

1. Diseño de la aplicación:

```
SELECT S.NombreTabla, S.NombreConstraint, X.NombreCol,w.NombreColPK,w.tipoDatoColPK
FROM (
(SELECT TABLE_NAME AS NombreTabla, CONSTRAINT_NAME AS NombreConstraint FROM
SYS.ALL_CONSTRAINTS WHERE OWNER = 'ISIS2304A231720') s
INNER JOIN
(SELECT TABLE_NAME as NombreTabla, COLUMN_NAME AS NombreCol , POSITION as
OrdenColPK, CONSTRAINT_NAME AS NombrePK
FROM SYS.ALL_CONS_COLUMNS WHERE OWNER = 'ISIS2304A231720') X
ON S.NombreTabla=X.NombreTabla AND S.NombreCONSTRAINT=X.NombrePK
INNER JOIN
(SELECT DISTINCT CONSTRAINT_NAME as NombrePK
FROM SYS.ALL_CONS_COLUMNS WHERE OWNER = 'ISIS2304A231720')Z
ON S.NombreCONSTRAINT=Z.NombrePK AND X.NombrePK=Z.NombrePK
INNER JOIN
(SELECT COLUMN_NAME AS NombreColPK,DATA_TYPE as tipoDatoColPK
FROM SYS.ALL_TAB_COLUMNS WHERE OWNER = 'ISIS2304A231720')W
ON X.NombreCol=W.NombreColPK )
ORDER BY s.NombreTabla desc;
```

	NOMBRETABLA	NOMBRECONSTRAINT	NOMBRECOL	NOMBRECOLPK	TIPODATOCOLPK
123	ITEMS_ORDEN	SYS_C00131149	ORDEN_ID	ORDEN_ID	NUMBER
124	ITEMS_ORDEN	SYS_C00131150	ITEMS_ID	ITEMS_ID	NUMBER
125	ITEMS_MENU	SYS_C00131251	MENU_ID	MENU_ID	NUMBER
126	ITEMS_MENU	IM_MENUS	MENU_ID	MENU_ID	NUMBER
127	ITEMS_MENU	SYS_C00131251	MENU_ID	MENU_ID	NUMBER
128	ITEMS_MENU	IM_MENUS	MENU_ID	MENU_ID	NUMBER
129	ITEMS_MENU	SYS_C00131251	MENU_ID	MENU_ID	NUMBER
130	ITEMS_MENU	IM_MENUS	MENU_ID	MENU_ID	NUMBER
131	ITEMS_MENU	SYS_C00131250	ITEMS_ID	ITEMS_ID	NUMBER
132	ITEMS_MENU	IM_ITEMS	ITEMS_ID	ITEMS_ID	NUMBER
133	ITEMS_MENU	SYS_C00131250	ITEMS_ID	ITEMS_ID	NUMBER
134	ITEMS_MENU	IM_ITEMS	ITEMS_ID	ITEMS_ID	NUMBER
135	ITEMS_MENU	SYS_C00131250	ITEMS_ID	ITEMS_ID	NUMBER
136	ITEMS_MENU	IM_ITEMS	ITEMS_ID	ITEMS_ID	NUMBER
137	ITEMS	SYS_C00131157	ID_RESTA...	ID_RESTAU...	NUMBER
138	ITEMS	SYS_C00151720	COSTO_PRODU	COSTO_PRODU	NUMBER
139	ITEMS	ITEMS_PK	ID	ID	NUMBER
140	ITEMS	SYS_C00131158	NOMBRE	NOMBRE	VARCHAR2
141	ITEMS	SYS_C00131159	TIPO	TIPO	VARCHAR2
142	ITEMS	SYS_C00131160	PRECIO	PRECIO	NUMBER
143	ITEMS	SYS_C00151718	NOMBREIN...	NOMBREINGLES	VARCHAR2
144	ITEMS	SYS_C00151719	TIEMPO_P...	TIEMPO_PR...	NUMBER
145	ITEMS	ITEMS_PK	ID	ID	NUMBER
146	ITEMS	SYS_C00131156	ID	ID	NUMBER
147	ITEMS	ITEMS_PK	ID	ID	NUMBER
148	ITEMS	SYS_C00131158	NOMBRE	NOMBRE	VARCHAR2
149	ITEMS	SYS_C00131160	PRECIO	PRECIO	NUMBER

Cambios relevantes:

- Se crearon nuevas tablas para poder cumplir con los nuevos requerimientos funcionales, en este caso EQUIV\_INGREDIENTE y EQUIV\_PRODUCTO, al igual que un atributo nuevo que representara la mesa en la que se encuentra el cliente.
- 2. Requerimientos funcionales:
  - 11: Para registrar los ingredientes equivalentes se tiene en cuenta que restaurante es el que esta realizando esa decisión ya que para otros restaurantes puede que esa equivalencia no se tenga en cuenta, así manejamos la consistencia de estos datos, las otras propiedades ACID se cumplen por medio de la lógica y los niveles de aislamiento de Oracle.
  - 12: al igual que en el requerimiento 11 se tiene en cuenta el restaurante que realiza la tarea, ya que se tienen que verificar otros factores para la consistencia de datos, por ejemplo a equivalencia de los dos ítems debe ser del mismo restaurante, no se deben poder hacer equivalencias con productos que no ofrecen.

13: en este requerimiento era sencillo crear problemas de consistencia, toca tener en cuenta que restaurante realiza la operación a que producto, cuanta cantidad desea agregar y la cantidad de los productos actual que puede variar dependiendo de las ventas del día anterior.

14:

En este requerimiento se usan las tablas de equivalencia de productos cuando el usuario pide un cambio de equivalencia la sentencia busca el producto y su equivalencia valida que al equivalencia sea correcta y realiza un update a la orden con el id de la equivalencia.

15:

Para esto se usa la tabla mesa y mesa \_orden donde al registrar un pedido a la mesa el usuario que registro asocia un pedido nuevo a una mesa para luego añadirle ítems.

16:

En este el restaurante comprueba si todo los pedidos ya se entregaron para esto el restaurante puede verificar si todo el pedido ya esta completo y se descuenta de su inventario , se añadió un estado a ítem orden que dice si fue entregado o no.

17: al cancelar un pedido se debe eliminar y actualizar información de muchas tablas. Es por eso que al momento de cancelar la orden se mira todos ls ítems que se ordenaron, en la tabla ITEMS se tiene que aumentar la cantidad de cada ítem correspondiente a la orden, luego se debe eliminar de la tabla ítems\_orden y orden la información respecto a la orden específica a cancelar, cualquier usuario restaurante puede cancelar una orden sin importar los ítems en esta.

### 3: Requerimientos de consulta:

7: este requerimiento no es muy complicado de realizar debido al diseño de la base de datos, entonces así el cliente solicitara solo menus siempre descomponemos la oden en ítems individuales para facilitar su consulta. Solo se realiza un get con el id del usuario (ya que tiene que estar registrado ) el cual nos devuelve una lista de todos los ítems solicitados por ese usuario.

8: en este requerimiento se pedía que se consultara la totalidad de pedidos de la rotonda, para el admin cuando realiza la consulta le mostrara la lista de restaurantes discriminados por ítem y ventas totales del ítem, si consulta como restaurante solo saldrán esta lista con los ítems del restaurantes