

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

FUNDAMENTOS DE BASES DE DATOS

Proyecto Final

David Hernández Uriostegui, 420003708

Santiago Raúl Díaz Pontón, 317164402

Diego Javier Padilla Lara, 420003007

June 16, 2022

0.1 Modelo Entidad-Relación

Entidades: Las entidades que manejamos en el modelo son:

1. **Taquería-** Es la entidad principal del modelo. Su llave es *IdTaqueria*, ya que cada taquería tiene un Id único. Sus atributos son:
 - **Email-** Email de la taquería
 - **Teléfono-** Número de teléfono de la taquería
 - **Dirección-** Dirección de la taquería, incluyendo el número, calle, Código Postal.
 - **Nombre-** El nombre de la taquería
2. **Empleado-** Es la entidad que define a los empleados de la empresa. Su llave principal es *RFC*. Sus atributos son:
 - **Teléfono-** Número de teléfono de la taquería
 - **Edad-** Edad del empleado
 - **Número de seguridad social-N.S.S:** El número de seguridad social del empleado
 - **Antigüedad-** Antigüedad que tiene el empleado en la empresa.
 - **Nombre-** Nombre del empleado
 - **Apellido paterno-** Apellido del empleado
 - **Apellido materno-** Apellido del empleado
 - **Dirección-** Dirección de la taquería, incluyendo el número, calle, Código Postal.
 - **IdTaqueria-** Taquería donde trabaja el empleado.

Los empleados pueden ser sólo un tipo de empleado:

- **Cajero-** Cajero de la taquería
 - **Parrillero-** Parrillero de la taquería
 - **Taquero-** Taquero de la taquería
 - **Mesero-** Mesero de la taquería
 - **Repartidor-** Los repartidores tienen atributos específicos.
 - **TipoVehículo-** Especifica si el conductor maneja una bicicleta o una motocicleta
 - **Modelo-** Modelo del vehículo
 - **Marca-** Marca del vehículo
 - **Placa-** Placa del vehículo
3. **Cliente-** Define a los clientes que compran en la taquería. Su llave principal es el *CURP* junto con el *IdTaqueria* de la taquería donde compraron. Sus atributos son:
 - **Nombre-** Nombre del cliente
 - **Apellido paterno-** Apellido paterno del cliente
 - **Apellido materno-** Apellido materno del cliente
 - **Dirección-** Dirección del cliente, incluyendo el número, calle, Código Postal.
 - **CURP-** El CURP del cliente.
 - **Email-** El email del cliente

- **PuntosTC**- Los puntos de la taquería que tiene el cliente.
4. **Ticket**- Se le entrega a el cliente cada que realiza una compra. Su llave principal es el *id* de la compra, junto con el *idCliente* que es único para cada cliente. Sus atributos son:
 - **Fecha**- La fecha en la que se realizó la compra
 - **RFCMesero**- El mesero que atendió
 - **TotalPago**- La cuenta total a pagar. Las compras se pueden realizar mediante los siguientes métodos de pago:
 - **Efectivo**- El pago es con efectivo
 - **Tarjeta**- El pago es con tarjeta
 - **Puntos**- El pago es con puntos del cliente
 5. **Inventario**- El inventario que contiene los items de la tienda. Su llave principal es el *Id* del inventario, junto con el *IdTaqueria* de la taquería cuyo inventario es. Sus atributos son:
 - **IdInventario**- El Id del inventario.
 - **CantidadRestante**- La cantidad que queda en el inventario
 - **Tipo**- El tipo de item que se encuentra en el inventario.
 - **IdTaqueria**- El Id de la taqueria del inventario.
 6. **Proveedor**- Es la persona que provee el inventario de la taquería. Su llave principal es el *RFC* del proveedor. sus atributos son:
 - **IdTaqueria**- El id de la taqueria
 - **Nombre**- El nombre del proveedor.
 - **Teléfono**- El teléfono del proveedor
 - **Email**- El email del proveedor
 - **DescripcionServicio**- La descripción de los items que el proveedor provee.
 7. **Compra**- El servicio que realiza el proveedor. Su llave principal es *idCompra*. Sus atributos son:
 - **idCompra** - El id de la compra
 - **IdTaquería**- El id de la taquería donde se realizó la compra
 - **PrecioCompra**- El precio que el proveedor cobró por la compra
 - **FechaCompra**- La fecha de la compra
 8. **Menú** Es el menú que tiene la taquería, en este se reflejan todos los platillos (*items*) que se encuentran disponibles. Su llave principal es *IdTaqueria*. Su único atributo es *item*.
 9. **item**- Son los platillos disponibles en el menú. Su llave principal es *IdItem*. Sus atributos son:
 - **Nombre**- Nombre del platillo
 - **IdItem**- El id del platillo
 - **Precio**- El precio del platillo
 - **Descripcion**- La descripcion del platillo
 10. **Promociones**- Son las promociones que hay en la taquería. Su llave principal es *idPromocion*. Sus atributos son:

- **IdPromocion**- El id de la promoción.
 - **Descripción**- Descripción de la promoción
 - **Precio**-El precio de la promoción
11. **Categoria**- Es la categoría a la que pertenece un platillo. Su llave principal es *IdCategoria*. Sus atributos son:
- **IdCategoria**-El Id de la categoría.
 - **NombreCategoria**- El nombre de la categoría.
12. **Ingrediente**- Son los ingredientes que contiene un platillo (*item*). Su llave principal es *IdIngrediente*. Sus atributos son:
- **IdIngrediente**- El Id del ingrediente
 - **Tipo**- El tipo del ingrediente
 - **Nombre**- El nombre del ingrediente.
 - **Cantidad**- Todo ingrediente se usa en una cantidad.
13. **RegistroHistórico**- Es el registro histórico que tienen todos los items del menú. Su llave principal es *IdItem*. Sus atributos son:
- **Fecha** - La fecha del registro del item
 - **Precio**- El precio del item en la fecha dada
 - **IdItem**- El Id del item del registro
 - **Insumos**- El precio de insumos.
14. **Salsa**- Las salsas son un ingrediente tan importante en la taquería que es una entidad aparte, fuera de los ingredientes. Su llave principal es *IdSalsa*. Sus atributos son:
- **Ingredientes**-Los ingredientes de la salsa
 - **IdSalsa**- El Id de la salsa
 - **Categoria**- La categoría de la salsa.
 - **Recomendación**- Con qué se recomienda la salsa.
 - **Picor** El picor es una entidad de las salsas, y tiene uno de los siguientes atributos:
 - **Dulce**- Es una salsa dulce
 - **Baja**- Tiene picor bajo
 - **Medio**- Tiene picor medio
 - **Alto**- Tiene picor alto
 - **Extremo**- Tiene picor extremo

Relaciones:

- **Trabajar**- Varios a uno entre Empleado y taquería.
- **Poseer**- Relación entre Cliente a tickets. Un cliente puede poseer muchos tickets de la taquería.
- **Comprar**- Uno a muchos. Un Cliente puede Comprar muchos items del menú.
- **Ordena**- Uno a muchos entre menú y items. Un menú ordena muchos items.

- *Provee*- Muchos a muchos entre proveedor con items. Muchos proveedores pueden proveer muchos productos.
- *Tiene*- Uno a uno. Un item tiene un registro histórico .
- *Utilizar*- Uno a muchos entre item e ingredientes. Un item puede tener muchos ingredientes.
- *Utiliza*- Uno a uno entre ingrediente y item. Un item tiene una cantidad específica de un ingrediente
- *Contiene*- Uno a muchos entre inventario e item. Un inventario tiene muchos items.
- *Tiene*- Uno a muchos entre Salsa e ingredientes. Una salsa puede tener muchos indredientes.
- *Tener*- Uno a uno entre Taquería y menú. Cada taquería tiene un menú.

Restricciones-

1. Los Empleados sólo pueden tener un tipo de trabajo; Cajero, Mesero, Parrillero, Tortillero, Taquero ó Repartidor.
2. El pago del cliente sólo puede ser: Con tarjeta, con efectivo ó con puntos de cliente
3. Cada Promoción es única por taquería.
4. Las salsas pueden tener único tipo de picor.
5. Cada taquería tiene un inventario
6. El atributo teléfono de la entidad empleado es multivaluado. Los empleados pueden tener más de un número de teléfono.
7. Cada taquería tiene un menú para ordenar los ítems de la tienda.

0.2 Modelo Relacional

Hagamos la traducción del modelo *entidad-relación* al modelo relacional.

- **Taquería-**(*RFC, Nombre, calle, número, Estado, Cpostal, Teléfono, Email*). Tiene como **PK** a *id*, representando el id de la taquería
 - **Parrillero-**(*Nombre, Apaterno, AMaterno, Calle, Número, Estado, Cpostal, Edad, NSS, Antigüedad, teléfono, Email*). Tiene como **PK** el *RFC* del empleado, y como **FK** el id de la taquería.
 - **Mesero-**(*Nombre, Apaterno, AMaterno, Calle, Número, Estado, Cpostal, Edad, NSS, Antigüedad, teléfono, Email*). Tiene como **PK** el *RFC* del empleado, y como **FK** el *id de la taquería*.
 - **Cajero-**(*Nombre, Apaterno, AMaterno, Calle, Número, Estado, Cpostal, Edad, NSS, Antigüedad, teléfono, Email*). Tiene como **PK** el *RFC* del empleado, y como **FK** el *id de la taquería*.
 - **Taquero-**(*Nombre, Apaterno, AMaterno, Calle, Número, Estado, Cpostal, Edad, NSS, Antigüedad, teléfono, Email*). Tiene como **PK** el *RFC* del empleado, y como **FK** el *id de la taquería*.
 - **Tortillero-**(*Nombre, Apaterno, AMaterno, Calle, Número, Estado, Cpostal, Edad, NSS, Antigüedad, teléfono, Email*). Tiene como **PK** el *RFC* del empleado, y como **FK** el *id de la taquería*.
 - **Repartidor-**(*Nombre, Apaterno, AMaterno, Calle, Número, Estado, Cpostal, Edad, NSS, Antigüedad, teléfono, Email, TieneTransportePropio, Marca, Modelo, Tipo*). Tiene como **PK** el *RFC* del empleado, y como **FK** el *id de la taquería*.
 - **Compra-**(*Marca, Tipo, FechaCompra, FechaCaducidad, CantidadComprada, PrecioCompra*). Tiene como **PK** el *id de la compra*, y como **FK** el *RFC* del proveedor que proveeó la compra.
 - **Inventario-**(*Tipo, FechaCaducidad, CantidadRestante*). Tiene como **PK** el *Tipo* de producto, y como **FK** el *ID* de la taquería
 - **Cliente-**(*Nombre, Apaterno, AMaterno, Calle, Número, Estado, Cpostal, Teléfono, Email, PermiteAdomicilio y PuntosTC*). Tiene como **PK** el *CURP* del cliente y como **FK** el *ID* de la taquería donde se realizó la compra.
 - **Ticket-**(*idTaqueria, Fecha, RFCMesero, EsEfectivo, EsTarjeta, EsConPuntos, Total*). Tiene como **PK** el *ID del ticket*, y como **FK** el *ID del cliente*.
 - **Item-**(*Nombre, Descripción, Precio*). Tiene como **PK** el *id del item*, y como **FK** el *ID de la categoría* a la cuál pertenece el producto.
 - **Registrohistórico-**(*Precio, PInsumos, Fecha*). Tiene como **PK** el *ID del registro*, y como **FK** el *ID del item* registrado
 - **Categoría-**(*Nombre*). Tiene como **PK** el *ID de la categoría*.
 - **Ingrediente-**(*Nombre, Tipo, Precio*). Tiene como **PK** el *ID del ingrediente*.
 - **Salsa-**(*Nombre, Picor, Recomendación, Precio250, Precio500, Precio1000*). Tiene como **PK** el *ID de la salsa*, y como **FK** el *ID de la categoría*.
- Las siguientes entidades y relaciones se crearon poder modelar de una mejor manera la base de datos, ya que así será más sencillo realizar consultas:
- **Proveedor-**(*Nombre, servicio*). Tiene como **PK** el *RFC del proveedor*

- **TelefonoProveedor**- Tiene como **PK** el *RFC del vendedor*, y como **PK** el *teléfono del proveedor*
- **EmailProveedor**. Tiene como **PK** el *RFC del vendedor*, y como **PK** el *email del proveedor*. A partir de *Abastecer*, podremos consultar los items más solicitados por los taqueros.
- **Abastecer**- Tiene como **PK** el *ID de la taquería* a abastecer, y como **FK** el *ID de la compra*. A partir de *Vender*, podremos consultar exactamente qué fue lo que consumió el usuario.
- **Vender**- (*Cantidad*). Tiene como **PK** el *ID del item vendido*, y como **FK** el *ID del ticket*. A partir de *Utilizar*, podremos consultar con mayor facilidad los ingredientes que más ocupan los taqueros.
- **Utilizar**- (*Cantidad*). Tiene como **FK** el *ID del item* donde se utilizará, y también el *ID del ingrediente* a usar como **FK**
- **OrganizarMenu**- Tiene como **FK** el *ID de la taquería* cuyo menú se organizará, y también como **FK** el *ID de la categoría*
- **UtilizarEnSalsa**(*Cantidad*). Tiene como **FK** el *ID de la salsa* a usar, y como **FK** el *ID del ingrediente*

0.3 Decisiones tomadas en cuanto al diseño de la Base de Datos

Durante el diseño y desarrollo del proyecto hubo ciertos cambios que se hicieron para que el modelo fuera más robusto y simple. Es decir usamos la técnica que hemos venido usando durante toda la carrera: **"Divide y vencerás"**.

Respecto al diseño en general de la base de datos, optamos por mantenerla lo más concisa posible, evitando información repetitiva y/o basura.

Para ésto, primero necesitábamos reconocer las entidades que estarían tomando un papel relevante, poniendo especial atención al caso de uso que se nos proporciona.

Cabe mencionar que al no tener toda la información específica, tuvimos que plantear soluciones para éstos casos. Lo cuál puede notarse en la entidad Compra e Inventario, pues si bien se nos habla de llevar un control de inventario, decidimos que era mejor separar las compras de los productos en almacén.

Un cambio importante a mencionar en el diseño es que originalmente teníamos pensado incluir a todos los empleados en la misma tabla; sin embargo, tras considerarlo mejor optamos por tener tablas separadas para cada tipo de empleado.

Luego, para guardar las "recetas" de los platillos se decidió tener esa información en una relación Utilizar dónde pudiésemos tener además de los ingredientes y donde se están utilizando las cantidades de los mismos, pensando en la practicidad.

Aún así, dejando de lado todo lo anterior, en general no nos desviamos demasiado de nuestro diseño original.

0.3.1 Relación de archivos

Todos los archivos contenidos en el directorio SQL/ son necesarios para la correcta inicialización y poblado de la base de datos. Con ésto en mente, a continuación se explica brevemente la relación que mantienen:

- Creación de la base de datos:

- **DDL.sql**: Este script se encarga de crear todas las tablas de nuestra base de datos, así como de comentarlas. Debe correrse al **principio**.
 - **Triggers.sql**: Aquí encontramos los triggers necesarios para el funcionamiento de la bd, es importante ejecutar éste script antes de poblar pues tenemos un trigger que se encarga de insertar usuarios default cada vez que se agrega una taquería al sistema (**usuariodefault**).
El otro trigger nos ayuda a insertar valores relevantes en el registro histórico cada vez que se detecta un cambio en los items **alhistorico**.
 - **SP.sql**: Tenemos un par de procedimientos auxiliares en el manejo de la base de datos. **nuevoitem** que se encarga de insertar un nuevo item del menú (incluyendo receta). Y **itemdelete**, el cuál elimina un item del menú por id junto con sus referencias.
- Poblar base de datos:
El único archivo necesario para poblar la base de datos es **Poblamiento.sql**
 - Operar base de datos:
Las queries solicitadas pueden encontrarse en **Queries.sql**

0.4 Población de la Base de Datos

Para la población de la Base lo que se buscó fue usar datos lo más concisos y similares posibles a los que se puede esperar encontrar en taquerías, así como datos que se deberían encontrar en México.

Nuestra herramienta principal para poder poblar las tablas

- taqueria
- parrillero
- taquero
- mesero
- cajero
- tortillero
- repartidor
- cliente
- ticket

Fue Mockaroo, esto ya que provee herramientas muy útiles para hacerlo, pero también creamos algunos Datasets para poder hacer más concisos los datos.

Creamos los siguientes Datasets

- Estados - Este dataset contiene todos los estados de la Republica Mexicana
- Nombres - Contiene nombres que se puede encontrar en México.
- Apellidos - Contiene apellidos que se puede encontrar en México.

Estos Datasets se usaron más que nada para generar las tablas que requerían atributos como *nombre*, *apellido* y *estado*.

Una vez teniendo ya generado los Datasets para *cliente* y *mesero* los pasamos a .csv para poder usarlos en el poblamiento de otras tablas para evitar inconsistencias y problemas de referenciamiento.

Después de haber creado estas tablas procedimos a crear las tablas relacionadas con los alimentos:

- categoria
- item
- salsa
- organizar_menu
- registro_historico
- ingrediente
- utilizar
- utilizar_en_salsa

Para poder hacer lo más preciso a la realidad estos datos, buscamos en internet distintos tipos de Tacos, Tortas, Gringas, Burritos, Tortas, Platos mexicanos y Bebidas.

Para la población de estas tablas no usamos Mockaroo, realizamos algunos scripts en **Python** en los que usamos las bibliotecas **Pandas** y **Numpy** ya que nos resultó más fácil y conveniente hacerlo de esta manera.

Una vez teniendo estas tablas proseguimos a poblar las tablas que se relacionaban con los productos que se ofrecen en las taquerías:

- ticket
- vender
- compra
- abastecer
- inventario
- proveedor
- telefono_proveedor
- email_proveedor

De hecho, pensamos que para la tabla *ticket* necesitábamos mínimo una cantidad de 100000 tickets para que fuera lo más acertado posible, sin embargo, decidimos dejarlo en una cantidad mucho más baja (5000), debido que al correr el script de Poblamiento con 50000 tickets tardaba bastante y en la mayoría de los integrantes provocaba que **pgAdmin4** dejara de funcionar. De tal manera que por lo antes mencionado en nuestro script de Poblamiento se optó por sólo incluir 5000 líneas de *INSERT INTO* en la tabla *ticket*.

Una vez habiendo poblado la tabla *ticket* procedimos a poblar la tabla *vender*. Nuevamente para poder poblar esta tabla de una manera más eficiente se usó **Python**, esto lo hicimos ya que igualmente guardamos el .csv de *ticket* y de ahí usamos los *id's*, de la misma manera que generamos un .csv de los *id's* de la tabla *item*.

De manera similar se poblaron las tablas restantes, a excepción de *proveedor*, *compra*, *telefono-proveedor* y *email-proveedor*, estas si se poblaron usando Mockaroo.

Cómo se puede observar buscamos que la población fuera lo más concisa posible para que fuera lo más parecida a lo que uno esperaría de una franquicia de taquerías. Por esta razón consideramos que usar más herramientas además de Mockaroo sería lo más óptimo. Esto ya que Python no nos limita sólo a generar 5000 queries por request, lo cual es verdaderamente ventajoso a la hora de generar datos aleatorios para poblar nuestra Base.

Creemos que el número de datos ingresados fue lo suficientemente bueno para poder realizar diferentes consultas complejas e interesantes, es verdad que si hubieramos dejado los 50000 tickets sería aún mucho más enriquecedor, pero no quisimos arriesgarnos a que el script provoque lo que le provocó a la mayoría de integrantes.