

---

## **Proyecto No. 2**

### **Desarrollo y Consulta de Bases de Datos Operativas**

#### **(Grupos de 3-4 integrantes)**

#### **Modalidad y fecha de entrega**

- El proyecto se hará en grupos de tres y cuatro integrantes: la idea es que haya un grupo de tres y 2 grupos de 4), y debe de ser enviado antes de la fecha límite de entrega: lunes 14 de octubre a las 23:59. Deberá asignarse con su grupo en Canvas.
- El proyecto se presentará en clases: máximo de 2 grupos por día, durante la semana del 14 al 18 de Octubre. Se deberá entregar previo a las fechas de presentación, aunque el orden y día de presentación será sorteado más adelante.
- No se permitirá la entrega o envío de proyecto más allá de la fecha límite. Todos los grupos entregan el mismo día aunque no presenten todos al mismo tiempo.

#### **Objetivo general del proyecto y problemática**

El proyecto contempla el uso de tecnologías de bases de datos para la creación y carga de modelos de datos, con el objetivo de utilizar lenguaje SQL para la gestión eficiente de restaurantes, enfocándose en reservas de mesas, control de inventario de insumos, y seguimiento de clientes para mejorar la toma de decisiones y la eficiencia operativa.

El proyecto busca modelar y crear una base de datos para un sistema de gestión de restaurantes, permitiendo la interacción con la base de datos desde un lenguaje de programación a elección, para implementar un sistema robusto que facilite las operaciones diarias, la gestión de recursos, y la optimización del servicio al cliente.

**Contexto del problema a modelar:** Los restaurantes a menudo enfrentan desafíos en la gestión de reservas, la optimización del uso de mesas, el control del inventario de alimentos y bebidas, y el seguimiento de las preferencias y comportamientos de los clientes. Para grandes cadenas de restaurantes, sin un sistema centralizado, se dificulta la toma de decisiones informadas, lo que puede llevar a pérdidas financieras, desperdicio de alimentos y una experiencia de cliente deficiente.

La información clave incluye detalles sobre las reservas (fecha, hora, mesa asignada, cliente), la gestión de insumos (existencias, fechas de caducidad), el personal del restaurante (meseros, chefs, gerentes), y el historial de visitas y preferencias de los clientes. El acceso a esta información permite mejorar la experiencia del cliente, prever la demanda, y gestionar los recursos de manera eficiente.

#### **Descripción del proyecto**

El proyecto consiste en la implementación de una aplicación de escritorio (consola) o web que utilice una base de datos relacional para soportar un sistema de gestión de restaurantes. Los casos de uso transaccionales que debe soportar incluyen:

1. **Sign in y log in de usuarios:** Incluyendo roles como administrador, mesero, y gerente.
2. **Gestión de reservas:** Registro y búsqueda de reservas de mesas, alertando sobre la disponibilidad de mesas y permitiendo su asignación eficiente.
3. **Gestión de inventario:** Búsqueda y control de insumos de cocina (alimentos, bebidas, utensilios), generando alertas sobre aquellos que están por terminarse o que están próximos a caducar (mínimo 15% del inventario para generar una alerta).
4. **Historial de clientes:** Registro completo del historial de visitas de cada cliente, incluyendo detalles de las reservas, preferencias (por ejemplo, platos favoritos), y observaciones sobre su experiencia.
5. **Control de cambios:** Bitácora automática para registrar inserciones, actualizaciones y eliminaciones en la base de datos (por ejemplo, cambios en el inventario, modificaciones de reservas).

Deberá implementar el uso de temas vistos en clase, incluyendo hasta **Índices y Triggers**, sobre la base de datos creada.

#### **Suposiciones que debe (o podrá) de tomar en cuenta:**

- La aplicación y su base de datos se encuentran alojadas en un servidor central al que los restaurantes de una cadena pueden acceder de forma remota. Esto quiere decir que desde el código que utilice para construir la lógica del proyecto, deberá de estar conectado directamente a su base de datos.
- Los clientes pueden tener múltiples reservas y preferencias.
- Los insumos de cocina están a nivel de ingredientes, no a nivel de platos. La recomendación es que haga un menú sencillo que contenga más o menos los mismos ingredientes y que tampoco sea tan extenso.
- Los insumos de cocina tienen fechas de caducidad y niveles mínimos de existencias.
- Trate de replicar ese mismo menú para todos los restaurantes (sedes).
- Obviamos detalles como la disposición física de la cocina o el diseño interior del restaurante.

Los pasos sugeridos para el inicio correcto del proyecto son (en este mismo orden):

1. Instalación de servicio PostgreSQL.
2. Diseño de base de datos mediante diagrama Entidad-Relación
3. Traducción del diseño al modelo relacional
4. Construcción del modelo relacional (tablas), incluyendo las correspondientes llaves primarias, llaves extranjeras, restricciones, índices, triggers, Etc.
5. Inserte datos de ejemplo. Trate de no usar una base de datos robusta al inicio para poder compartirla fácilmente con el resto de miembros del grupo.
6. Implementación de la funcionalidad del sistema. Trate de hacerlo por módulos, divídase el trabajo para cada caso de uso transaccional que deberá tener su sistema.

Adicionalmente se debe implementar un módulo de reportería para usuarios administradores, que incluya los siguientes reportes:

- i. Top 10 de los platos más vendidos.
- ii. Top 10 de los clientes más frecuentes.
- iii. Top 5 de los clientes con mayores reservas y su preferencia de platos.
- iv. Reporte mensual de insumos a punto de terminarse o caducar.
- v. Comportamiento de sucursales con mayor cantidad de reservas y ventas.

### Especificación de tecnología

- Sistema manejador de base de datos: PostgreSQL
- Lenguaje de programación a discreción

### Temas a reforzar

- Lenguaje SQL: DDL / DML (PostgreSQL)
- Reportería descriptiva basada en consultas SQL
- Aplicación de Base de Datos hacia el negocio.

### Documentos a entregar

- Documentación de modelo de datos correspondiente a la Base de Datos utilizada.
- Se debe generar un diagrama E/R de la base de datos utilizada
- Archivo comprimido con el código fuente de la aplicación desarrollada
- Código fuente correspondiente utilizado en la creación de su aplicación, incluyendo también cualquier dato pre cargado existente sobre la BD que construyan

### Evaluación

- Diseño de base de datos y DER: 15%
- Implementación de base de datos: 15%
- Casos de uso transaccionales: 40%
- Implementación de reportes: 20%
- Calidad de la presentación: 10%

○ **Total: 100% (15 puntos netos)**

### Puntos extras (10 pts máximo):

- Creatividad, apartado físico y de diseño (frontend) del aplicativo construido.