

Proyecto Final

Taquerías “Taco Riendo”

Fecha de entrega

15 de junio de 2022



El señor José “el cuñado” Cruz es el dueño de la cadena mexicana de tacos “Taco Riendo”, ha comenzado un plan de expansión dentro de la República Mexicana gracias al éxito obtenido en Ciudad de México. Durante su visita a la Expo Taco del presente año en el WTC, el señor José detectó que muchas de las franquicias de tacos más exitosas utilizan software que les permite crear menús dirigidos o promociones para sectores específicos de la población. Tras esto, el señor Pepe se acercó a la empresa Computólogos A.C. para que ésta le brindara una solución parecida a la de las franquicias, pero ajustándose a sus posibilidades económicas y de mercado. Tras la reunión de levantamiento de requerimientos, el líder del proyecto identificó las siguientes reglas de negocio:

- La carta de sus locales incluye **tacos, burritos, quesadillas, gringas, tortas, platillos mexicanos, salsas y bebidas** (se adjuntan algunos ejemplos de menús de taquerías, para que se den alguna idea), los precios se suelen colocar con etiquetas fluorescentes y en caso de que deban ser actualizados, se pega una nueva etiqueta encima. A don Pepe le interesa poder llevar un **registro histórico** de los precios de todos sus productos y de los insumos de los mismos, ya que le interesa saber en qué momento puede sacar alguna promoción.
- Se desea poder otorgar **tickets por cada consumo** (puede ser por comensal o mesa), el cual debe tener la siguiente información: **fecha, sucursal, mesero que atendió, detalle de los productos solicitados** con el subtotal en cada caso y el **total del consumo**.
- Se necesita identificar la **cantidad aproximada de porciones de ingredientes** necesaria para **cada una de las preparaciones** que se realizan en la taquería, esto con el fin de **minimizar** los desperdicios generados (los datos no tienen que ser reales, ten en cuenta que es solo una simulación, solo trata que tengan algún sentido).
- Se desea crear un programa llamado **Taquero de corazón**, el cual otorgará el **10% del consumo en puntos**, lo cuales podrán ser intercambiados por productos **dentro de la taquería**, cada punto equivale a un peso.
- De igual forma, suele manejar promociones como: “**Lunes de 2x1 en tacos al pastor**”, miércoles de “**Taco Chinito**” en el que **todos los tacos** cuestan \$10.00 o “**Viernes de tortuga**” (2x1 en tortas).
- La taquería ofrece **servicio a domicilio** por lo que se necesitarán los datos de los clientes (nombre completo, CURP, dirección, teléfono, correo electrónico); con este servicio, los clientes podrán pagar solo con **efectivo o tarjeta de crédito/débito**.
- Se desea tener un **control de pago a empleados** a través del pago **vía nómina** y **otorgar seguridad social**; para lograr esto, se requiere la información personal de los empleados (nombre completo, dirección, RFC, edad, número de seguridad social, antigüedad, dirección, teléfono y correo electrónico). Adicionalmente se busca crear un **programa de bonos para empleados**, con el que cada empleado que cumpla **2 años** laborando en la taquería se le agregará a su pago de nómina \$1,500.00. Los clientes también pueden ser alguno de los empleados, en cuyo caso, el **servicio será gratuito**.
- Se debe considerar una solución que pueda ser escalable para poder eventualmente controlar sucursales, ya que don Pepe tiene pensando abrir al **menos una en los próximos meses** y posteriormente, abrir **una en cada alcaldía de la CDMX** y más adelante, **una en cada uno de los estados** de la República Mexicana.

- Derivado del punto anterior, **los empleados sólo podrán estar ligados a una sucursal**; en el caso de los clientes, **no se debe duplicar** su información en la base de datos.
- Se busca tener un **control de inventarios** especificando el **día de compra**, **fecha de caducidad**, **cantidad**, **marca del producto adquirido**, **precio de compra** (ingredientes, mesas, sillas, bancos, platos, servilleteros, etcétera).
- También interesa almacenar **información de los proveedores**, ya que en un futuro interesará hacer alianzas comerciales, a fin de obtener precios más competitivos.
- Un rubro importante es lo relacionado a las **salsas**, ya que las de este lugar son muy famosas y se desea presentarlas en una sección detallando: **ingredientes**, **nivel de picor** (dulce, bajo, medio, alto, extremo) y **recomendación de platillo** a acompañar. De igual forma, derivado del éxito de sus salsas, los clientes pueden obtenerlas para sus eventos, **las presentaciones que se tienen son**: frasco de 250 mg., de medio kilo o bien por litro.
- Es posible que un cliente **no desee dar sus datos** (por cuestiones del aviso de privacidad), en este caso, no se le niega el servicio y se debe mantener una imagen que llamaremos **"cliente default"** al cual se ligará el servicio. Este cliente existe por cada sucursal. Los datos en este caso son: nombre completo (**sucursal**), CURP (dejaremos el **RFC de la taquería**), dirección, teléfono, correo electrónico (**estos son los que corresponden a la taquería**). Este tipo de cliente no puede hacer pedido a domicilio.
- Existen **5 tipos de empleados**: parrilleros, taqueros, meseros, cajeros, tortilleros, repartidores.
- En el caso de los **repartidores**, además de la información definida para los empleados, se requiere del **número de licencia** (si aplica) y si cuenta o no con transporte (**motocicleta o bicicleta**). En el caso de contar con transporte, se requieren los siguientes datos: **marca, modelo, tipo** (motocicleta/bicicleta).
- Por cuestiones de mercadotecnia, se necesita saber las **distintas formas** en que los clientes pagan los servicios: **efectivo, tarjeta de crédito, o puntos** etc.

Para este desarrollo, se desea tener toda la información almacenada en una **base de datos** para que don Pepe pueda llevar un mejor control de sus taquerías. El contrato se adjudicó a **Computólogos A.C.** y se deberá presentar una herramienta funcional para el **15 de junio de 2022**.

Consideraciones

- Antes de iniciar, debes tener en mente que es un proyecto final de **Bases de Datos** y que lo principal a evaluar será eso: tu **base de datos**. La aplicación puede tener un gran diseño o funcionalidad, pero no tiene gran relevancia, no pierdas demasiado tiempo en este punto.
- Leer las reglas de negocio para la taquería y comprenderlas lo mejor posible. Es poco probable que se les proporcionen todas las reglas y será necesario que debas recoger requerimientos adicionales para el correcto funcionamiento de tu proyecto. Este documento no pretende ser exhaustivo y, en caso de incertidumbre, deberás preguntar para afinar.
- Deberás utilizar el **SMBD PostgreSQL** para la implementación de la base de datos.
- Es necesario utilizar **todos los aspectos** revisados en el curso (salvo el tema de transacciones).
- Indicar las **llaves primarias**, tratando que sean naturales.
- Crear el **modelo Entidad – Relación** asociado al caso de uso especificado. Se debe justificar el diseño de manera concisa e indicar de qué forma su esquema empata con las reglas anteriores. Recuerda que un buen diseño de BD es lo suficientemente claro (en sus entidades) con respecto al mundo real. Se recomienda dedicar tiempo suficiente a este apartado.
- Traducir al **modelo relacional**. Este apartado deberá no ser muy complicado si dedicaste el tiempo apropiado al punto anterior.
- Indicar cuáles son las **dependencias funcionales**. Evita aquellas que son triviales y/o redundantes. Tu esquema debe hacer explícitas las **llaves foráneas**. Evita tener más trabajo en el punto siguiente.

- (Opcional) Normalizar tu **diseño de base de datos** del punto anterior, utilizando el conjunto de dependencias funcionales que identificaste. Para fines de este proyecto basta con llegar a **3NF**. Para calificar este apartado no basta con indicar únicamente las relaciones resultantes, se debe indicar el **procedimiento y justificación**. Plasmar el modelo normalizado en un **nuevo esquema relacional**.
- **Construir la base de datos** del esquema definido (ya sea que hayas normalizado o no). Deberás asegurar **Integridad Referencial** y tratar de tener un esquema robusto, agregando la mayor cantidad de restricciones vistas en clase y que consideres importantes para la lógica del negocio.
- Generar **procedimientos almacenados y disparadores** para apoyar el trabajo y restricciones de tu Base de Datos. Deberás generar **al menos dos de cada uno**.
- **Poblar la base de datos**. Es complicado establecer una cantidad mínima de registros por cada tabla, considera los datos **mínimos necesarios** para poder ver en funcionamiento **todos los aspectos** que este documento describe y trata de que no sean de la misma naturaleza. Puedes apoyarte en **Generate Data** o **Mockaroo** (<https://mockaroo.com>) para esta labor, o alguna otra herramienta que conozcas (en este caso, deberás indicar cuál utilizaste).
- **Construir un diccionario de datos** que contenga definiciones de campos y las abreviaturas o formatos de la base de datos. El formato es libre y pueden utilizar la herramienta de edición que deseen.

Entregables

- El **diseño de la Base de Datos** en modelo **Entidad – Relación** (será necesario utilizar la herramienta **Draw.io** para este entregable)
- El **esquema de la Base de Datos** traducido a **Modelo Relacional**, indicando el conjunto de **dependencias funcionales** válidas para el problema que se te plantea. Deberás apoyarte en **Draw.io** para crear el modelo lógico.
- El **esquema de la Base de Datos normalizado** según la **Tercera Forma Normal**, es imprescindible que **muestres todos** los pasos de la normalización. Se deberá generar el nuevo esquema de base de datos, después de este proceso, utilizar el software indicado en el punto anterior (**Solo si se optó por normalizar**).
- Un **script en SQL** que contenga todas las instrucciones necesarias para crear el **esquema de la Base de Datos**. Debes considerar que deberá ser un esquema restringido que asegure **integridad referencial**. El sistema debe rechazar toda inserción de valores que no estén permitidos en los campos en los que se ha listado el dominio válido. Lo mismo debe ocurrir con las actualizaciones. Se debe vigilar la integridad referencial, de dominio y de entidad. Se deben especificar las políticas para el manejo de dicha vigilancia y reflejarse en la base de datos de manera automática.
- Un **script en SQL** con las instrucciones para **poblar la Base de Datos** con los datos que utilizaste para hacer tus pruebas. La cantidad de datos se deja a tu elección, pero deberá ser la mínima requerida para probar el funcionamiento completo del sistema (un número pobre de tuplas será penalizado).
- Un **script en SQL** donde especifiques **todas y cada una de las consultas SQL** que se solicitan y/o que hayas considerado en tu proyecto final.
- Un **script SQL** donde especifiques los **procedimientos almacenados y disparadores** que diseñaste para la Base de Datos. Deberán estar correctamente documentados e indicar el objetivo que tiene cada uno de ellos
- Los documentos con el diccionario de la base de datos diseñada y con el resultado de las **consultas que diseñaste**.

Todos los documentos para entregar deben tener **excelente presentación**, recuerda que formas parte de un equipo de desarrollo y se deberá hacer una entrega al dueño de la taquería “Taco Riendo”.

Obtención de información

Como se puede observar, es un sistema del cual se pueden extraer una **buena cantidad de reportes** con información relevante. Deberás de proponer al menos **15 consultas no triviales** (deberán de ir **más allá** de la simple utilización de JOIN y ordenamiento) que pongan a prueba tu manejo en SQL para recuperación de información; considera que **deben proporcionar información interesante** al CEO de Taco Riendo, por ejemplo: conocer las ventas de productos en cada sucursal, el número clientes atendidos por turno, sucursal o período; ventas por tipo de productos o por forma de pago,