



# Fundamentos de Bases de Datos

Facultad de Ciencias UNAM

M.I. Gerardo Avilés Rosas  
gar@ciencias.unam.mx

José Enrique Vargas Benítez < josevb@ciencias.unam.mx >

Marco Antonio Pérez Castellanos < universo1993@hotmail.com >



Proyecto Final

Taquerías "Tacoste"

Fecha de entrega

10 de junio de 2017



El señor José "el cuñado" Cruz es el dueño de la cadena mexicana de tacos "**Tacoste**", 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, platillos y bebidas** (se adjunta el menú de la taquería), los precios se colocan con etiquetas fluorescentes y en caso de que deban ser actualizados se pega una nueva etiqueta. A don Pepe le interesa poder llevar un **registro histórico** de los precios y productos; porque le interesa saber en qué momento poder sacar una promoción.
- Se desea otorgar **tickets por cada consumo** (por comensal o mesa).
- Se necesita identificar la **cantidad aproximada** de porciones de ingredientes necesarias a comprar o preparar para vender, esto con el fin de minimizar los desperdicios generados.
- Se desea crear un programa llamado **taquero mucho**, el cual otorgará el **10% del consumo** en puntos, lo cuales podrán ser intercambiados por productos dentro de la taquería, cada punto equivale a **50 centavos**.
- De igual forma, suele manejar promociones como: "**Los jueves pozoleros**", donde el cliente puede pedir todo el pozole que pueda comerse, lo viernes de "**taco amigo**" (2x1 en todos los tacos), los "**martes de tortuga**" (2x1 en tortas y/o hamburguesas).
- Le interesa poder ofrecer **servicio a domicilio** por lo que se necesitarán los datos de los clientes (nombre, dirección, teléfono, correo electrónico); con este servicio, los clientes podrán pagar con **efectivo o tarjeta de crédito/débito**.

- No se tiene un **control de pago a empleados**, se busca tenerlo **vía nómina** y **otorgar seguridad social**; para lograr esto, se necesitará la información personal de los empleados (nombre completo, dirección, RFC, CURP, tipo de sangre, edad, entre otros que consideres relevantes).
- Se busca crear un **programa de bonos para empleados**, con el que cada empleado que cumpla **5 años** laborando en la taquería se le entregarán **\$1,000.00**.
- La solución deberá ser escalable para poder controlar sucursales debido a que se tiene pensando abrir una en los próximos meses y eventualmente, abrir al menos 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 y marca del producto adquirido, precio de compra (mesas, sillas, bancos, platos, servilleteros). También interesa almacenar información de los proveedores, ya que en un futuro interesará hacer alianzas comerciales, a fin de obtener precios más competitivos.
- Se quiere crear una página donde la gente pueda conocer la taquería; aquí, un rubro importante será la **sección de 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 30 mg., de medio kilo o bien por litro.
- Se espera poder **controlar los platillos de la carta**, así como **sus precios** desde alguna aplicación, (p.ej. web) de manera que esto alimente los precios mostrados en la carta en línea.
- Se necesita crear un **aviso de privacidad** para cumplir con la norma establecida por el **Instituto Nacional de Transparencia, Acceso a la Información y Protección de Datos Personales** (INAI) ya que se manipularán datos personales.
- 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. El este cliente existe por cada sucursal.
- Existen **5 tipos de empleados**: parrilleros, taqueros, meseros, cajeros, tortilleros.
- Para el servicio a domicilio se requiere de un **nuevo tipo de empleado**, el cual, además de la información antes especificada requiere del **número de licencia** (si aplica) y si cuenta o no con transporte (**motocicleta** o **bicicleta**).
- Por cuestiones de mercadotecnia, se necesita saber las **distintas formas** en que los clientes pagan los servicios: efectivo, tarjeta de crédito, etc.
- Es importante asegurar integridad y calidad en los datos tanto como sea posible.
- Obtener reportes en cualquier momento en formato PDF o impresión.

El contrato ha sido adjudicado a **Computólogos A.C.** quien deberá presentar una herramienta funcional para **10 de enero de 2018**.

## Consideraciones:

- Antes de iniciar, ten en mente que este es un proyecto 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 ese apartado.
- Leer las reglas de negocio para la cadena de taquerías y comprenderlas lo mejor posible. No se proporcionan necesariamente todas las reglas y será necesario que tengas que especificar requerimientos adicionales para el correcto funcionamiento de tu proyecto. Este documento no pretende ser exhaustivo y, en caso de incertidumbre, se deberá preguntar para afinar.
- Deberás utilizar el **SABD SQL Oracle** para tu implementación y deberás utilizar **todos los aspectos** revisados en el curso (salvo el tema de transacciones).
- Indicar las **llaves primarias**, tratando que sean naturales.
- Crear el modelo **E/R** asociado al caso de uso especificado. Se debe justificar el diseño de manera concisa e indicar de qué forma su esquema empata con la especificación anterior. Recuerda que un buen diseño tiene la cualidad de ser claro en sus entidades con respecto al mundo real. Se recomienda dedicar tiempo a este apartado.
- Efectuar la traducción al **modelo relacional**. Este apartado deberá no ser muy complicado si dedico el tiempo apropiado al punto anterior.
- Con respecto al modelo diseñado, se deben indicar cuáles son las **dependencias funcionales**. Evita aquellas que sean triviales. El esquema debe hacer explícitas las **llaves foráneas**. Evita relaciones redundantes, de lo contrario tendrá más trabajo en el punto siguiente.
- **Normalizar el esquema de bases de datos** del punto anterior, recurre al conjunto de dependencias funcionales que se identificaste. Para fines de este proyecto basta con llegar a **3NF**, puedes buscar **BCNF** en cuyo caso de evaluará como puntos extra. Debes indicar las relaciones resultantes y el procedimiento. Plasma las relaciones del esquema normalizado en un **nuevo esquema relacional**.
- **Construir la base de datos** del esquema del punto anterior. Deberás asegurar **Integridad Referencial**. Deberás tener un esquema de base de datos robusto, de manera tendrás que agregar que será necesario agregar la mayor cantidad de restricciones vistas en clase.
- Se deberán generar algunos **procedimientos almacenados** y **disparadores** para apoyar el trabajo de su base de datos.
- Se deberá **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 trate que estas no sean de la misma naturaleza. Pueden apoyarse en **Generate Data** para esta labor ([www.generatedata.com](http://www.generatedata.com)).
- Tu proyecto deberá incluir una aplicación **funcional** (puede ser escrita en **JAVA** o para Web), deberá incluir una interfaz gráfica sencilla. Tu aplicación debe tener una **alta cohesión** y **bajo acoplamiento**, se recomienda utilizar el **modelo vista controlador** para el diseño e implementación de la misma.
- Se deberá construir un **diccionario de datos** que contenga definiciones de campos y las abreviaturas o formatos de la base de datos.

### Entregables:

- El **diagrama de clases** de tu aplicación (JAVA o WEB).
- El **diseño** de la **Base de Datos** en modelo **Entidad – Relación** (será necesario utilizar la herramienta **DIA o SQL Power Architect** 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 (puedes apoyarte en la herramienta **SQL Power Architect** o alguna otra que conozcas).
- El **esquema de la Base de Datos normalizado** según la **Forma Normal** que consideres conveniente para la solución del problema, es imprescindible que **justifiques tu decisión** y 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.
- 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** 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 (al menos dos de cada uno) 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
- Todos los documentos a entregar deben tener excelente presentación, recuerda que formas parte de un equipo de desarrollo y se deberá hacer una entrega al CEO de la cadena **Tacoste**.
- Se debe verificar que **Dia y/o SQL Power Architect** puedan abrir todos los esquemas que envíen.
- El proyecto que contenga la **aplicación WEB o JAVA (deberá incorporar interfaz gráfica)**. Debes indicar los pasos a seguir para poder instalarlo e implementarlo.

### 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** que pongan a prueba tu manejo en SQL para recuperación de información; considera que deben proporcionar información interesante al **CEO de Tacoste**: 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, productos más vendidos, etcétera. Este punto se deja a tu completa elección. Es importante que no se cree la base de datos teniendo en mente la satisfacción de los reportes listados arriba. Se considera un error grave de diseño orientar el mismo a procesos y no a datos.

Las consultas que se te solicitan forman el núcleo primario de reportes ejecutivos, por esta razón deberán obtenerse y entregarse con presentación profesional. Deberás entregar **documento** (en el editor que prefieras) que muestre el resultado de la evaluación de las **15 consultas que se solicitan**. Deberán tener una presentación ejecutiva (p.e. encabezados, logos, pies de página, etc.).

**Evaluación:**

- Modelo Entidad-Relación y modelo Relacional: **15%**
- Normalización: **15%**
- Scripts SQL (creación y poblamiento): **15%**
- Consultas SQL: **20%**
- Reportes: **10%**
- Aplicación: **15%**
- Documentación: **10%**

La entrega es de acuerdo a los criterios para entrega de tareas (descritos en la página del curso). Deberás de enviar un comprimido (**.zip o .tgz**) a la dirección de correo electrónico de **Enrique**:

[josevb@ciencias.unam.mx](mailto:josevb@ciencias.unam.mx)

que contenga lo siguiente:

- Todos los archivos que se indicaron en el punto anterior.
- Todos los archivos necesarios para poner tu proyecto en funcionamiento.
- Un **documento PDF** en donde reflejes todas las decisiones de diseño que tomaste al momento de plantear tu solución, en este documento deberá aparecer una relación de todos los archivos que estás entregando y que contenga tu nombre completo.

**Restricciones:**

- **Los trabajos serán recibidos únicamente a hasta las 11:59 p.m. del día indicado.**
- **La tarea puede ser en equipos de hasta 3 personas.**
- **Dos días antes de la entrega sólo se responderán aquellas dudas concretas y que reflejen un avance de al menos el 85% del proyecto final.**
- **Aquellas dudas que indiquen que se haya dejado la elaboración del proyecto “al cuarto para las 12”, simplemente no se resolverán.**

**Nota:**

Cualquier duda o comentario que pudiera surgirti al hacer tu tarea, recuerda que cuentas con el grupo de **Facebook** y el correo de la materia: [basesdatos@ciencias.unam.mx](mailto:basesdatos@ciencias.unam.mx), en donde seguramente encontrarás las respuestas que necesitas.

