)

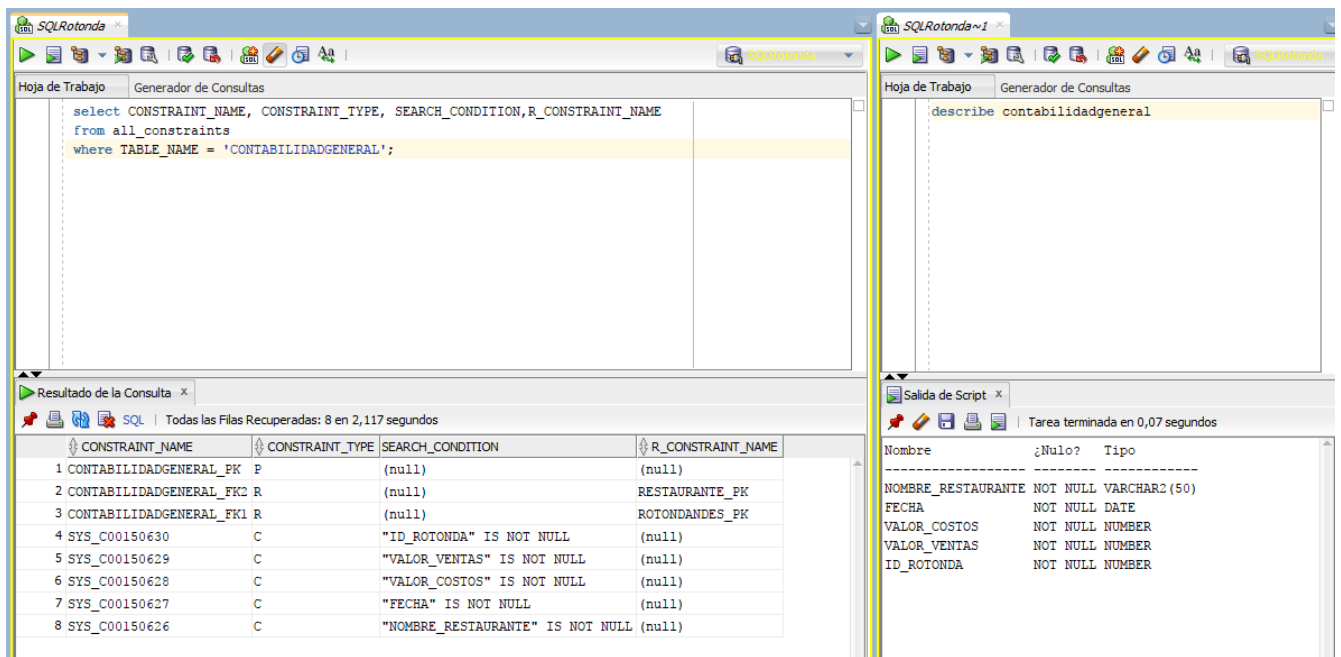
Nombre	Nulo?	Tipo
CEDULA	NOT NULL	NUMBER
NOMBRE	NOT NULL	VARCHAR2(50)
CORREO	NOT NULL	VARCHAR2(50)
ID_ROTONDA	NOT NULL	NUMBER

Imagen 4: Resultados de la consulta para Cliente.

## ContabilidadGeneral

Nombre	ContabilidadGeneral		
Descripción	Guarda las ventas realizadas en la rotonda		
Llave	Nombre_restaurante, fecha		
Atributo	Tipo	Descripción	Características/Restricciones
Nombre_restaurante	Varchar 20	Nom.tres	NN, PK, FK restaurante.nombre
Fecha	DATE	Fecha cont.	NN, PK
Valor costos	Number	Costo vent.	NN
valor ventas	Number	Pre. ventas	NN
idRotonda	Number	rotoda	NN, Fkrotonda.id

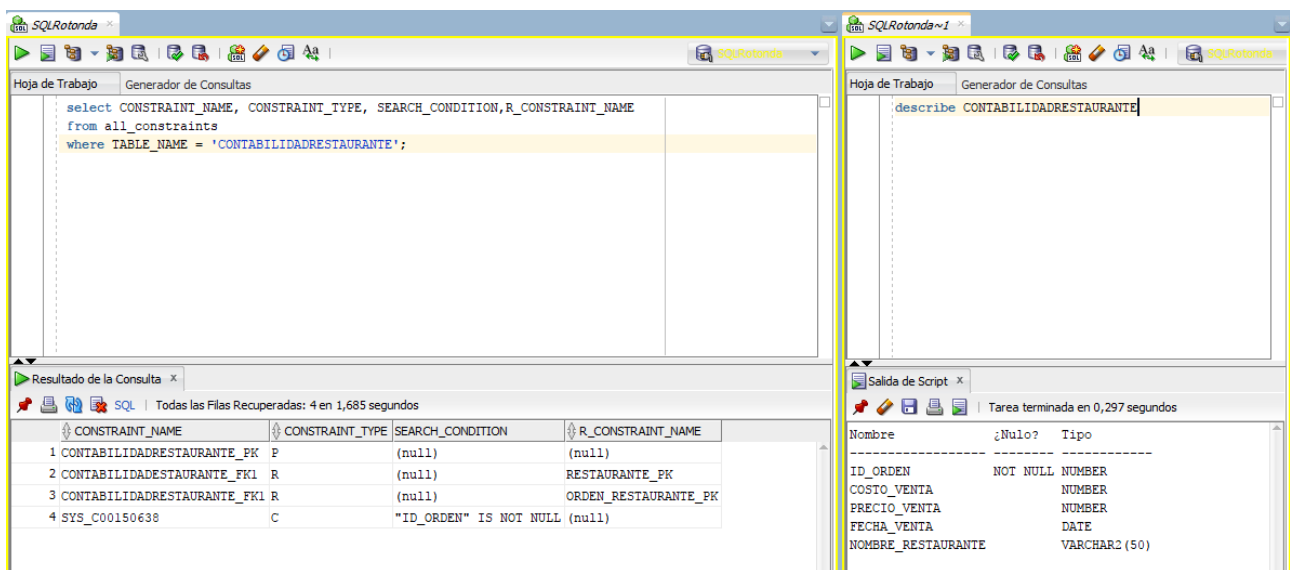
Tabla 9: Descripción de la tabla ContabilidadGeneral.



## ContabilidadRestaurante

Nombre	ContabilidadRestaurante		
Descripción	Guarda las ventas realizadas en un restaurante.		
Llave	idOrden		
Atributo	Tipo	Descripción	Características/Restricciones
IdOrden	Number	id orden	NN, PK
Costo Venta	Number	Costo.	NN
Precio Venta	Number	Precio.	NN
Fecha Venta	DATE	Fecha.	NN
Nombre R	Varchar 20	N. resta,	NN, Fkrestaurante.nombre

Tabla 10: Descripción de la tabla ContabilidadRestaurante.

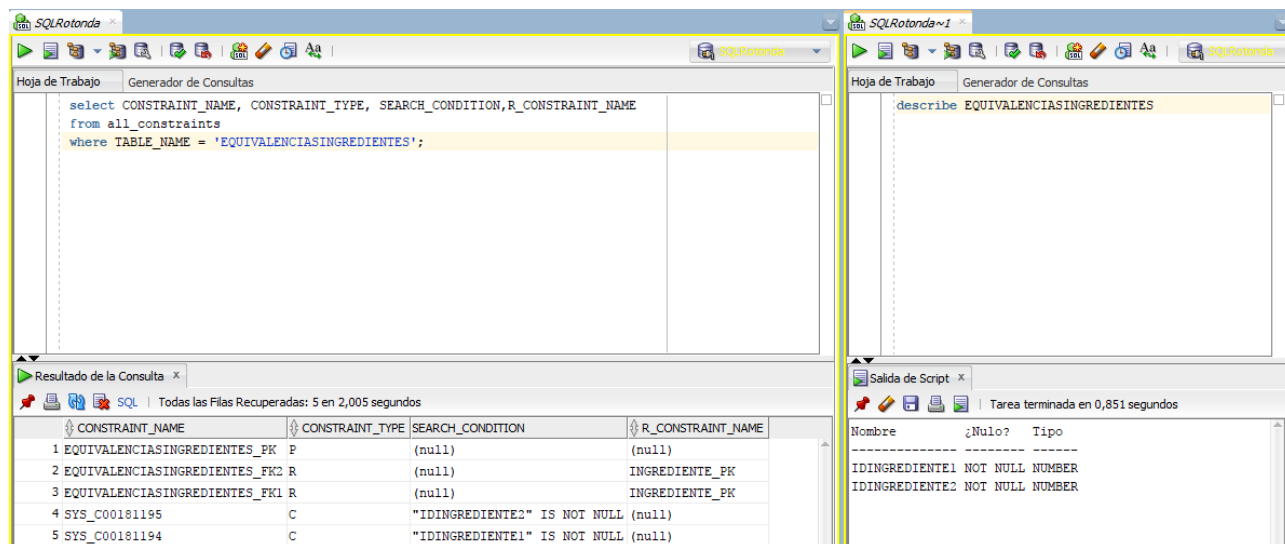




## EquivalenciasIngredientes

Nombre	EquivalenciasIngredientes		
Descripción	Registra las equivalencias entre ingredientes.		
Llave	IdIngrediente1, IdIngrediente2		
Atributo	Tipo	Descripción	Características/Restricciones
IdIngrediente1	Number	Id primer ingrediente.	NN, FK Tingrediente.id, PK
IdIngrediente2	Number	Id segundo ingrediente.	NN, FK Tingrediente.id, PK

Tabla 11: Descripción de la tabla EquivalenciasIngredientes.



SQLRotonda

Hoja de Trabajo Generador de Consultas

```
select CONSTRAINT_NAME, CONSTRAINT_TYPE, SEARCH_CONDITION, R_CONSTRAINT_NAME
from all_constraints
where TABLE_NAME = 'EQUIVALENCIASINGREDIENTES';
```

Resultado de la Consulta

Todas las Filas Recuperadas: 5 en 2,005 segundos

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_CONSTRAINT_NAME
1 EQUIVALENCIASINGREDIENTES_PK	P	(null)	(null)
2 EQUIVALENCIASINGREDIENTES_FK2	R	(null)	INGREDIENTE_PK
3 EQUIVALENCIASINGREDIENTES_FK1	R	(null)	INGREDIENTE_PK
4 SYS_C00181195	C	"IDINGREDIENTE2" IS NOT NULL	(null)
5 SYS_C00181194	C	"IDINGREDIENTE1" IS NOT NULL	(null)

Salida de Script

Tarea terminada en 0,851 segundos

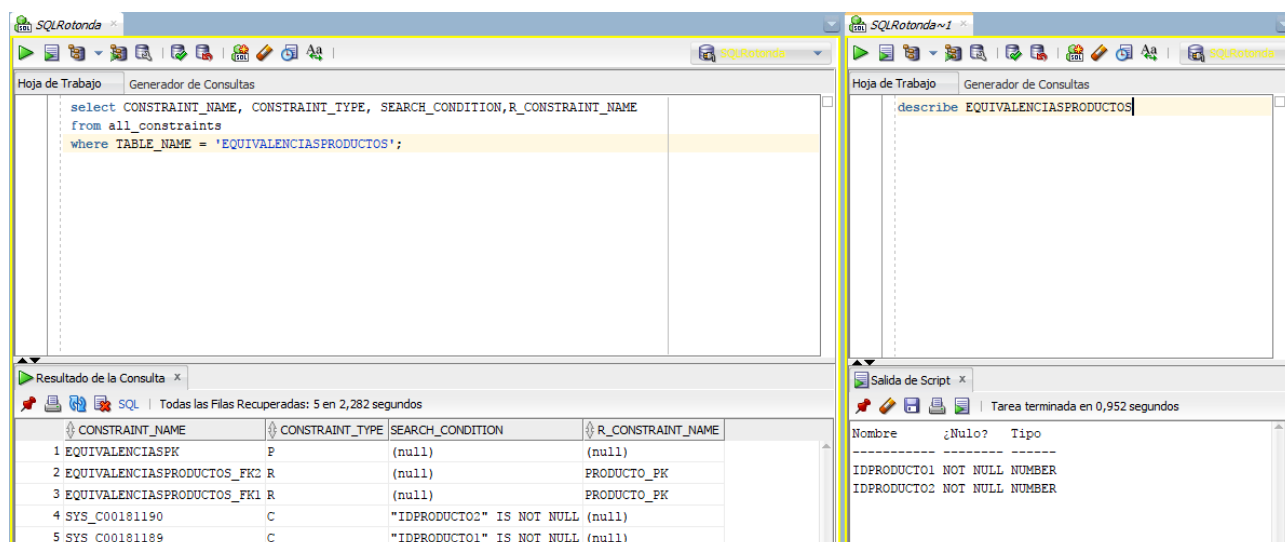
Nombre	¿Nulo?	Tipo
IDINGREDIENTE1	NOT NULL	NUMBER
IDINGREDIENTE2	NOT NULL	NUMBER

Imagen 7: Resultados de la consulta para EquivalenciasIngredientes.

## EquivalenciasProductos

Nombre	EquivalenciasProductos		
Descripción	Registra las equivalencias entre productos.		
Llave	IdProducto1, IdProducto2		
Atributo	Tipo	Descripción	Características/Restricciones
IdProducto1	Number	Id primer producto.	NN, FK Tproducto.id, PK
IdProducto2	Number	Id segundo producto.	NN, FK Tproducto.id, PK

Tabla 12: Descripción de la tabla EquivalenciasProductos.



SQLRotonda

Hoja de Trabajo Generador de Consultas

```
select CONSTRAINT_NAME, CONSTRAINT_TYPE, SEARCH_CONDITION, R_CONSTRAINT_NAME
from all_constraints
where TABLE_NAME = 'EQUIVALENCIASPRODUCTOS';
```

Resultado de la Consulta

Todas las Filas Recuperadas: 5 en 2,282 segundos

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_CONSTRAINT_NAME
1 EQUIVALENCIASPK	P	(null)	(null)
2 EQUIVALENCIASPRODUCTOS_FK2	R	(null)	PRODUCTO_PK
3 EQUIVALENCIASPRODUCTOS_FK1	R	(null)	PRODUCTO_PK
4 SYS_C00181190	C	"IDPRODUCTO2" IS NOT NULL	(null)
5 SYS_C00181189	C	"IDPRODUCTO1" IS NOT NULL	(null)

Salida de Script

Tarea terminada en 0,952 segundos

Nombre	¿Nulo?	Tipo
IDPRODUCTO1	NOT NULL	NUMBER
IDPRODUCTO2	NOT NULL	NUMBER

Imagen 8: Resultados de la consulta para EquivalenciasProductos.

Ingrediente

Nombre	Ingrediente		
Descripción	Guarda los ingredientes de la rotonda		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
idRotonda	Number	id del ing	NN, PK
Nombre	Varchar 20	Nom. ing.	NN
Des. Español.	Varchar 20	Desp. Esp.	NN
Des. Inglés.	Varchar 20	desp. lsp.	NN

Tabla 13: Descripción de la tabla Ingredientes.

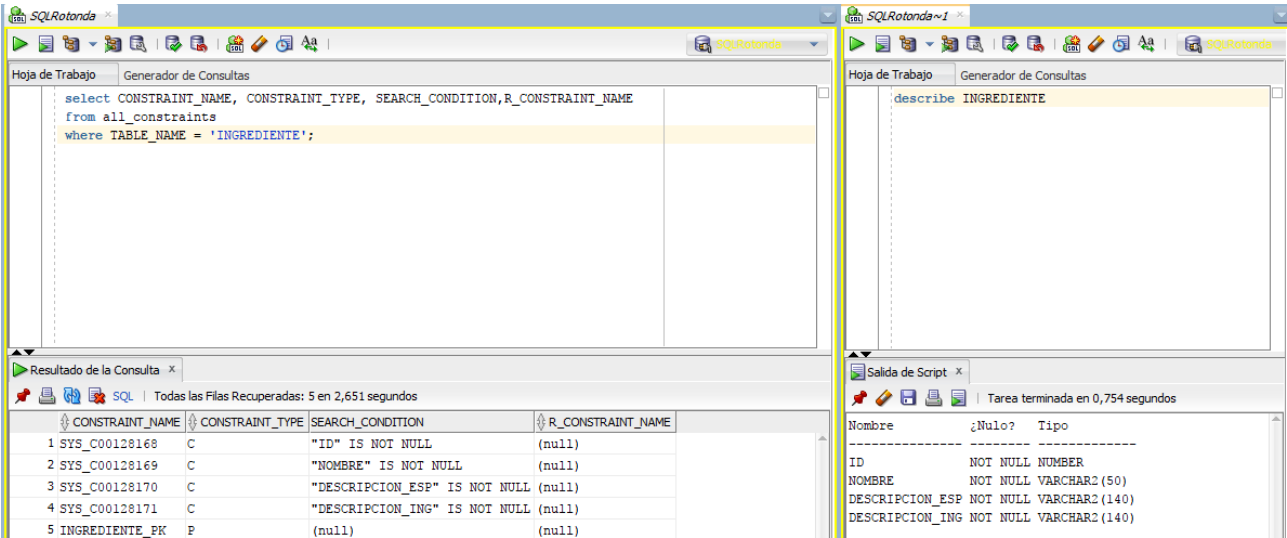


Imagen 9: Resultados de la consulta para Ingredientes.

IngredienteProducto

Nombre	IngredienteProducto		
Descripción	Guarda las relaciones entre el ingrediente y los productos.		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
idIngrediente	Number	id ing.	NN, PK, Fkingrediente.id
idProducto	Number	id prod.	NN, PK, Fkproducto.id

Tabla 14: Descripción de la tabla IngredienteProducto.

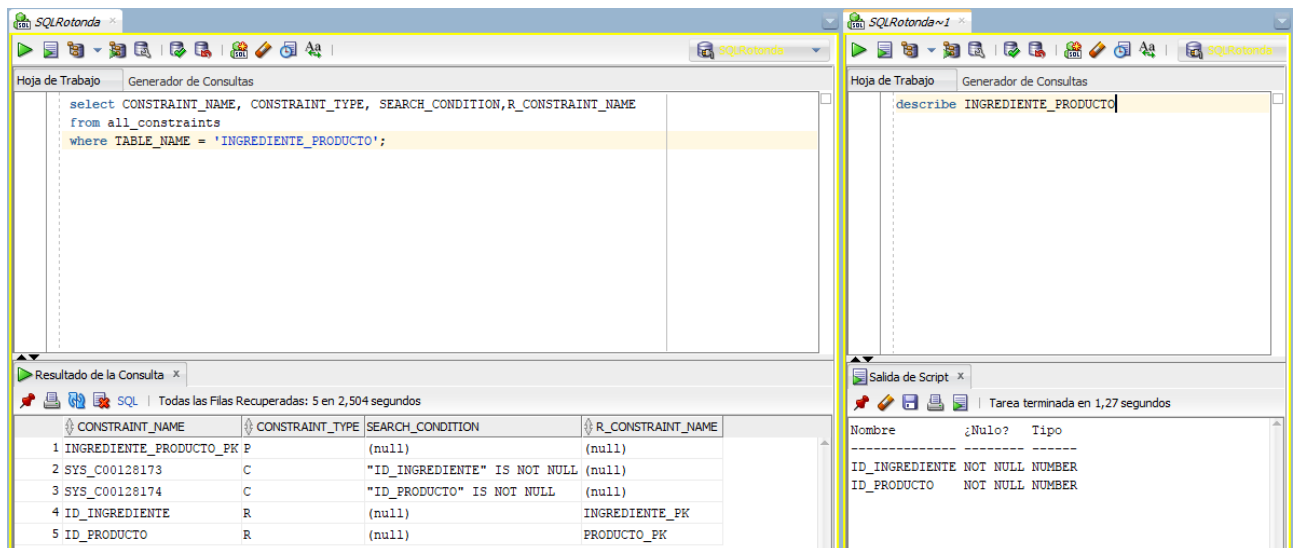
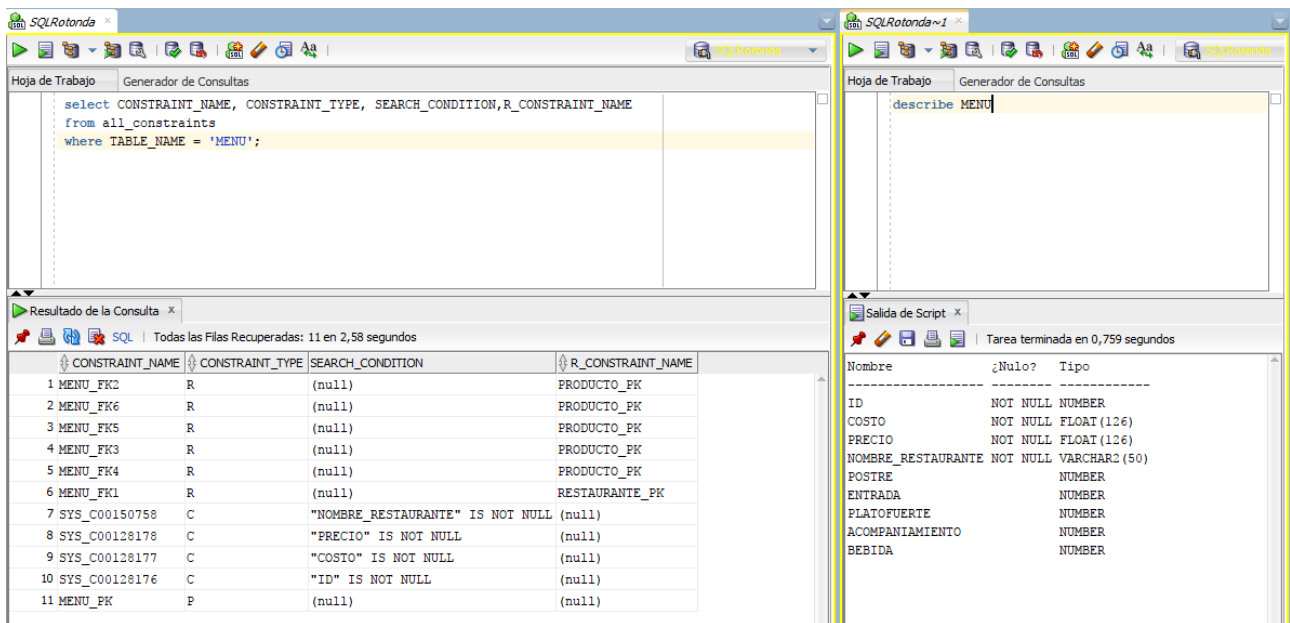


Imagen 10: Resultados de la consulta para IngredienteProducto.

### Menú

Nombre	Menú		
Descripción	Representa los menús de los restaurantes		
Llave	idMenu		
Atributo	Tipo	Descripción	Características/Restricciones
idMenu	Number	identificador	NN, PK
Costo	Number	Costo men.	NN
Precio	Number	Precio men.	NN
N. Restaurante	Varchar 20	Restaurante	NN, Fkrestaurante.nombre
Postre	Number	id prod.	NN, Fkproducto.id
Entrada	Number	id prod.	NN, Fkproducto.id
PlatoFuerte	Number	id prod.	NN, Fkproducto.id
Acomp	Number	id prod.	NN, Fkproducto.id
Bebida	Number	id prod.	NN, Fkproducto.id

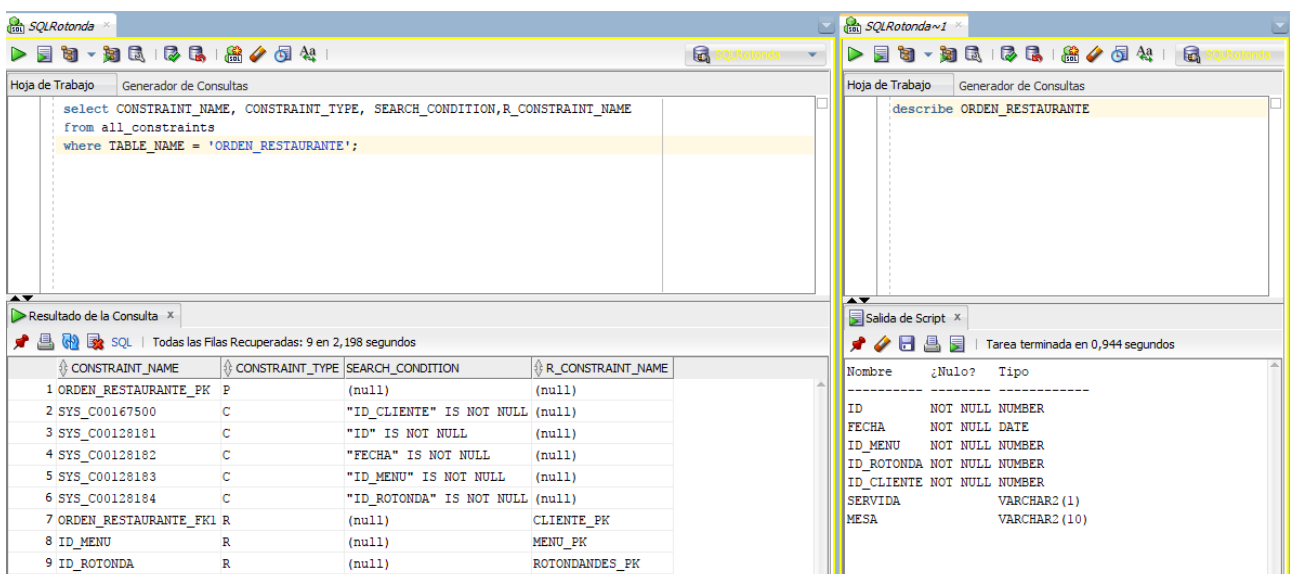
Tabla 15: Descripción de la tabla Menú.



## OrdenRestaurante

Nombre	OrdenRestaurante		
Descripción	Representa las ordenes en la rotonda		
Llave	idMenu		
Atributo	Tipo	Descripción	Características/Restricciones
idOrden	Number	id	NN, PK
Fecha	DATE	fecha	NN
IdMenu	Number	Menú.	NN, Fkmenu.id
IdRotonda	Number	Rotonda.	NN, Fkrotonda.id
IdCliente	Number	Cliente.	Fkcliente.cedula
Servida	Varchar 1	¿Está lista?	NN
Mesa	Varchar 5	Mesa orden.	NN, Fkproducto.id

Tabla 16: Descripción de la tabla OrdenRestaurante.



## Cliente Preferencia

Nombre	Cliente preferencia		
Descripción	Guarda las preferencias de los clientes		
Llave	idCliente		
Atributo	Tipo	Descripción	Características/Restricciones
IdCliente	Number	Id.	NN, PK,Fkcliente.cedula
Preferencia	Varchar 20	Preferen.	NN

Tabla 17: Descripción de la tabla Preferencia Cliente.

The screenshot shows the SQLRotonda interface. On the left, a query window displays the SQL statement: `select CONSTRAINT_NAME, CONSTRAINT_TYPE, SEARCH_CONDITION, R_CONSTRAINT_NAME from all_constraints where TABLE_NAME = 'PREFERENCIACLIENTE';`. Below it, the 'Resultado de la Consulta' (Query Result) pane shows the following data:

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_CONSTRAINT_NAME
1 PREFERENCIACLIENTE_PK	P	(null)	(null)
2 PREFERENCIACLIENTE_FK1	R	(null)	CLIENTE_PK
3 SYS_C00153280	C	"ID_CLIENTE" IS NOT NULL (null)	

On the right, another window shows the 'describe PREFERENCIACLIENTE' command. Below it, the 'Salida de Script' (Script Output) pane displays the table structure:

Nombre	¿Nulo?	Tipo
ID_CLIENTE	NOT NULL	NUMBER
PREFERENCIA		VARCHAR2 (50)

Imagen 13: Resultados de la consulta para Preferencia Cliente.

## Producto

Nombre	Producto		
Descripción	Guarda los productos.		
Llave	idProducto		
Atributo	Tipo	Descripción	Características/Restricciones
IdProducto	Number	id	NN, PK
Nombre	Varchar 20	nombre	NN
IdCategoria	Number	cat. Prod.	NN, Fkcategoria.id
D. Esp.	Varchar 20	desc.	NN
D. Ing.	Varchar 20	desc.	NN
Tiempo	Number	Tiem. P.	NN
Costo	Number	costo.	NN
Precio	Number	Precio	NN
cantidad	Number	Cantidad.	NN
Restaurante	Varchar 20	N. resta.	NN, Fkrestaurantes.nombre
Cant. Máxima.	Number	Max. Cant.	NN

Tabla 18: Descripción de la tabla Producto.

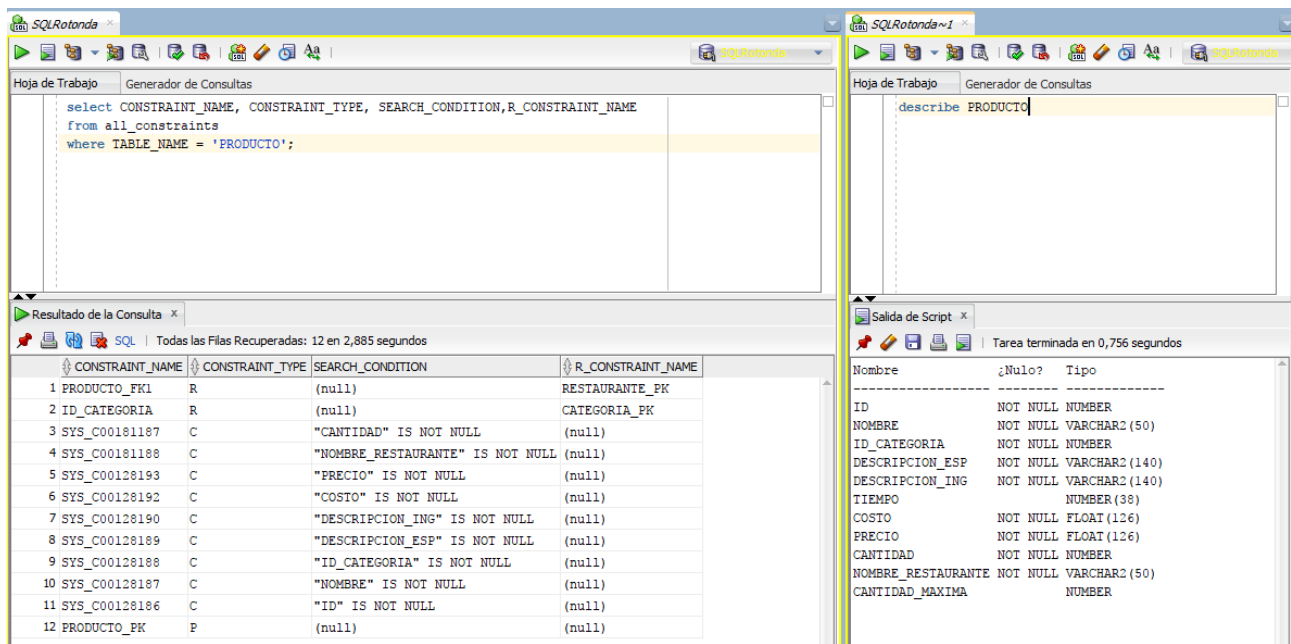


Imagen 14: Resultados de la consulta para Producto.

## Reserva

Nombre	Reserva		
Descripción	Guarda las reservas.		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
id	Number	id res.	NN, PK
Fecha	DATE	fecha	NN
Num_Comensales	Number	cant. Cli.	NN
Id_Cliente	Number	cliente	NN, Fkzcliente.cedula
Id_zona	Number	zona res.	NN, Fkzona.id

Tabla 19: Descripción de la tabla Reserva.

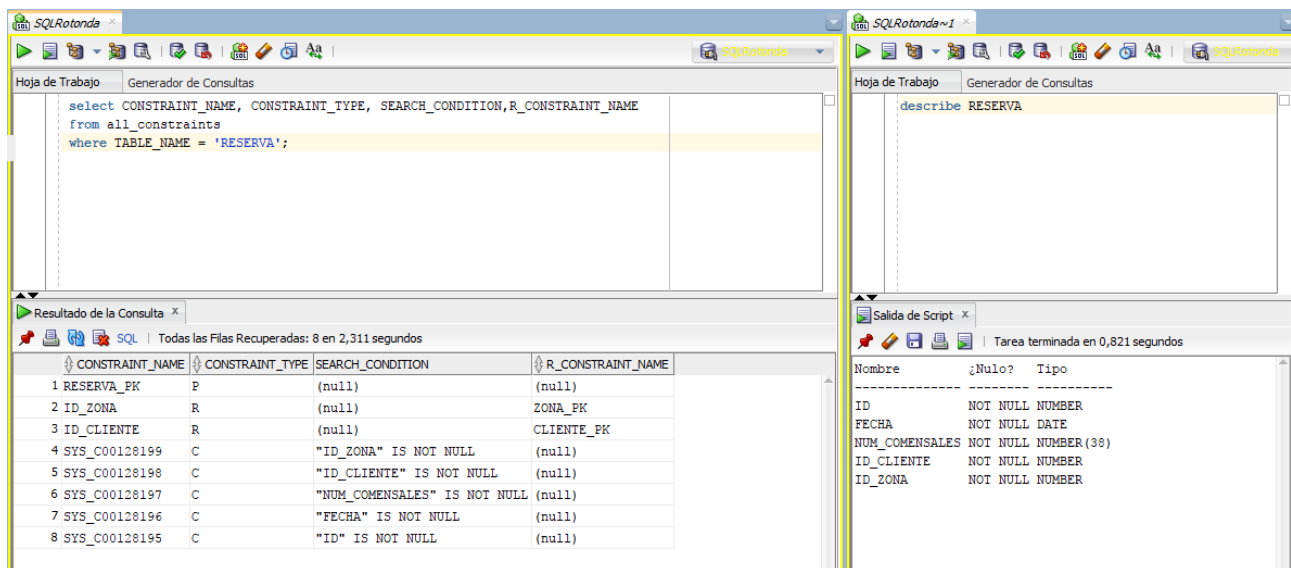


Imagen 15: Resultados de la consulta para Reserva.

# Restaurante

Nombre	Restaurante		
Descripción	Guarda los restaurantes.		
Llave	nombre_restaurante		
Atributo	Tipo	Descripción	Características/Restricciones
nombre_restaurante	Varchar 20	id res.	NN, PK
Tipo_comida	Varchar 20	Comida.	NN
Pag_Web	Varchar 100	web	NN
Encargado	Varchar 20	Gerente.	NN
Id_zona	Number	Zona.	NN, Fkzona.id
id_rotonda	Number	rotonda	NN, Fkrotonda.id

Tabla 20: Descripción de la tabla Restaurante.

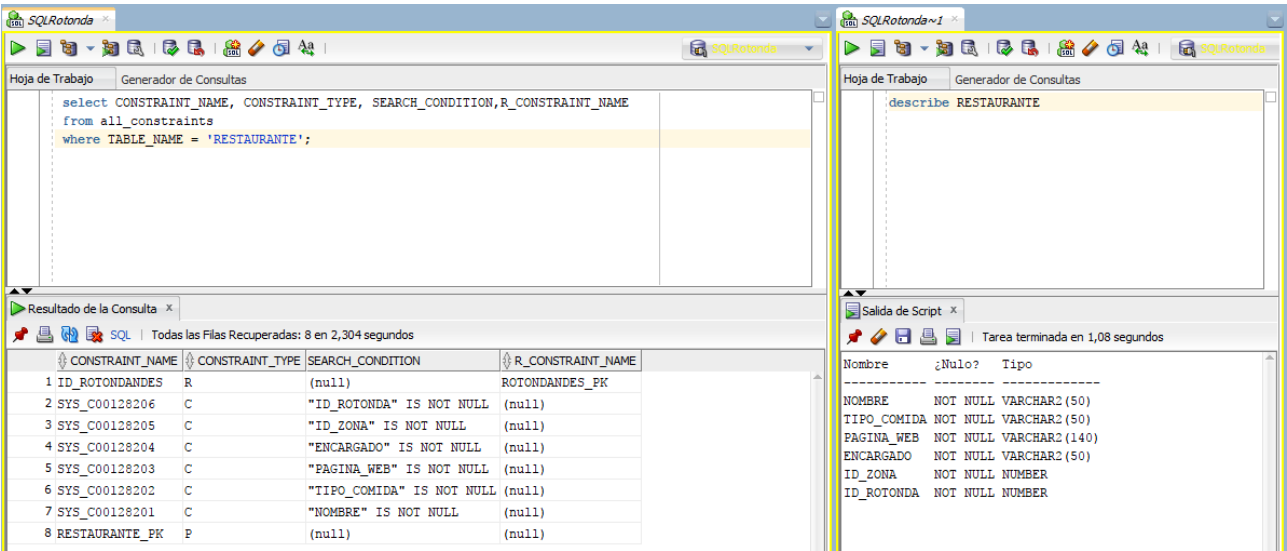


Imagen 16: Resultados de la consulta para Restaurante.

# RotondAndes.

Nombre	Rotondandes		
Descripción	Guarda las rotondas		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
id	Number	id rot.	NN, PK
nombre	Varchar 20	nombre	NN

Tabla 21: Descripción de la tabla RotondAndes.

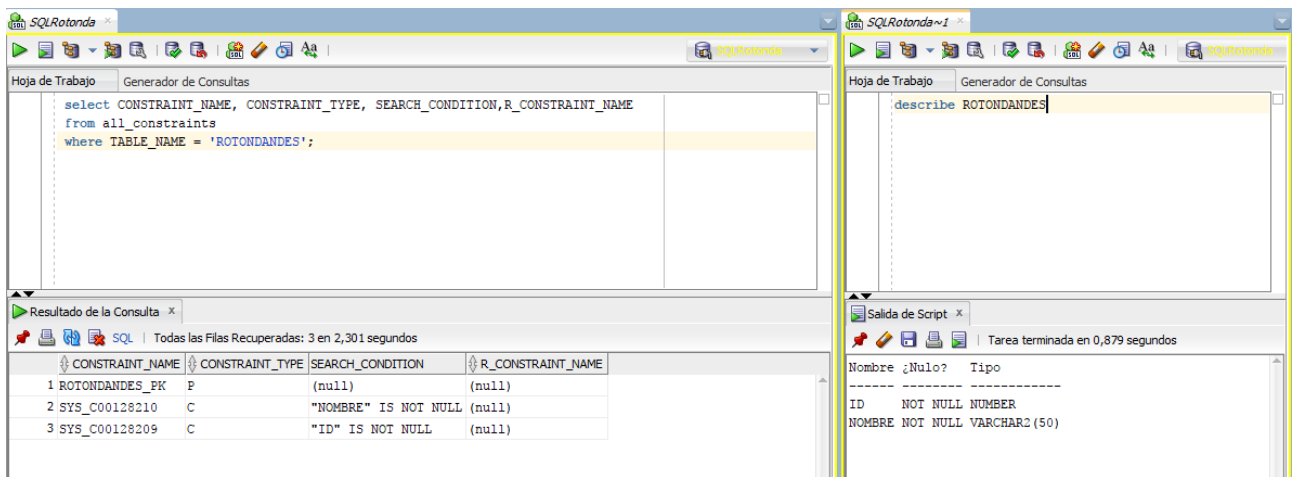


Imagen 17: Resultados de la consulta para RotondAndes.

## Tipo

Nombre	Tipo		
Descripción	Guarda los tipos.		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
id	Number	id tipo	NN, PK
nombre	Varchar 20	Desc.	NN

Tabla 22: Descripción de la tabla Tipo.

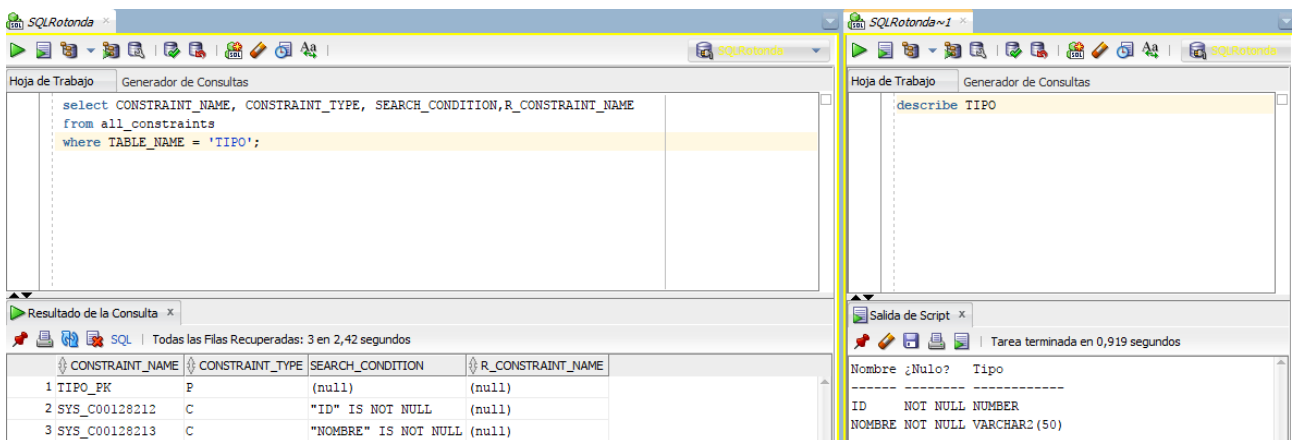


Imagen 18: Resultados de la consulta para Tipo.

## Tipo Producto

Nombre	Tipo Producto		
Descripción	Guarda los tipos para productos.		
Llave	idTipo, idProducto		
Atributo	Tipo	Descripción	Características/Restricciones
idTipo	Number	id tipo.	NN, PK, Fktipo.id
idProducto	Number	id Prod.	NN, PK, Fkproducto.id

Tabla 23: Descripción de la tabla Tipo Producto.



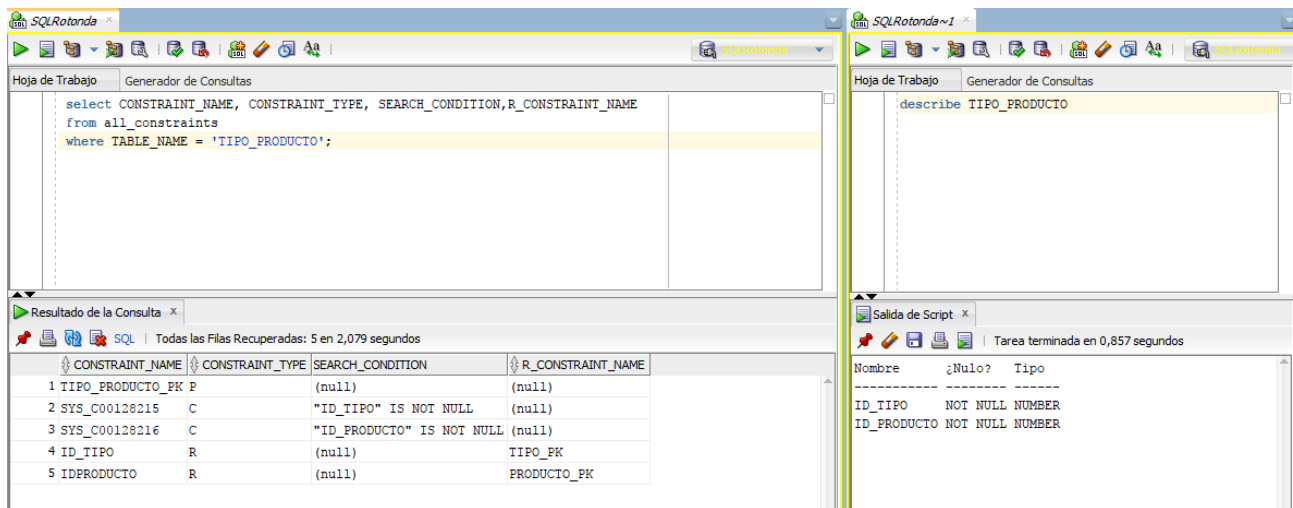


Imagen 19: Resultados de la consulta para Tipo Producto.

## Zona

Nombre	Zona		
Descripción	Guarda las zonas de las rotondas		
Llave	idZona		
Atributo	Tipo	Descripción	Características/Restricciones
idZona	Number	id zona.	NN, PK
Nombre	Number	nombre	NN
EsAbierta	Varchar 2	libre.	NN
Capacidad	Number	Cap.	NN
Apta	Varchar 2	publico.	NN
Condiciones	Varchar 20	Condicio	NN
IdRotonda	Number	rotonda	NN, Fkrotonda.id

Tabla 24: Descripción de la tabla Zona.

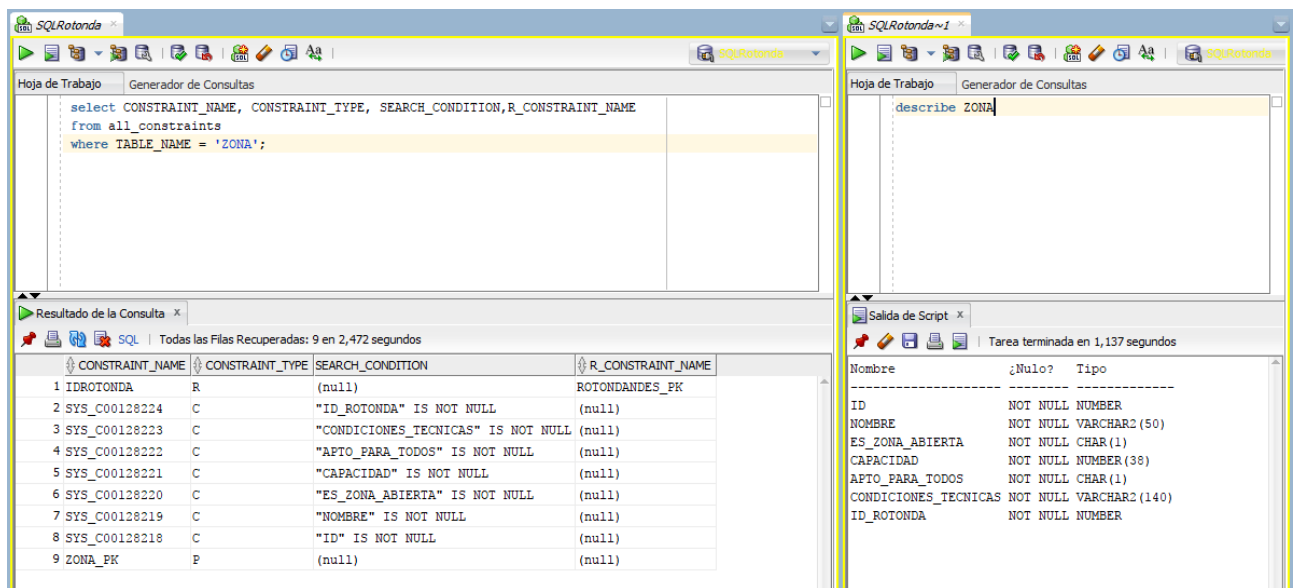


Imagen 20: Resultados de la consulta para Zona.

c. Documente la lógica de los nuevos requerimientos a desarrollar, descritos en la sección de caso de estudio de este documento. En este punto se requiere definir los mecanismos que utiliza para garantizar las propiedades ACID del requerimiento.

Para cada requerimiento será detallada cada una de las propiedades ACID, mediante una tabla y especificación de la propiedad.

En cada transacción, la durabilidad está garantizada con la impresión sobre un LOG de transacciones. Si el programa falla, las transacciones estarán allí para registrarse. Estas transacciones, son escritas a nivel de DAO. Esta bitácora, se encuentra en la carpeta origen de este documento.

#### RF 11-12 Registro de equivalencias

Atomicidad	Todo o nada, se cumple.	
Consistencia	Es garantizada si agrega o no las equivalencias.	
	Nivel	Comentario.
Aislamiento	Read committed.	Como siempre se realiza commit después de obtener un estado valido en la DB. Este es el nivel de aislamiento para estas transacciones.

Tabla 25: propiedades ACID en RF 11-12.

#### RF 13. Surtir restaurante

Atomicidad	Todo o nada, se cumple.	
Consistencia	Es garantizada, siempre	
	Nivel	Comentario.
Aislamiento	Read committed.	Si dos adminRestaurante realizaran esta acción casi simultánea, quedaría el registro del segundo que la realizo. Se vería en el log.

Tabla 26: propiedades ACID en RF 13.

#### RF14 - Registrar pedido de un producto – con equivalencias

Atomicidad	Garantiza la transaccionalidad	
Consistencia	Es garantizada, realizando commit o rollback.	
	Nivel	Comentario.
Aislamiento	Read committed.	El registro de la orden se realizará siempre y cuando exista la disponibilidad de los productos, leer lo que se encuentra en la DB garantiza el manejo correcto de la transacción.

Tabla 27: propiedades ACID en RF 14.

## RF15 - Registrar pedido de productos de una mesa

Para este requerimiento, se utilizó el requerimiento 9 de la iteración anterior.

Atomicidad	Garantiza la transaccionalidad, solo si logra pedir para toda la mesa se hace la transacción	
Consistencia	Es garantizada, realizando commit o rollback.	
	Nivel	Comentario.
Aislamiento	Read committed.	Para registrar la orden de la mesa se verifica si todos los menús se pueden pedir, ordenes individuales. Si esto no se cumple, no se realiza el pedido.

Tabla 28: propiedades ACID en RF 15.

## RF16 - Registrar servicio de una mesa

Para este requerimiento, se utilizó el requerimiento 10 de la iteración anterior.

Atomicidad	Garantiza la transaccionalidad, las ordenes individuales de los comensales son actualizadas como servidas.	
Consistencia	Es garantizada, realizando commit o rollback.	
	Nivel	Comentario.
Aislamiento	Read committed.	Al utilizar Read committed en esta transacción, se garantiza que la lectura de las ordenes es correcta. Actualizándolas como servidas.

Tabla 29: propiedades ACID en RF 16.

## RF17 – Cancelar pedido

Atomicidad	Garantiza la transaccionalidad, la orden se cancela o no.	
Consistencia	Es garantizada, realizando commit o rollback.	
	Nivel	Comentario.
Aislamiento	Read committed.	Utilizando este tipo de lectura, se logra verificar si la orden ya está lista, en caso contrario, es cancelada.

Tabla 29: propiedades ACID en RF 16.

Requerimientos de consulta.

Estos, cuentan con nivel de aislamiento Read committed. Leen lo que se encuentra en la base de datos confirmado al momento, de esta manera, se evitan lecturas falsas o fantasmas a la hora de ponderar estos datos.

### 3. (80 %) Construcción de la aplicación

a. Ajuste las tablas creadas en Oracle de acuerdo a las decisiones del punto anterior.

Anteriormente fueron descritos los cambios en las tablas, en el primer punto de análisis.

c. Desarrolle o ajuste las clases correspondientes a los nuevos requerimientos, de forma que complete o modifique los requerimientos funcionales y cumpla con las restricciones de negocio. En particular, ajuste lo necesario para satisfacer las condiciones **ACID** sobre las operaciones de negocio en los nuevos requerimientos **y en los implementados en la iteración 2.**

- i. Desarrollo y/o ajustes a los servicios REST para cumplir con los nuevos requerimientos.
- ii. Cambios y desarrollo de las transacciones en RotondAndesMaster
- iii. Cambios y desarrollo en los DAO.

Estos cambios se pueden apreciar en las diferentes clases del proyecto al cual pertenece esta carpeta.

- Todos los requerimientos tienen el mismo peso en la evaluación.
- Los bonos serán acreditados con un máximo de 2% sobre el valor total de la nota.
- **Debe** enviar el archivo de Postman.

d. Verifique el comportamiento transaccional de los requerimientos que implican actualización, inserción o borrado de información, mediante la implementación de escenarios de prueba. Genere y documente dichos escenarios, mediante archivos Postman, que le deben asegurar el correcto funcionamiento de la aplicación y la corrección y calidad de los datos en la base de datos.

- En un archivo Excel documente claramente, **para cada requerimiento**, cuáles son los datos que le permiten realizar las pruebas, tanto para los casos de terminación exitosa como los fallidos. Indique cuáles son las respuestas esperadas que corresponden a los datos de prueba.

Estas pruebas, colecciones de pruebas, se encuentran en la subcarpeta PruebasPostman, contenida en la carpeta docs. El archivo de Excel, se encuentra en la carpeta docs., bajo el nombre de "Ejemplos fallos, éxitos pruebas de requerimientos".

### Bonos

¿Qué diferencias hay en el manejo transaccional por parte de un contenedor de aplicaciones con respecto al manejo dado por parte del programador de la aplicación?

Explique claramente las ventajas y desventajas de cada uno de ellos y cómo se refleja en su arquitectura de software y diseño detallado de su **RotondAndes**.

Como ejemplo de contenedor de aplicaciones y, por ende, manejador de transacciones, se tomará EJB. Para él, se describirá el manejo transaccional dado y la función general que cumple como interface empresarial. Tras ser introducido, será contrastado con el manejo transaccional dado en esta iteración.

Cualquier programador puede desarrollar un componente de una aplicación, parte del sistema, tomemos el ejemplo de cliente Banco; otro programador, podrá ensamblarlo a su aplicación. La funcionalidad de EJB es contener componentes, "algo así como el sistema operativo en el que estos residen" ("Sesión 1: Introducción a la tecnología EJB," s.f.).

EJB provee varios servicios, de acuerdo con el blog de introducción al tema de la Universidad de Alicante, algunos son:

- Manejo de transacciones: apertura y cierre de transacciones asociadas a las llamadas a los métodos del bean.
- Concurrencia: llamada simultánea a un mismo bean desde múltiples clientes.
- El contenedor EJB debe gestionar el acceso concurrente a los mismos y por ello debe usar transacciones.
- Persistencia: sincronización entre los datos del bean y tablas de una base de datos.
- Gestión de mensajes: manejo de Java Message Service (JMS).
- Escalabilidad: posibilidad de constituir clusters de servidores de aplicaciones con múltiples hosts para poder dar respuesta a aumentos repentinos de carga de la aplicación con sólo añadir hosts adicionales.
- Adaptación en tiempo de despliegue: posibilidad de modificación de todas estas características en el momento del despliegue del bean. ("Sesión 1: Introducción a la tecnología EJB," s.f.).

Otro tema tratado por EJB es el de la privacidad y manejo de las transacciones, verificar que el cliente pueda hacer esa transacción, iniciarla y terminarla. También sabe si se debe actualizar la información actual, la presente en memoria principal, con la que se acaba de plasmar en la DB.

Ahora, serán señaladas algunas ventajas del manejo de transacciones mediante un contenedor de aplicaciones, EJB para este caso de estudio.

	EJB	Manejo Actual
Complejidad	"Escribir un bean, es como escribir una clase"	Se debe manejar por niveles DAO, TM...
Portabilidad	J2EE es el soporte necesario para ejecutarlas.	Es necesario un wildFly.
Integración	Existen más posibilidades de integración de apps.	RotondAndes debería adaptarse a la aplicación de integración. Mayor dificultad.

Despliegue distribuido	Posibilidad de varios despliegues desde la misma IP.	Despliegue de manera local e individual.
Manejo transaccional	Se adapta la interfaz y controla las transacciones y concurrencia del programa.	Manuales, creados e interpretados por el programador en función del problema.
Control concurrencia		

Tabla 30: Contraste en diferentes aspectos entre EJB y el manejo actual. EJB tiene dominio en estos aspectos.

Por otro lado, el manejo actual de las transacciones y servicios tiene también sus ventajas:

Posibilidad de manejo para transacciones de mayor complejidad, también, el control de las mismas.

Mayor control del proceso y aprendizaje, al utilizar esta forma de manejo para las transacciones, se comprende la manera en la que son ejecutadas por el servidor e interpretadas por la DB.

Diversidad en los resultados de las consultas, diferentes tipos de objetos, como Producto Venta, pueden ser retornados tras las consultas, al aumentar la complejidad de las mismas o buscar datos muy específicos entre tablas.

Referencias.

Sesión 1: Introducción a la tecnología EJB. (s.f.). Recuperado el 28 de octubre de 2017 de <http://www.jtech.ua.es/j2ee/2003-2004/abierto-j2ee-2003-2004/ejb/sesion01-apuntes.htm>