



Universidad Nacional del Nordeste

Facultad de Ciencias Exactas y Naturales y Agrimensura

Licenciatura en Sistemas de Información

Base de Datos I

Año Lectivo 2025

SISTEMA DE GESTIÓN DE VENTAS “Gestor Franquicias”

Grupo N° 3

Integrantes:

Apellido y Nombre: Benitez, Lucas Emmanuel

DNI: 43930915

Apellido y Nombre: Diaz, Cristian Leandro

DNI: 42734520

Apellido y Nombre: Fernandez Gotusso, Maria Daniela

DNI: 43822520

Apellido y Nombre: Machuca, Juan Gabriel

DNI: 40048379

ÍNDICE

CAPÍTULO I: INTRODUCCIÓN	0
a) Tema	0
b) Definición o planteamiento del Problema	0
c) Objetivo del Trabajo Práctico	1
i) Preguntas Generales	1
ii) Preguntas Específicas	1
iii) Objetivos Generales	1
iv) Objetivos Específicos	2
d) Descripción del Sistema	2
e) Alcance	2
CAPÍTULO II: MARCO CONCEPTUAL O REFERENCIAL	4
CAPÍTULO III: METODOLOGÍA SEGUIDA	8
a) Cómo se realizó el Trabajo Práctico	8
b) Herramientas (Instrumentos y procedimientos)	8
CAPÍTULO IV: DESARROLLO DEL TEMA	10
a) Diagrama de Modelo Relacional	10
b) Diccionario de Datos	10
CAPÍTULO V: CONCLUSIONES	20
BIBLIOGRAFÍA	21

CAPÍTULO I: INTRODUCCIÓN

a) Tema

Automatización de la gestión de ventas e inventario para una red de franquicias.

b) Definición o planteamiento del Problema

Actualmente, muchas tiendas de comercio minorista, en particular las que operan bajo un modelo de franquicias, dependen de procesos manuales para la gestión de ventas, el control de inventarios y la administración de clientes. Este enfoque tradicional es propenso a errores, genera una alta ineficiencia operativa y retrasa la toma de decisiones estratégicas.

La falta de un sistema de gestión centralizado y automatizado en cada sucursal de la franquicia provoca una desconexión entre las áreas clave (ventas, inventario y atención al cliente). Como resultado, la productividad se ve afectada, ya que el personal invierte tiempo en tareas repetitivas y de registro manual, como la actualización de inventarios o la generación de reportes. Esta situación conduce a problemas críticos, como la falta de visibilidad en tiempo real del stock, la duplicación de datos y la inconsistencia en los registros, lo que impide una respuesta ágil a los cambios en la demanda del mercado y afecta negativamente la calidad del servicio al cliente.

La fragmentación de la información dificulta la obtención de reportes consolidados y precisos a nivel de franquicia, lo que limita la capacidad de la gerencia para supervisar el rendimiento general, identificar oportunidades de mejora y tomar decisiones basadas en datos confiables.

c) Objetivo del Trabajo Práctico

El objetivo principal de este trabajo es desarrollar un sistema de gestión automatizado, denominado Gestor Franquicias, que proporcione una solución integral a los problemas de ineficiencia y falta de control en las operaciones diarias de una red de franquicias.

Además, el proyecto incorpora la investigación y aplicación de conceptos avanzados como el análisis de datos para la toma de decisiones, la implementación de un robusto sistema de permisos a nivel de usuario en la base de datos y la optimización de consultas para garantizar un rendimiento óptimo.

i) Preguntas Generales

¿Cómo se puede optimizar la gestión de ventas, inventarios y clientes para una red de

franquicias?

ii) Preguntas Específicas

- ¿Cómo podemos asegurar que la búsqueda de una factura de venta sea eficiente y rápida?
- ¿Qué estrategia permite optimizar la carga y actualización del inventario de productos?
- ¿Cómo se puede implementar un control de stock efectivo y en tiempo real para cada producto?
- ¿De qué manera se puede gestionar el registro y la eliminación de clientes de forma segura y eficiente?
- ¿Cómo podemos agilizar el proceso de registro de una venta, minimizando errores y tiempo de espera?

iii) Objetivos Generales

Desarrollar un sistema de gestión automatizado que resuelva la problemática de la falta de control y la ineficiencia en las operaciones de ventas, inventario y clientes en un modelo de franquicias.

iv) Objetivos Específicos

- Reducir significativamente el tiempo y los errores asociados al registro manual de inventario y a las tareas de búsqueda de datos.
- Optimizar los procesos de búsqueda y recuperación de facturas de venta.
- Proveer informes y reportes detallados que permitan analizar el rendimiento de ventas y el comportamiento del inventario.
- Implementar un sistema de control de stock que evite la falta de productos y facilite la toma de decisiones sobre reabastecimiento.

d) Descripción del Sistema

El sistema Gestor Franquicias está diseñado para ser la herramienta central de administración de tiendas físicas de artículos de bazar y polirrubro con múltiples sucursales en la provincia de Corrientes. Su propósito es centralizar el registro y la administración de ventas, clientes, productos y otros datos comerciales.

El sistema cuenta con un esquema de perfiles de usuario que asigna funcionalidades específicas según el rol del personal: gerente, vendedor y administrador del sistema. Los vendedores tienen acceso a la gestión diaria de clientes y productos, así como al procesamiento de ventas. El gerente general supervisa el rendimiento de las ventas, gestiona el personal y accede a informes detallados. Finalmente, el administrador del sistema es responsable del mantenimiento, la seguridad y las copias de seguridad de la base de datos.

e) Alcance

Este proyecto se centrará en el procesamiento y la administración de datos clave relacionados con las ventas. El sistema abordará las interacciones fundamentales del negocio: el registro de qué vendedor realizó una venta, a qué cliente se le vendió, qué productos fueron adquiridos y con qué método de pago.

El alcance del proyecto no incluye funcionalidades de integración con proveedores para el reabastecimiento de stock, ni se profundizará en el análisis de datos a un nivel que exceda los informes y reportes básicos para la toma de decisiones a nivel gerencial.

CAPÍTULO II: MARCO CONCEPTUAL O REFERENCIAL

Tema 1: "Procedimientos y funciones almacenadas"

- **Procedimientos Almacenados (Stored Procedures)**

Los Procedimientos Almacenados representan bloques de código SQL precompilado que residen en el servidor de la base de datos. Funcionan como subrutinas o programas que se ejecutan bajo demanda, actuando como una unidad transaccional y lógica.

- **Propósito:** Sirven para ejecutar tareas complejas de manipulación de datos (como la inserción, actualización o eliminación), realizar lógica de negocio condicional, o ejecutar varias consultas en secuencia.
- **Interacción:** Pueden recibir parámetros de entrada para personalizar su ejecución y devolver parámetros de salida con resultados o mensajes de estado.
- **Ventajas Clave:** Ofrecen un mejor rendimiento debido a su compilación previa y facilitan la seguridad, ya que permiten a los usuarios interactuar con los datos solo a través de la lógica encapsulada, sin necesidad de permisos directos sobre las tablas subyacentes.

- **Funciones Almacenadas (Stored Functions)**

Las Funciones Almacenadas son objetos de base de datos diseñados para la devolución obligatoria de un resultado. Su propósito primordial es realizar cálculos, transformaciones de datos o devolver información específica.

- **Diferencia Fundamental:** A diferencia de los procedimientos, una función siempre regresa un valor (puede ser un valor escalar, como un número o texto, o una tabla de datos).
- **Uso:** Están integradas directamente en el lenguaje de consulta. Se pueden invocar dentro de sentencias SQL, como la lista de columnas de un SELECT o en las cláusulas WHERE o HAVING.
- **Restricciones:** Generalmente, las funciones están sujetas a reglas más estrictas y no están destinadas a realizar operaciones que alteren el estado de la base de datos (es decir, no deben modificar datos en tablas de forma permanente).

Tema 2: “Gestión de Transacciones y Transacciones Anidadas”

El manejo de datos en un sistema de administración centralizado demanda un mecanismo riguroso que garantice la integridad operacional. Las transacciones

representan la unidad fundamental de trabajo, diseñadas para ser indivisibles. Su implementación es esencial, ya que adhieren a las propiedades ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad), asegurando que las operaciones críticas (como el registro de una venta que afecta tanto a la tabla Venta como al stock del Producto) se ejecuten completamente o no se ejecuten en absoluto. Dentro de esta estructura, las transacciones anidadas se emplean para modular y estructurar operaciones más complejas. Permiten iniciar transacciones menores dentro del contexto de una transacción principal ya activa. Este enfoque es crucial para gestionar la lógica de negocio con múltiples dependencias, ya que facilita el manejo de errores intermedios. Garantiza que, si una sub-operación falla, la función de reversión (ROLLBACK) pueda restablecer el sistema a su estado original sin comprometer la integridad de la base de datos.

Tema 3: "Optimización de consultas a través de índices"

- Fundamentos de los Índices

Un índice en una base de datos relacional es una estructura de datos especializada diseñada para acelerar la búsqueda de información. Se asemeja al índice de un libro, donde en lugar de leer cada página, se localiza rápidamente la ubicación del contenido deseado. El objetivo primordial de un índice es reducir drásticamente el tiempo de ejecución de las consultas (Query Time) al evitar el escaneo completo de la tabla (Full Table Scan).

- Clasificación y Aplicaciones de los Índices

Los sistemas de gestión de bases de datos (DBMS) ofrecen una variedad de índices, cada uno optimizado para diferentes escenarios:

- Índice Único (Unique Index): Más allá de la velocidad, su función principal es garantizar la unicidad de los valores en una o más columnas, previniendo la inserción de datos duplicados y asegurando la integridad de los datos.
- Índice Geoespacial (Spatial Index): Específicamente diseñado para gestionar y consultar datos con dimensiones geométricas, como ubicaciones, mapas o áreas geográficas.
- Índice Hash (Hash Index): Emplea una función hash para mapear claves de columna a direcciones de memoria. Son extremadamente rápidos para búsquedas exactas de igualdad, generalmente almacenados en memoria para un acceso instantáneo.
- Índice Filtrado (Filtered Index): Permite aplicar el mecanismo de

indexación solo a un subconjunto de las filas de la tabla, optimizando el espacio en disco y mejorando la eficiencia para consultas que siempre aplican un criterio de filtrado específico.

Tema 4: "Vistas y vistas indexadas"

1. Vistas (Views): La Capa de Abstracción

Una Vista es una tabla lógica o abstracta que actúa como una ventana personalizada hacia los datos subyacentes de la base de datos. Se definen mediante una sentencia SQL almacenada, pero no guardan datos propios; su contenido se genera en tiempo real (dinámicamente) cada vez que son consultadas, utilizando la información de las tablas base.

- **Funciones Arquitectónicas y Ventajas**

Las vistas son una herramienta esencial para la administración y el diseño de sistemas, ofreciendo múltiples beneficios:

- **Simplificación de Consultas:** Abstractúan la complejidad de joins intrincados o lógica compleja. Un usuario puede consultar una vista simple en lugar de reescribir una consulta extensa, fomentando la reutilización de código.
- **Mecanismo de Seguridad:** Permiten establecer una capa de seguridad fina. Se puede conceder acceso a una vista que solo muestre columnas no sensibles, mientras que las columnas con datos críticos (como salarios o información personal) permanecen ocultas en las tablas base.
- **Personalización:** Adaptan la percepción del esquema para diferentes usuarios o aplicaciones, asegurando que cada grupo solo vea el subconjunto de datos que le es pertinente.
- **Compatibilidad:** Facilitan la evolución del esquema de la base de datos sin romper la compatibilidad con aplicaciones antiguas, ya que se puede modificar una tabla base y usar la vista para mantener la estructura de datos que la aplicación espera.

2. Vistas Indexadas (Indexed Views o Materialized Views)

Las Vistas Indexadas (también conocidas como Vistas Materializadas en otros sistemas) representan una poderosa técnica de optimización que rompe con la naturaleza virtual de las vistas tradicionales.

- **Naturaleza Materializada:** A diferencia de una vista estándar, el conjunto de

resultados de una vista indexada se calcula y se almacena físicamente en el disco como una tabla real.

- Optimización del Rendimiento: Su principal objetivo es la mejora drástica del rendimiento para consultas específicas (a menudo aquellas con agregaciones complejas, como GROUP BY o SUM).
- Estructura de Índices: Llevan un índice asociado que se crea directamente sobre esta tabla almacenada. Este índice permite que el optimizador de consultas acceda a los resultados precalculados de manera extremadamente eficiente, en lugar de recalcular la vista a partir de las tablas base cada vez.

CAPÍTULO III: METODOLOGÍA SEGUIDA

En este capítulo se describe el proceso de trabajo adoptado para el desarrollo del proyecto, así como las principales herramientas y procedimientos utilizados. El objetivo es ofrecer una visión clara de cómo se organizó el equipo y qué recursos permitieron llevar adelante las distintas etapas del sistema.

a) Cómo se realizó el Trabajo Práctico

El desarrollo de este proyecto se llevó a cabo mediante un enfoque grupal y colaborativo, aprovechando herramientas digitales para garantizar una gestión de trabajo remoto fluida y eficiente.

- Control de Versiones y Gestión de Código (GitHub): Utilizamos GitHub para el control de versiones, lo que permitió a todos los miembros del equipo integrar sus contribuciones de código de manera organizada, facilitando el seguimiento del progreso y la trazabilidad de los cambios.
- Documentación y Archivos (Google Drive): Empleamos Google Drive para centralizar el almacenamiento y la organización de toda la documentación relevante, asegurando que el material fuera accesible y estuviera actualizado para todos los participantes.

b) Herramientas (Instrumentos y procedimientos)

Para la construcción del sistema de gestión de ventas, se seleccionó un conjunto de herramientas específicas que cubrieron las fases de diseño, implementación y documentación de la base de datos:

- Diseño de la Base de Datos (Modelado Conceptual y Lógico):
 - ERD Plus: Herramienta clave para el diseño inicial del Modelo Entidad-Relación (MER).
 - Diagrama Entidad-Relación (DER): Representación gráfica que visualiza las entidades principales y sus relaciones dentro del sistema.
 - Diagrama de Modelo Relacional: Representación esquemática de la base de datos que muestra las tablas definitivas con sus campos (columnas) y tipos de datos correspondientes.
 - SQL Server: Servidor de base de datos elegido para la persistencia y la gestión de la información transaccional del sistema.
- Documentación de la Estructura:
 - Diccionario de Datos: Documento esencial que detalla de forma exhaustiva los atributos de la base de datos (campos, tipos de datos,

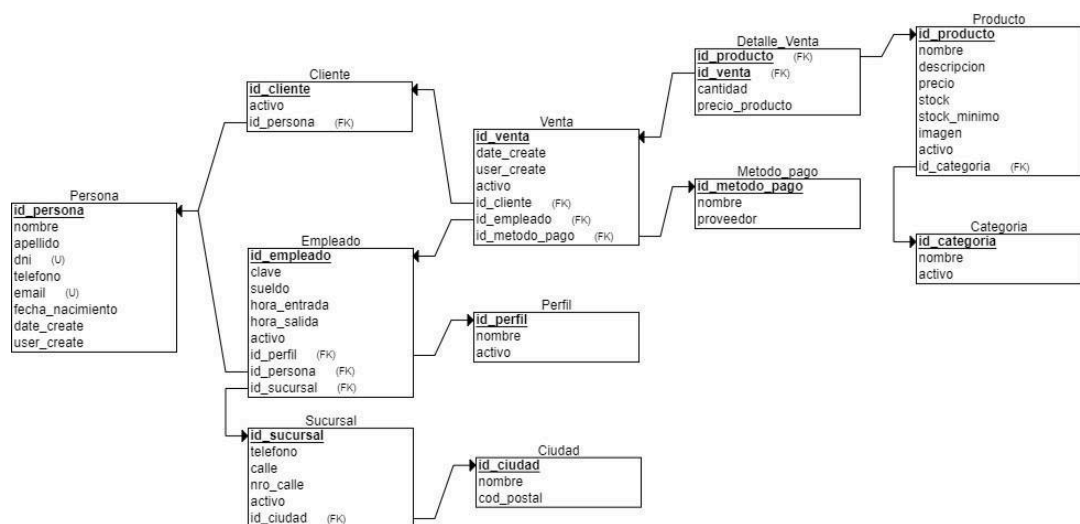
descripciones y las relaciones entre tablas), sirviendo como referencia técnica fundamental.

CAPÍTULO IV: DESARROLLO DEL TEMA

En este capítulo se presenta un análisis detallado y la estructuración de la información recopilada para el desarrollo de Gestor Franquicias. Para ello, se utilizaron herramientas de modelado de datos que permitieron visualizar las entidades, tablas y sus interconexiones de manera clara y precisa.

a) Diagrama de Modelo Relacional

Un Diagrama de Modelo Relacional, comúnmente conocido como Diagrama Entidad-Relación (DER), es una representación gráfica que ilustra la estructura de la base de datos. Este diagrama muestra cómo las entidades (tablas) se relacionan entre sí a través de llaves primarias y foráneas, proporcionando una visión clara del flujo de datos y la organización del sistema. El modelo siguiente representa el diseño de la base de datos para el sistema Gestor Franquicias.



b) Diccionario de Datos

El Diccionario de Datos es una herramienta esencial para la gestión de la información. Proporciona una descripción completa y estructurada de cada elemento de la base de datos. En él se detallan los nombres de las tablas, sus campos (atributos), los tipos de datos, longitudes, descripciones y las restricciones o relaciones que existen entre ellos. A continuación, se presenta el diccionario de datos correspondiente al modelo relacional de Gestor Franquicias, detallando la información de cada tabla.

Características de la tabla			
nombre	Perfil		
módulo	Perfil		
descripción	esta tabla se diseñó para almacenar la información de los perfiles		
Características de los datos			
campo	tipo	long	significado
id_perfil	int	4	identificación única para un perfil
nombre	varchar	100	indica el nombre del perfil
activo	bit	1	indica el estado del perfil (0-inactivo. 1-activo)
Restricciones			
campo	tipo restricción		
id_perfil	PRIMARY KEY		
activo	DEFAULT		
Claves Foráneas			
campo	entidad asociada		
-	-		

Características de la tabla			
nombre	Persona		
módulo	Persona		
descripción	esta tabla se diseñó para almacenar la información de las personas		
Características de los datos			
campo	tipo	long	significado
id_persona	int	4	identificación única para una persona
nombre	varchar	100	indica el nombre de la persona
apellido	varchar	100	indica el apellido de la persona
dni	numeric	8	indica el DNI de la persona
telefono	varchar	15	indica el teléfono de la persona
email	varchar	200	indica el email de la persona
fecha_nacimiento	date	-	indica la fecha de nacimiento de la persona

date_create	date		indica la fecha de creación del registro
user_create	int	4	indica el usuario de DB que creó el registro
Restricciones			
campo		tipo restricción	
id_persona		PRIMARY KEY	
email		UNIQUE	
dni		UNIQUE	
date_create		DEFAULT	
user_create		DEFAULT	
Claves Foráneas			
campo		entidad asociada	
-		-	

Características de la tabla			
nombre	Empleado		
módulo	Empleado		
descripción	esta tabla se diseñó para almacenar la información de los empleados		
Características de los datos			
campo	tipo	long	significado
id_empleado	int	4	identificación única para una persona
clave	varchar	200	indica la clave del empleado
sueldo	float	-	indica el sueldo del empleado
hora_entrada	time	-	indica el horario de entrada del empleado
hora_salida	time	-	indica el horario de salida del empleado
id_perfil	int	4	indica la identificación única del tipo de perfil del empleado
id_persona	int	4	indica la identificación única de la persona
id_sucursal	int	4	indica la identificación única de la sucursal a la que pertenece el empleado

activo	bit	1	indica el estado del empleado (0-inactivo. 1-activo)
Restricciones			
campo		tipo restricción	
id_empleado		PRIMARY KEY	
id_perfil		FOREIGN KEY	
id_persona		FOREIGN KEY	
id_sucursal		FOREIGN KEY	
activo		DEFAULT	
Claves Foráneas			
campo		entidad asociada	
id_perfil		Perfil	
id_persona		Persona	
id_sucursal		Sucursal	

Características de la tabla			
nombre		Cliente	
módulo		Cliente	
descripción		esta tabla se diseñó para almacenar la información de los clientes	
Características de los datos			
campo	tipo	long	significado
id_cliente	int	4	identificación única para un cliente
id_persona	int	4	indica la identificación única de la persona
activo	bit	1	indica el estado del cliente (0-inactivo. 1-activo)
Restricciones			
campo		tipo restricción	
id_cliente		PRIMARY KEY	
id_persona		FOREIGN KEY	
activo		DEFAULT	

Claves Foráneas	
campo	entidad asociada
id_persona	Persona

Características de la tabla			
nombre	Sucursal		
módulo	Sucursal		
descripción	esta tabla se diseñó para almacenar la información de las sucursales		
Características de los datos			
campo	tipo	long	significado
id_sucursal	int	4	identificación única para una sucursal
calle	varchar	100	indica el nombre de la calle de la sucursal
telefono	varchar	15	indica el numero de telefono de la sucursal
nro_calle	int	4	indica el número de calle de la sucursal
id_ciudad	int	4	indica la identificación única para una ciudad
activo	bit	1	indica el estado de la sucursal (0-inactivo. 1-activo)
Restricciones			
campo	tipo restricción		
id_sucursal	PRIMARY KEY		
id_ciudad	FOREIGN KEY		
activo	DEFAULT		
Claves Foráneas			
campo	entidad asociada		
id_ciudad	Ciudad		

Características de la tabla	
nombre	Ciudad
módulo	Ciudad

descripción		esta tabla se diseñó para almacenar la información de las ciudades	
Características de los datos			
campo	tipo	long	significado
id_ciudad	int	4	identificación única para una sucursal
nombre	varchar	100	indica el nombre de la ciudad
cod_postal	varchar	100	indica el código postal de la sucursal
Restricciones			
campo		tipo restricción	
id_ciudad		PRIMARY KEY	
Claves Foráneas			
campo		entidad asociada	
-		-	

Características de la tabla			
nombre	Venta		
módulo	Venta		
descripción	esta tabla se diseñó para almacenar la información de una venta		
Características de los datos			
campo	tipo	long	significado
id_venta	int	4	identificación única para una venta
date_create	date	-	indica la fecha y hora en el que el registro fue creado
user_create	varchar	100	indica el nombre del usuario que creó el registro
id_cliente	int	4	indica la identificación única para un cliente
id_empleado	int	4	indica la identificación única para un empleado
id_metodo_pago	int	4	indica la identificación única para un método de pago
activo	bit	1	indica el estado de la venta (0-inactivo. 1-activo)

Restricciones	
campo	tipo restricción
id_venta	PRIMARY KEY
date_create	DEFAULT
user_create	DEFAULT
id_cliente	FOREIGN KEY
id_empleado	FOREIGN KEY
id_metodo_pago	FOREIGN KEY
activo	DEFAULT
Claves Foráneas	
campo	entidad asociada
id_cliente	Cliente
id_empleado	Empleado
id_metodo_pago	Metodo_pago

Características de la tabla			
nombre	Detalle_Venta		
módulo	Detalle_Venta		
descripción	esta tabla se diseñó para almacenar la información del detalle de una venta		
Características de los datos			
campo	tipo	long	significado
id_producto	int	4	identificación única para un producto
id_venta	int	4	identificación única para una venta
cantidad	int	4	indica la cantidad de productos
precio_producto	float	-	indica el precio del producto
Restricciones			
campo	tipo restricción		
id_producto	PRIMARY KEY		
id_venta	PRIMARY KEY		
id_producto	FOREIGN KEY		
id_venta	FOREIGN KEY		

Claves Foráneas	
campo	entidad asociada
id_producto	Producto
id_venta	Venta

Características de la tabla			
nombre	Producto		
módulo	Producto		
descripción	esta tabla se diseñó para almacenar la información de los productos		
Características de los datos			
campo	tipo	long	significado
id_producto	int	4	identificación única para un producto
nombre	varchar	100	indica el nombre del producto
descripcion	varchar	200	indica la descripción del producto
precio	float	-	indica el precio del producto
stock	int	4	indica el stock actual del producto
stock_minimo	int	4	indica el stock de reposición del producto
imagen	varchar	200	indica la imagen del producto
activo	bit	1	indica el estado del producto (0-inactivo. 1-activo)
id_categoria	int	4	indica la identificación única de la categoría del producto
Restricciones			
campo	tipo restricción		
id_producto	PRIMARY KEY		
id_categoria	FOREIGN KEY		
activo	DEFAULT		
Claves Foráneas			
campo	entidad asociada		
id_categoria	Categoría		

Características de la tabla			
nombre		Categoria	
módulo		Categoria	
descripción		esta tabla se diseñó para almacenar la información de los productos	
Características de los datos			
campo	tipo	long	significado
id_categoria	int	4	identificación única para una categoría
nombre	varchar	100	indica el nombre de la categoría
activo	bit	1	indica el estado de la categoria (0-inactivo. 1-activo)
Restricciones			
campo		tipo restricción	
id_categoria		PRIMARY KEY	
activo		DEFAULT	
Claves Foráneas			
campo		entidad asociada	
-		-	

Características de la tabla			
nombre	Metodo_pago		
módulo	Metodo_pago		
descripción	esta tabla se diseñó para almacenar la información de los métodos de pagos		
Características de los datos			
campo	tipo	long	significado
id_metodo_pago	int	4	identificación única para un método de pago
nombre	varchar	100	indica el nombre del método de pago (efectivo,transferencia, crédito,etc)
proveedor	varchar	100	indica el nombre de la entidad que

			facilita el uso del método de pago
Restricciones			
campo		tipo restricción	
id_metodo_pago		PRIMARY KEY	
Claves Foráneas			
campo		entidad asociada	
-		-	

CAPÍTULO V: CONCLUSIONES

Tema 1: "Procedimientos y funciones de almacenado"

Aunque el INSERT directo puede ser más veloz en términos brutos, los procedimientos almacenados son ampliamente superiores en entornos reales debido a su seguridad, consistencia, mantenibilidad y control.

Tema 2: "Gestión de Transacciones y Transacciones Anidadas"

La implementación de transacciones y transacciones anidadas resultó ser una estrategia clave para asegurar la integridad y consistencia del sistema. Este mecanismo no solo protege la atomicidad de las operaciones críticas, como el registro de ventas y la actualización de stock, sino que también optimiza el manejo de errores, garantizando que los datos sean reversibles en caso de fallos.

Tema 3: "Optimización de Índices en SQL Server"

Esta serie de pruebas demostró que el rendimiento de una consulta no depende de si existe un índice, sino de si existe el diseño correcto del índice. Logramos transformar una consulta de 4.6 segundos en una de medio segundo y con una lectura lógica de 7833 a solo 78. Esto comprueba que una estrategia de indexación correcta, diseñada específicamente para las consultas que debe resolver, mejora notablemente el rendimiento de una base de datos.

Tema 4: "Vistas y Vistas Indexadas en SQL Server"

Las Vistas y Vistas Indexadas, son esenciales para la optimización de consultas y la garantía de un rendimiento óptimo. Estos elementos son cruciales para cumplir con los objetivos del sistema, como la provisión de informes gerenciales detallados y la optimización de los procesos de búsqueda y recuperación de datos.

BIBLIOGRAFÍA

Tema 1: "Procedimientos y funciones de almacenado"

<https://learn.microsoft.com/es-es/sql/relational-databases/stored-procedures/stored-procedures-database-engine?view=sql-server-ver16>

Tema 2: “Gestión de Transacciones y Transacciones Anidadas”

<https://learn.microsoft.com/es-es/office/client-developer/access/desktop-database-reference/transaction-processing>

<https://learn.microsoft.com/es-es/sql/t-sql/language-elements/transactions-transact-sql?view=sql-server-ver16>

https://programacion.net/articulo/transacciones_en_sql_server_299

<https://www.sqlshack.com/transactions-in-sql-server-for-beginners/>

Tema 3: “Optimización de Índices en SQL Server”

<https://learn.microsoft.com/es-es/sql/relational-databases/indexes/clustered-and-non-clustered-indexes-described?view=sql-server-ver17>

Tema 4: "Vistas y Vistas Indexadas en SQL Server"

<https://learn.microsoft.com/en-us/sql/t-sql/statements/create-view-transact-sql?view=sql-server-ver17>

<https://learn.microsoft.com/es-es/sql/relational-databases/views/create-indexed-views?view=sql-server-ver16>