

Melisa Dayana Mendizabal Meléndez	Interacción humano computador
Carné: 23778	Sección 40

### Reflexión Individual sobre el Proceso

- **¿Por qué eligieron este sistema de gestión de bases de datos (DBMS)?  
¿Qué ventajas y desventajas tiene en comparación con otros?**

Se optó por un DBMS relacional porque ofrece robustez, escalabilidad y facilidad para manejar relaciones complejas mediante claves primarias y foráneas.

#### **Ventajas:**

- **Integridad y consistencia de datos:** Gracias a las restricciones (NOT NULL, UNIQUE, CHECK) y las relaciones definidas, se asegura la calidad de la información. Debido al tipo de datos que se maneja, estos evitan redundancia, además que el no usar una relacional es más complejo almacenar datos que puede comportarse como objetos.
- **Consultas complejas:** SQL permite realizar operaciones complejas con JOIN, subconsultas y agrupaciones.
- **Escalabilidad:** Es sencillo agregar nuevas entidades y relaciones conforme el sistema crece.

#### **Desventajas:**

- **Complejidad en la configuración y mantenimiento:** Puede requerir conocimientos avanzados para optimizar consultas y manejar transacciones.
  - **Rendimiento en sistemas altamente distribuidos:** En entornos con altísima concurrencia o volúmenes masivos de datos, a veces se prefieren soluciones NoSQL, la cual no es la situación de la farmacia.
- **¿Qué estándares o criterios usaron para diseñar su base de datos?**
    - Se siguieron las normas de normalización para evitar redundancias y asegurar la integridad referencial.
    - Se aplicaron estándares de nombres claros y consistentes en tablas, columnas y restricciones.
    - Se definieron criterios de integridad (uso de claves primarias, foráneas y restricciones CHECK) y se utilizaron valores por defecto para automatizar registros (como CURRENT\_TIMESTAMP).
    - Se priorizó la eficiencia en las consultas a través de índices en campos críticos.
  - **¿Cuáles son las entidades más importantes del modelo y por qué?**
    - **Inventario:** Es la entidad central, ya que gestiona todos los movimientos de productos entre locales, compras, ventas y descartes.

Registra información esencial como fecha, usuario, precios y cantidades.

- Productos y Lotes: Permiten gestionar la información de los medicamentos, especialmente considerando la fecha de vencimiento y la existencia de múltiples lotes para un mismo producto.
- Usuarios y Roles: Son fundamentales para controlar el acceso y garantizar la seguridad del sistema, permitiendo asignar permisos específicos a cada perfil.
- Ventas y Compras: Estas entidades se encargan de registrar transacciones, vinculando a clientes, proveedores y los movimientos en inventario, lo que es crítico para el seguimiento financiero.

- **¿Cómo aplicaron las técnicas de normalización en su diseño? ¿Qué problemas evitaron gracias a esto?**

Se aplicaron las primeras tres formas normales:

- Primera forma normal (1FN): Cada celda almacena un único valor, eliminando la duplicidad de información.
- Segunda forma normal (2FN): Se eliminan dependencias parciales mediante la separación de datos en distintas tablas (por ejemplo, separar los números de teléfono en una entidad aparte).
- Tercera forma normal (3FN): Se evita la dependencia transitiva, lo que ayuda a mantener la consistencia y facilita actualizaciones.

Problemas evitados:

- Redundancia de datos: Se evita almacenar información duplicada que podría generar inconsistencias.
- Anomalías de actualización: Con un diseño normalizado, se reducen los errores al actualizar o eliminar registros.
- Mantenimiento complicado: La estructura modular facilita la gestión y el escalamiento del sistema.

- **¿Cómo definieron restricciones y valores por defecto para garantizar la integridad de los datos?**

- Se establecieron restricciones como NOT NULL, UNIQUE y CHECK en campos críticos (por ejemplo, en el email de Usuarios o el descuento en Promociones) para evitar datos inválidos.
- Se asignaron valores por defecto en columnas como creacion y fecha (usando CURRENT\_TIMESTAMP) para automatizar el registro de la fecha y hora de inserción de datos.
- Las claves foráneas se definieron para asegurar la integridad referencial, garantizando que las relaciones entre tablas sean coherentes.

- **¿Cómo abordaron los cambios en la estructura de la base de datos?**
  - Se utilizó un enfoque iterativo y modular, permitiendo la evolución del esquema sin afectar las operaciones existentes.
  - Se planificaron migraciones de datos mediante scripts SQL que actualizan la estructura conforme se identifican nuevas necesidades. Además de realizar la actualización en git, también se fue modificando el diagrama.
  - La utilización de un DBMS relacional facilita la aplicación de cambios en la estructura gracias a su soporte para transacciones y herramientas de gestión de esquemas.

- **¿Cómo seleccionaron los datos de prueba para garantizar que el diseño es funcional?**

Nos basamos en una copia del inventario real de la farmacia, ya que una parte se encuentra almacenada en un documento de Excel y otra parte en físico. Luego, fuimos adaptando el formato existente al de la propuesta de base de datos, para luego realizar los queries.

- **¿Cuál fue tu contribución específica en el desarrollo del proyecto?**  
**¿Cómo se organizó el trabajo en el equipo?**

Participé en todo el diseño del esquema relacional, definiendo las tablas y sus relaciones, y colaboré en la implementación de restricciones, índices y valores por defecto. También contribuí en la planificación de las posibles migraciones, pero no las ejecuté, solo cambié el diagrama.

Organización del equipo: Se distribuyeron las tareas en función de las fortalezas de cada miembro diseño, implementación, pruebas. Se utilizó un sistema de control de versiones (como Git) para gestionar de forma colaborativa el desarrollo del proyecto.

- **¿Sientes que trabajaste equitativamente en comparación con tus compañeros? ¿Qué hubieras hecho diferente en este proyecto?**

No, porque inicialmente nos repartimos la entrega en base a la clase de software, por obvias razones, en esa clase no se pedían tantos detalles como aquí, además que el hecho de que cada uno hiciera un diagrama parecía lo más justo. El problema fue la complejidad que se fue dando entre más datos y requisitos funcionales se tomaban en cuenta, al final mi aportación fue un poco más pesada que la del resto.