



## Práctica 02 - Análisis de Requerimientos

### Fundamentos de Bases de Datos

#### Integrantes:

- Aceves Higareda Mayra Guadalupe  
ahigareda.mayra@ciencias.unam.mx  
310646246
- Badillo Lora Carlos  
carlosb@ciencias.unam.mx  
415083504
- Moreno Ruíz Jesús Fernando  
chuchini@ciencias.unam.mx  
414001967
- Servín Mote Edson  
edson@ciencias.unam.mx  
308292503

**Profesor:** Gerardo Avilés Rosas

**Laboratorio:** Erick Orlando Matla Cruz

Universidad Nacional Autónoma de México  
Facultad de Ciencias



---

## 1. Propósito del sistema (requerimientos candidato)

El propósito es hacer un sistema computacional que ayude a agilizar la operación de una taquería y proporcionar información relevante sobre su funcionamiento; para ello se requiere considerar los datos que se describen a continuación con las siguientes características:



- **Para el cliente:** Se requiere asignar un número de cliente único, así como saber su nombre, teléfono, correo electrónico para que reciba promociones de la taquería, su fecha de nacimiento para recibir un regalo por su cumpleaños y su dirección, para hacer entregas a domicilio.



- **Para el empleado:** Se necesita identificar a los empleados de manera única. Se cuenta con su nombre, el papel que desempeña dentro de la taquería, su fecha de ingreso, fecha de nacimiento, así como el salario que recibe.



- **Para las materias primas:** Se necesita contar con un inventario de las materias primas y productos con los que se cuentan, con el fin de poder hacer un presupuesto y planeación para futuras compras.



- **Para la sucursal:** Cada sucursal tiene una dirección, horario de servicio y varios números de teléfono para poder recibir pedidos y entregarlos a domicilio.



---

## 2. Comprensión del contexto del sistema

Los **empleados** compran las materias primas y realizan el inventario. Se necesita saber la fecha del pedido y el nombre del empleado que realiza la compra. Cada empleado trabaja en una única sucursal. Los *meseros* tienen asignadas cierto número de mesas y las atienden de manera simultánea. Toman la orden de los clientes, anotando la cantidad y la palabra/número clave de lo que se solicita. El *administrador* se encarga de registrar los pedidos por mesa; generar las cuentas y realizar los pagos con tarjeta de crédito/débito. Los *taqueros* y *ayudantes* únicamente se dedican a preparar las órdenes.

El *supervisor* se encarga de verificar que cada quien realice su trabajo; toma los datos de los clientes que no están registrados para que reciban promociones; actualiza el menú y los precios; genera reportes sobre los clientes y las órdenes que han realizado.

Los **clientes** eligen del menú lo que van a consumir y pagan la cuenta, recibiendo una lista detallada de lo que se consume y el total a pagar.



*Entidades:* Cliente, Empleado, Orden, Pago, Sucursal, Materias Primas, Menú.

### 3. Requerimientos funcionales:

- a) Añadir orden (Cliente)
- b) Actualizar menú (Cliente)
- c) Añadir cliente (Administrador)
- d) Borrar cliente (Cliente)
- e) Buscar cliente (Administrador)
- f) Pagar factura (Cliente)
- g) Eliminar producto del pedido (Cliente)
- h) Añadir producto del pedido (Cliente)
- i) Consultar saldo (Administrador)

A continuación presentamos diagramas de secuencia que modelan algunos requerimientos funcionales:

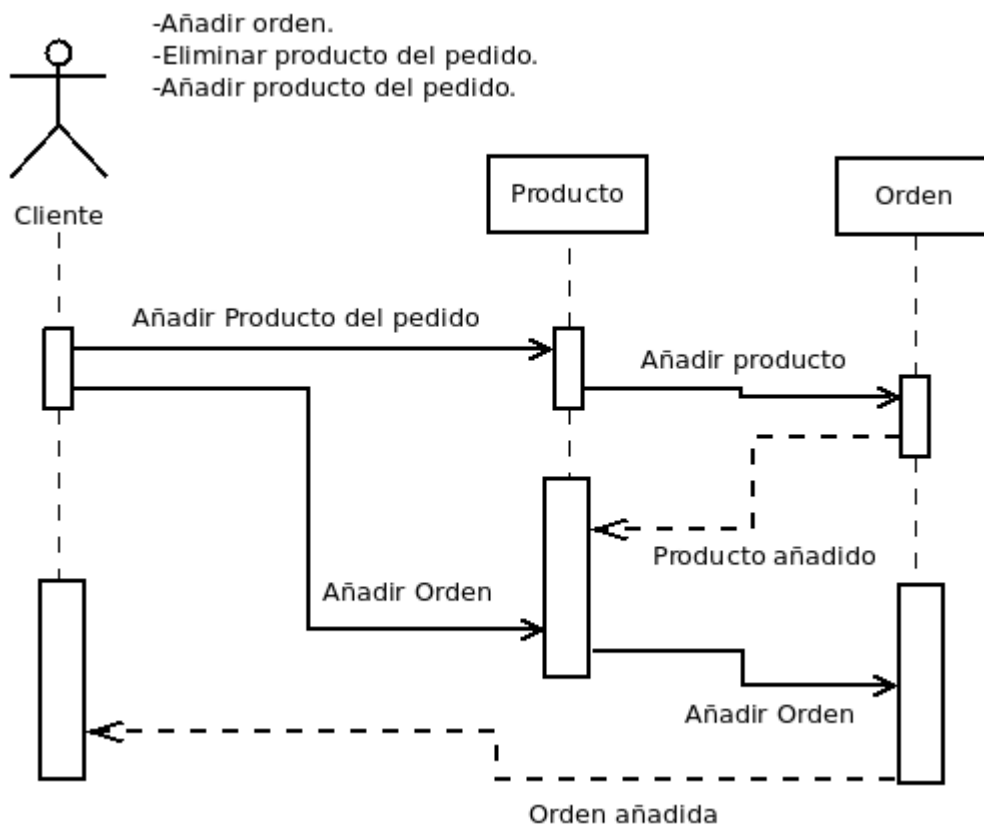


Figura 1.0. Diagrama de secuencia de los R.F.

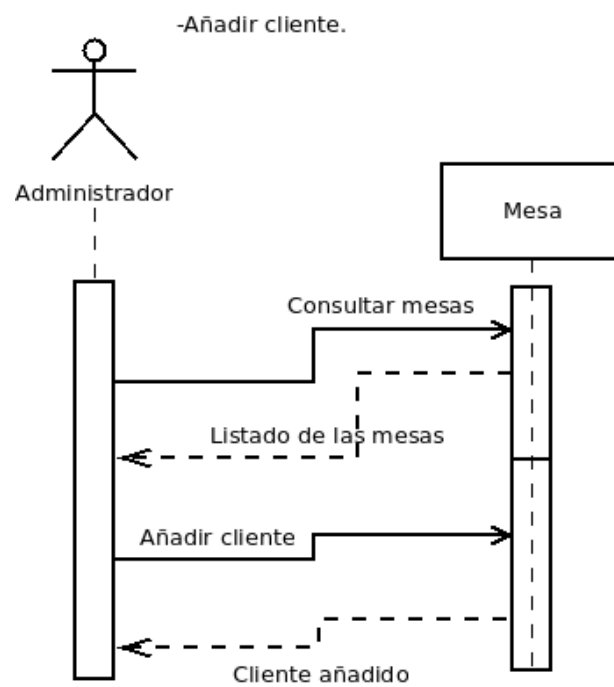


Figura 2.0. Diagrama de secuencia del R.F.

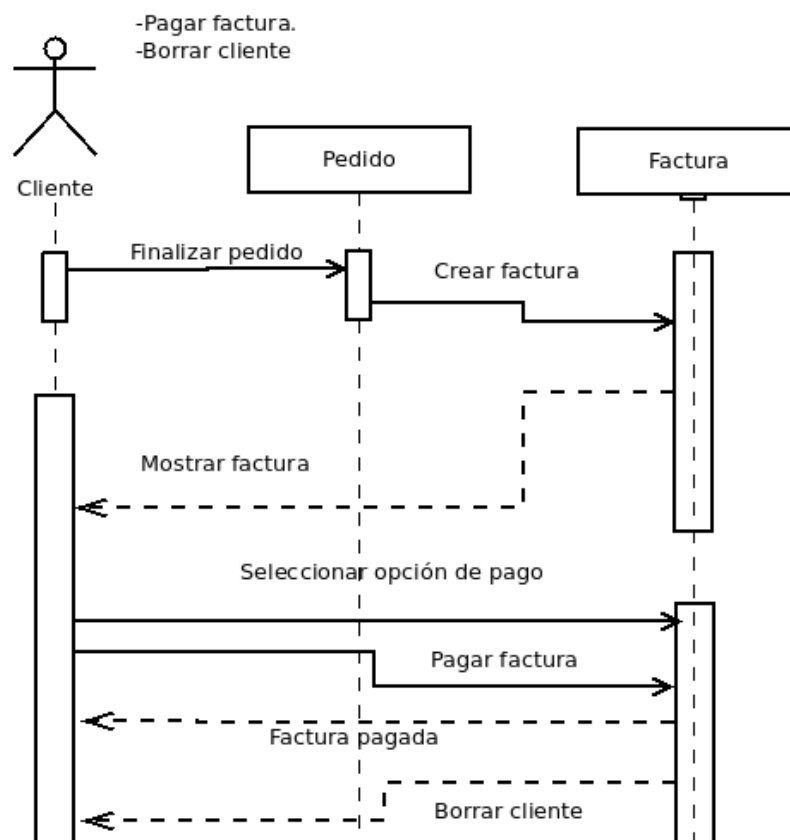


Figura 3.0. Diagrama de secuencia del R.F.

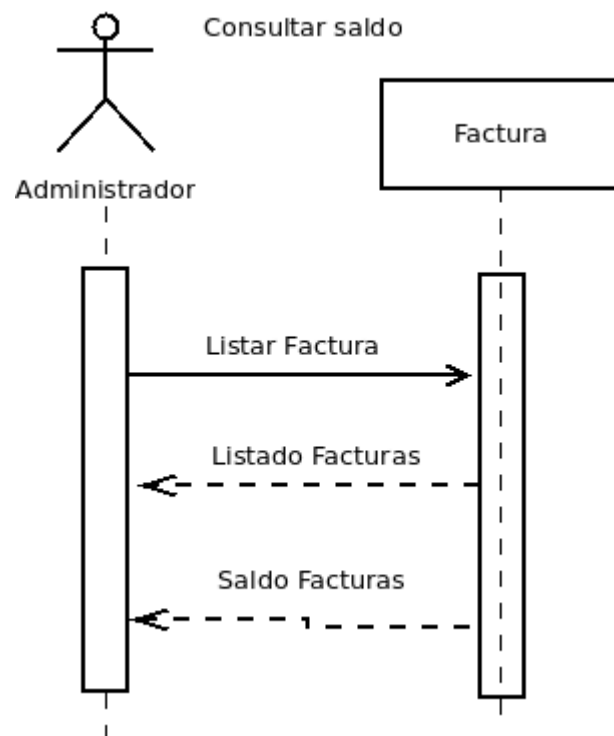


Figura 4.0. Diagrama de secuencia del R.F.

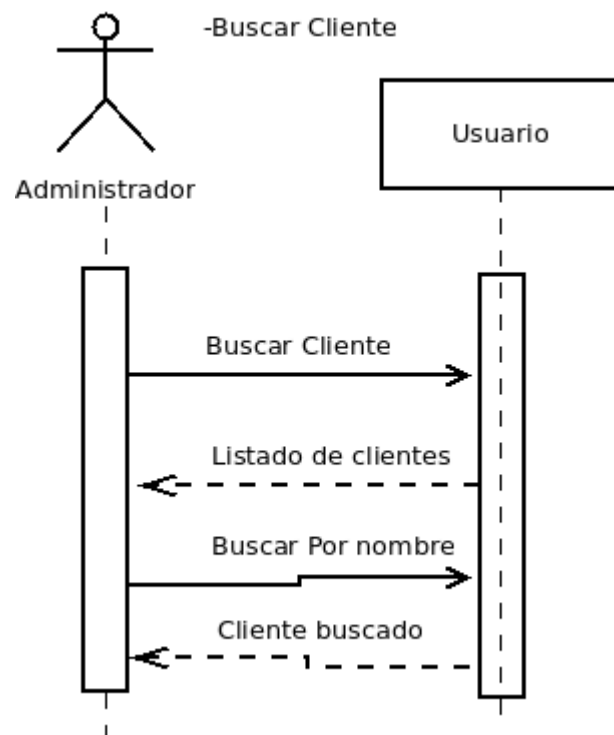


Figura 5.0. Diagrama de secuencia del R.F.

---

## 4. Requerimientos no funcionales

### a) *Requerimientos no funcionales asociados a requerimientos funcionales.*

- i) Controlar todas las entradas realizadas por todos los tipos de usuario del sistema.
- ii) Contar con las respectivas políticas de la empresa.

### b) *Requerimientos no funcionales no asociados a requerimientos funcionales.*

- i) La interfaz debe ser sencilla de comprender y accesible para cualquier tipo de usuario.
- ii) El programa y la manipulación de archivos se realiza en Java. La base de datos es almacenada en archivos con formato “.txt”.
- iii) El sistema debe contar con un tiempo de respuesta corto.
- iv) El sistema debe tener acceso únicamente a personal autorizado mediante un buen mecanismo de seguridad.

## Ventajas y Desventajas de un Sistema de Archivos

### Ventajas

- Son fáciles de diseñar, construir y usar.
- Permiten almacenar cualquier tipo de información sencilla.
- Opción perfecta para aplicaciones pequeñas, medianas e independientes.



### Desventajas

- Hay que definir estructuras de almacenamiento.
- Se debe garantizar la fiabilidad (caídas o accesos no autorizados).
- Proporcionar mecanismos de manipulación de la información.
- No se pueden evitar los datos anómalos.
- Los archivos convencionales son difíciles de ampliar.

---

## *Ventajas y Desventajas de una Base de Datos*

En una base de datos, tenemos las siguientes características:

- Un solo depósito de datos definido una sola vez y puede accederse por varios usuarios.
- Representación de relaciones complejas entre datos.
- El SMBD tiene la BD y su descripción completa.
- Los programas de acceso se escriben con independencia de archivos específicos.
- Control de redundancia.
- Restricciones de acceso.
- Restricciones de integridad.
- Proporciona respaldo y recuperación.
- Reducción de tiempo de desarrollo.
- Flexibilidad.
- Disponibilidad de datos actualizados.



Sin embargo, uno de los principales problemas de las bases de datos es su costo de mantenimiento y de seguridad. Además, en casos donde la velocidad de procesado sea muy importante, las bases de datos muestran su peor cara. El tiempo necesario para crear, leer, modificar o eliminar un dato en una base de datos es superior al tiempo que se necesitaría con un archivo convencional.



---

---

## *Conclusiones*

En el caso de la práctica 2 desarrollada en Java y usando un sistema de archivos, podemos hacer la comparación mediante las ventajas y desventajas de la sección anterior; y principalmente, con la reducción de tiempo en el desarrollo del sistema, ya que parece ser que es más complicado debido a que el programa en cuestión no asegura que sea muy eficiente; además que la seguridad, de igual manera, no es muy buena para situaciones más reales.

Es una buena manera de aprender las diferencias entre el desarrollo de un programa y la implementación que sigue una lógica más adecuada para una base de datos.

---

## *Bibliografía*

- Avilés, G., (s. f.), *Modelo Entidad Relación*, [diapositivas de PowerPoint].  
Recuperado de:  
<http://www.engineerhunters.com/gerardo/fbd/material/uploaden/Gerardo/Presentaciones/02ModeloER.pdf>
- Fernández, V. “Desarrollo de sistemas de información: una metodología basada en el modelado”. Editorial UPC, 2006, España, 220 p.