



UNIVERSIDAD POLITÉCNICA DE VICTORIA
INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN

Base de datos: Panadería

Integrantes:

José Natanahel Medina Santillan

Cristian Eduardo Rodriguez Balboa

Ignacio Daniel Alvarez Rios

Carlos Miguel Rosas Ortíz

Profesora: Eréndira Gutiérrez Meza

Asignatura: Inteligencia de negocios

Grupo: ITI-07477

Lugar: Ciudad Victoria, Tamaulipas

Diciembre, 2020

Índice

| | |
|--|-----------|
| 1. Planteamiento del Problema | 5 |
| 2. Justificación | 5 |
| 3. Objetivos generales | 6 |
| 4. Objetivos específicos | 6 |
| 5. Diagrama Entidad-Relación | 7 |
| 6. Diccionario de datos de panadería | 8 |
| 6.1. Tabla Proveedores | 8 |
| 6.2. Tabla Inventario | 8 |
| 6.3. Tabla ProInventario | 8 |
| 6.4. Tabla Recetas | 9 |
| 6.5. Tabla Producción | 9 |
| 6.6. Tabla DetalleEncargo | 10 |
| 6.7. Tabla Encargos | 10 |
| 6.8. Tabla Empleados | 11 |
| 6.9. Tabla Usuarios | 11 |
| 6.10. Tabla Cargos | 11 |
| 6.11. Tabla Clientes | 12 |
| 6.12. Tabla Ventas | 12 |
| 6.13. Tabla DetalleVentas | 12 |
| 6.14. Tabla Productos | 13 |
| 7. Implementación | 14 |
| 7.1. Carga, limpieza y transformación de datos | 14 |
| 7.1.1. ¿Qué son los procesos ETL? | 15 |
| 7.1.2. ¿Qué es un Data Warehouse? | 16 |
| 7.2. Implementación de funciones | 17 |
| 7.3. Transformación de la columna Salario de la tabla Cargos | 17 |
| 7.4. Modificación de la columna Teléfono de la tabla Clientes | 19 |
| 7.5. Modificación de las columnas Fecha_encargo y Fecha_entrega de la tabla DetalleEncargo | 22 |
| 8. Data Warehouse | 23 |
| 9. Diccionario de datos de Data Warehouse panadería | 25 |
| 9.1. Tabla Dim_Categorias | 25 |
| 9.2. Tabla Dim_Clientes | 25 |
| 9.3. Tabla Dim_Empleados | 25 |
| 9.4. Tabla Hechos_Ventas | 26 |
| 9.5. Tabla Dim_Productos | 26 |

| | |
|--|-----------|
| 9.6. Tabla Dim_Tiempo | 27 |
| 10.Llenado de las tablas del Data Warehouse | 28 |
| 11.Creación del cubo OLAP | 29 |
| 12.Creación de informes | 31 |
| 12.1. Informe de Ventas | 31 |
| 12.2. Informe de Categorías | 32 |
| 12.3. Informe de Empleados | 33 |
| 12.4. Informe de Clientes | 34 |
| 13.Conclusión | 35 |
| 14.Referencias | 36 |

Índice de tablas

| | | |
|-----|----------------------------------|----|
| 1. | Tabla proveedores | 8 |
| 2. | Tabla inventario | 8 |
| 3. | Tabla ProInventario | 8 |
| 4. | Tabla recetas | 9 |
| 5. | Tabla producción | 9 |
| 6. | Tabla DetalleEncargo | 10 |
| 7. | Tabla encargos | 10 |
| 8. | Tabla empleados | 11 |
| 9. | Tabla usuarios | 11 |
| 10. | Tabla cargos | 11 |
| 11. | Tabla clientes | 12 |
| 12. | Tabla ventas | 12 |
| 13. | Tabla detalleventas | 12 |
| 14. | Tabla productos | 13 |
| 15. | Tabla Dim_Categoria | 25 |
| 16. | Tabla Dim_Clientes | 25 |
| 17. | Tabla Dim_Empleados | 25 |
| 18. | Tabla Hechos_Ventas | 26 |
| 19. | Tabla Dim_Productos | 26 |
| 20. | Tabla Dimension_tiempo | 27 |

Índice de figuras

| | | |
|-----|--|----|
| 1. | Diagrama entidad-relación | 7 |
| 2. | Ejemplo de ETL | 15 |
| 3. | Tabla Cargos | 17 |
| 4. | Tabla Cargos modificada | 17 |
| 5. | Tabla Cargos restaurada | 18 |
| 6. | Columna Teléfono Cambiada a tipo texto | 19 |
| 7. | Columna Telefono dividida | 20 |
| 8. | Columna Telefono combinada | 21 |
| 9. | Columnas Fecha_encargo y Fecha_entrega modificadas | 22 |
| 10. | Esquema copo de nieve del Data WareHouse | 24 |
| 11. | Ejemplo al implementar un Cubo Olap | 30 |
| 12. | Informe de Ventas | 31 |
| 13. | Informe de Categorías | 32 |
| 14. | Informe de Empleados | 33 |
| 15. | Informe de Clientes | 34 |

1. Planteamiento del Problema

Con el paso del tiempo las necesidades cambian y crecen constantemente, para poder adaptarse a los nuevos tiempos tecnológicos se debe realizar una aceptación y aplicación de la tecnología en nuestras actividades diarias con lo que estando en un ambiente laboral y una buena administración de los datos se vuelve algo interesante y necesario para la buena de administración de un negocio.

Podemos observar que surgen problemas desde los negocios pequeños hasta a los de mayor magnitud. El problema principal es para los negocios pequeños que son más cerrados a las nuevas tecnologías que van apareciendo, analizaremos un negocio pequeño que será una panadería demostrando que incluso este negocio tiene la gran necesidad de administrar sus datos de forma correcta. Para así optimizar procesos como lo son producción, ventas, control de empleados y así optimizar tiempo.

El problema surge a partir de que muchas panaderías que son de empresas pequeñas que no cuentan con un sistema o una base de datos donde tengan una buena administración de los datos, procesos de venta, inventariado, control de empleados, etc. La solución a este problema es la implementación de inteligencia de negocios, de esta forma se tendrá un mejor manejo de los datos y procesos. Se podrán visualizar los datos que el usuario necesite, ya sea para mejorar su producción, ver un historial de ventas, ver la información de sus empleados y sus clientes. Teniendo estos datos incluso se podrá tener en cuenta si la empresa se puede expandir.

2. Justificación

Realizando un análisis podemos decir que la necesidad del almacenamiento y administración de los datos es algo importante que debe tomarse en cuenta con gran prioridad, para tener un crecimiento constante y ser un competidor capaz de poder evolucionar con el tiempo y aplicar las nuevas tecnologías necesarias que vayan apareciendo.

Principalmente el crecimiento de las cantidades de los datos es el problema más notorio ya que constantemente habrá ventas, registros de productos, etc. Y en base a todos estos datos existen posibilidades de obtener datos estadísticos que nos proporcionarán información para realizar análisis precisos y tener una base para la toma de decisiones en el negocio.

Se realiza la siguiente investigación con el fin de implementar la inteligencia de negocios en una empresa y proveer una buena administración para lograr facilitar la recolección, el almacenamiento y procesamiento de datos generando la formación de los fundamentos para la toma de decisiones estratégicas las cuales serán la guía perfecta para el éxito y una gran prosperidad gracias a la implementación de esta.

3. Objetivos generales

Implementar una base de datos que permitirá la aplicación de la inteligencia de negocios que permita a la empresa, acceder a los datos, optimizando el tiempo en la obtención de los datos y mejorar el análisis de la información para el área de ventas, control de empleados e información de los clientes, sirviendo como soporte para la toma de decisiones de manera oportuna a las necesidades del negocio.

4. Objetivos específicos

- Analizar el desarrollo de las ventas de los productos para generar un informe de resultados.
- Identificar a los empleados para realizar análisis de su rendimiento e información.
- Identificar cuales son los productos mas comprados por algún cliente determinado.

5. Diagrama Entidad-Relación

A continuación, se muestra el diagrama Entidad-Relación realizado para la base de datos de la panadería en la figura 1.

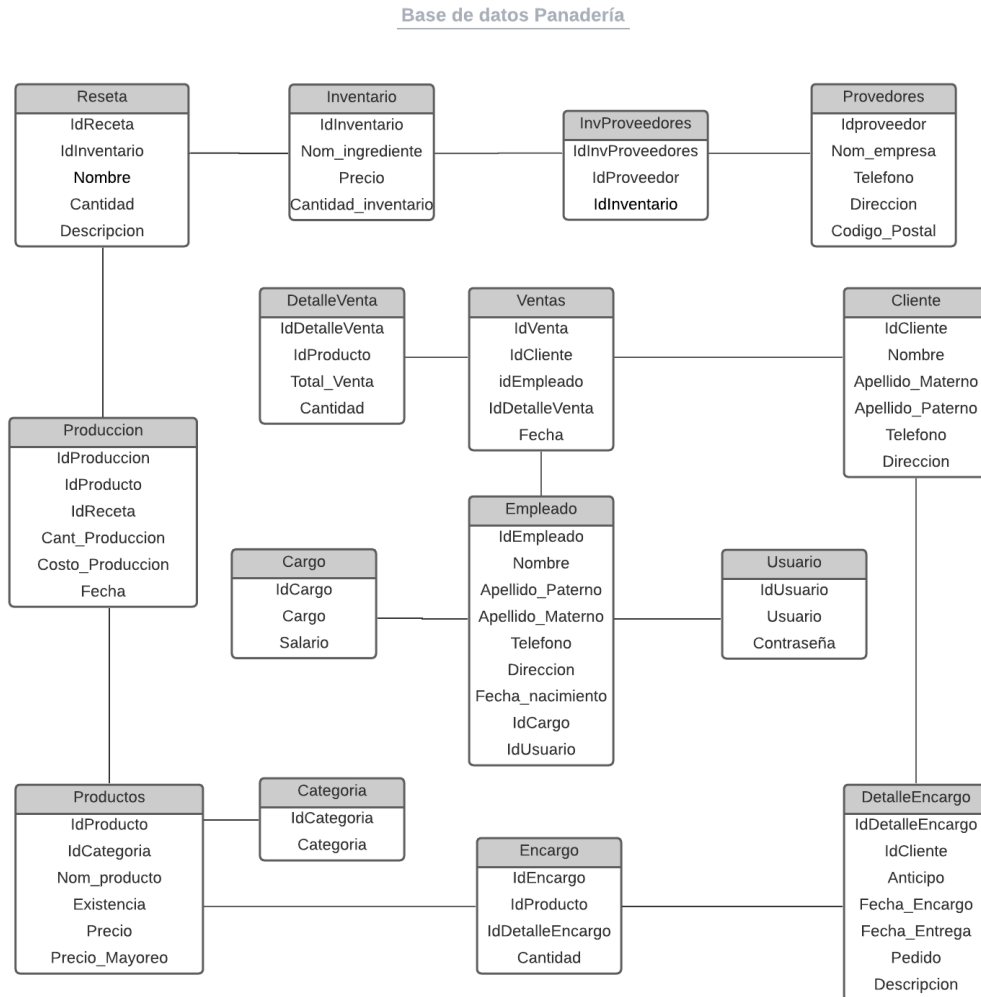


Figura 1: Diagrama entidad-relación de la base de datos panadería

6. Diccionario de datos de panadería

6.1. Tabla Proveedores

La tabla Proveedores almacenara toda la información que se tenga referente a los proveedores con los que cuente la panadería. Como se muestra en la tabla 1.

| Proveedores | | | |
|---------------|---------|--------------|--|
| Campo | PK o FK | Tipo de dato | Descripción |
| IdProveedor | PK | INT | Almacena el identificador de cada fila de la tabla |
| Nom_empresa | | VARCHAR(50) | Almacena el nombre de la empresa del proveedor |
| Teléfono | | VARCHAR(15) | Almacena el número de contacto del proveedor |
| Dirección | | VARCHAR(50) | Almacena la dirección del proveedor |
| Código_Postal | | VARCHAR(10) | Almacena el código postal |

Tabla 1: Tabla proveedores

6.2. Tabla Inventario

La tabla Inventario almacenará la información sobre los productos con los que se cuenten para realizar el producto. Como se muestra en la tabla 2.

| Inventario | | | |
|-----------------|---------|--------------|--|
| Campo | PK o FK | Tipo de dato | Descripción |
| IdInventario | PK | INT | Identificador de cada fila de la tabla inventario |
| Nom_ingrediente | | VARCHAR(50) | Almacena el nombre de los ingredientes |
| Precio | | DECIMAL(5,2) | Almacena el precio de compra de los ingredientes |
| Cantidad | | DECIMAL(7,2) | Almacena la cantidad de ingredientes |
| Unidad | | VARCHAR(25) | Almacena el contenido de los ingredientes (Kg, ml) |

Tabla 2: Tabla inventario

6.3. Tabla ProInventario

La tabla ProInventario sirve como intermediario para las tablas Proveedores e Inventario, ya que entre estas tablas se generaba una relación de muchos a muchos. Para resolver este problema se decidió crear la tabla ProInventario. Como se muestra en la tabla 3.

| ProInventario | | | |
|---------------|---------|--------------|---------------------------------------|
| Campo | PK o FK | Tipo de dato | Descripción |
| IdProveedor | PK | INT | Identificador de la tabla proveedores |
| IdInventario | PK | INT | Identificador de la tabla inventario |

Tabla 3: Tabla ProInventario

6.4. Tabla Recetas

La tabla Recetas almacena las recetas de los productos. Como se muestra en la tabla 4.

| Recetas | | | |
|--------------|---------|--------------|---|
| Campo | PK o FK | Tipo de dato | Descripción |
| IdReceta | PK | SMALLINT | Llave primaria de la tabla Recetas |
| IdInventario | FK | SMALLINT | Llave foránea de la tabla Inventario |
| Nombre | | VARCHAR(50) | Almacena el nombre de la receta |
| Cantidad | | SMALLINT | Almacena la cantidad que haya de un producto |
| Descripción | | VARCHAR(100) | Guarda los ingredientes y la preparación necesaria para la receta |

Tabla 4: Tabla recetas

6.5. Tabla Producción

La tabla Producción contendrá la información referente a la producción de pan que se haya tenido en cierta fecha, teniendo como campos la cantidad de la producción, el costo de la producción, así como la fecha en la que se realizó. Como se muestra en la tabla 5.

| Producción | | | |
|------------------|---------|--------------|--|
| Campo | PK o FK | Tipo de dato | Descripción |
| IdProduccion | PK | INT | Identificador de cada fila de la tabla producción |
| IdProducto | FK | INT | Identificador de llave foránea de la tabla productos |
| Cant_Produccion | | DECIMAL(5,3) | Almacena la cantidad de producto realizado |
| Costo_Produccion | | DECIMAL(5,3) | Almacena el costo que tuvo la elaboración de cierto producto |
| Fecha | | DATE | Almacena la fecha en que se realizó cierto producto |

Tabla 5: Tabla producción

6.6. Tabla DetalleEncargo

La tabla DetalleEncargo almacenara los detalles que se hagan de los encargos. Como se muestra en la tabla 6.

| DetalleEncargo | | | |
|----------------|---------|--------------|---|
| Campo | PK o FK | Tipo de Dato | Descripción |
| IdEncargo | FK | INT(5) | Llave Foránea |
| FechaEncargo | | DATE | Fecha de inicio |
| FechaEntrega | | DATE | Fecha de Fin |
| IdCliente | FK | INT(5) | Llave Foránea |
| Anticipo | | VARCHAR(50) | Anticipo de los clientes |
| Pedido | | VARCHAR(50) | Los pedidos de los clientes |
| Descripción | | VARCHAR(100) | Descripción de algún detalle o problema |

Tabla 6: Tabla DetalleEncargo

6.7. Tabla Encargos

La tabla Encargo llevara el control de los encargos que se hagan a la panadería, almacenando sus identificadores y la cantidad del producto que se pida. Como se muestra en la tabla 7.

| Encargo | | | |
|------------|---------|--------------|-----------------------|
| Campo | PK o FK | Tipo de Dato | Descripción |
| IdEncargo | FK | INT(5) | Llave Foránea |
| IdProducto | FK | INT(5) | Llave Foránea |
| Cantidad | | INT(50) | Cantidad de productos |

Tabla 7: Tabla encargos

6.8. Tabla Empleados

La tabla Empleados contendrá la toda la información de los empleados con los que cuenta la panadería. Como se muestra en la tabla 8.

| Empleado | | | |
|-------------------|---------|--------------|------------------------------------|
| Campo | PK o FK | Tipo de Dato | Descripción |
| IdEmpleado | PK | INT(5) | Llave Primaria |
| Nombre | | VARCHAR(50) | nombres de los empleados |
| ApellidoPaterno | | VARCHAR(50) | Apellido paternos de los empleados |
| ApellidoMaterno | | VARCHAR(50) | Apellido materno de los empleados |
| Telefono | | VARCHAR(12) | Números telefónicos |
| Direccion | | VARCHAR(50) | Ubicación de vivienda |
| FechaDeNacimiento | | DATE | Fecha en que nació |
| IdEncargo | FK | INT(5) | Llave Foránea |
| IdUsuario | FK | INT(5) | Llave Foránea |

Tabla 8: Tabla empleados

6.9. Tabla Usuarios

La tabla Usuarios almacenara los usuarios de los empleados para ingresar al sistema. Como se muestra en la tabla 9.

| Usuario | | | |
|-----------|---------|--------------|-------------------|
| Campo | PK o FK | Tipo de Dato | Descripción |
| IdUsuario | PK | INT(5) | Llave Primaria |
| Usuario | | VARCHAR(50) | Nombre de usuario |
| Password | | VARCHAR(50) | Contraseña |

Tabla 9: Tabla usuarios

6.10. Tabla Cargos

La tabla Cargos contendrá todos los cargos o funciones que haya en la panadería. Como se muestra en la tabla 10.

| Cargo | | | |
|---------|---------|--------------|--------------------------|
| Campo | PK o FK | Tipo de Dato | Descripción |
| IdCargo | PK | INT(5) | Llave Primaria |
| Cargo | | VARCHAR(25) | Cargo a ejercer |
| Salario | | INT(10) | Salario que se le pagará |

Tabla 10: Tabla cargos

6.11. Tabla Clientes

La tabla Clientes almacenara la información de los clientes. Como se muestra en la tabla 11.

| Cliente | | | |
|-----------------|---------|--------------|--|
| Campo | PK o FK | Tipo de Dato | Descripción |
| IdCliente | PK | INT(5) | Llave Primaria |
| Nombre | | VARCHAR(50) | Nombre del cliente |
| ApellidoPaterno | | VARCHAR(50) | Apellido paterno del cliente |
| ApellidoMaterno | | VARCHAR(50) | Apellido materno del cliente |
| Telefono | | VARCHAR(15) | Numero de teléfono o celular del cliente |
| Direccion | | VARCHAR(50) | Dirección del cliente |

Tabla 11: Tabla clientes

6.12. Tabla Ventas

La tabla Ventas llevara el control de las ventas, almacenando los identificadores de Clientes, Empleados y la fecha en que se realizó la venta. Como se muestra en la tabla 12.

| Ventas | | | |
|------------|---------|--------------|---|
| Campo | PK o FK | Tipo de Dato | Descripción |
| IdVenta | PK | INT(5) | Llave Primaria |
| IdCliente | FK | INT(5) | Identificador primario de la tabla cliente |
| IdEmpleado | FK | INT(5) | Identificador primario de la tabla Empleado |
| Fecha | | DATE | Fecha de la venta |

Tabla 12: Tabla ventas

6.13. Tabla DetalleVentas

La tabla DetalleVentas almacenara todos los detalles de las ventas realizadas, almacenando los identificadores de Ventas y Productos, además, almacenara el total de la venta y la cantidad de producto que se vendió. Como se muestra en la tabla 13.

| DetalleVentas | | | |
|---------------|---------|--------------|--|
| Campo | PK o FK | Tipo de Dato | Descripción |
| IdVenta | PK | INT(5) | Llave Primaria |
| IdProducto | FK | INT(5) | Identificador primario de la tabla Productos |
| TotalVenta | | FLOAT | Total a pagar de la venta |
| Cantidad | | INT(5) | Cantidad de producto |

Tabla 13: Tabla detalleventas

6.14. Tabla Productos

La tabla Productos almacenará la información de los productos que venda la panadería, así como llevar un control de la existencia del producto, el precio de venta del producto y el precio de venta al mayoreo de cada producto. Como se muestra en la tabla 14.

| Productos | | | |
|---------------|---------|--------------|------------------------------------|
| Campo | PK o FK | Tipo de Dato | Descripción |
| IdProducto | PK | INT(5) | Identificador primario de la tabla |
| Nom_producto | | VARCHAR(50) | Nombre del producto |
| Existencia | | INT(5) | Cantidad de producto en existencia |
| Precio | | FLOAT | Precio del producto |
| precioMayoreo | | FLOAT | Precio del producto al mayoreo |

Tabla 14: Tabla productos

7. Implementación

7.1. Carga, limpieza y transformación de datos

Se realizó el siguiente análisis a la base de datos, con el fin de implementar la inteligencia de negocios en una empresa y proveer una buena administración para lograr facilitar la recolección, el almacenamiento y procesamiento de datos generando la formación de los fundamentos para la toma de decisiones estratégicas, las cuales serán la guía perfecta para el éxito y una gran prosperidad gracias a la implementación de esta.

Junto a la implementación de la inteligencia de negocios va integrado algo muy importante, que es la carga y transformación de datos la cual es fundamental para tener los datos impecables sin detalles que perjudiquen su manejo, para esto la carga de datos puede ser realizada en Power BI con una conexión a SQL Server donde los datos son transformados para aplicar una excelente limpieza y transformación.

Anteriormente se ha estado mencionando sobre la carga y transformación de datos, estos procesos son los más importantes en la elaboración de un Data Warehouse, los procesos ETL son muy útiles y beneficiosos para las organizaciones por su capacidad para integrar grandes bases de datos, logrando así una visión única global que permite, a los analistas y directivos, tomar las decisiones estratégicas adecuadas.

7.1.1. ¿Qué son los procesos ETL?

Los procesos ETL son un término estándar que se utiliza para referirse al movimiento y transformación de datos. Se trata del proceso que permite a las organizaciones mover datos desde múltiples fuentes, reformatearlos y cargarlos en otra base de datos (denominada Data Mart o Data Warehouse) con el objeto de analizarlos. También pueden ser enviados a otro sistema operacional para apoyar un proceso de negocio.

Los procesos ETL son una parte de la integración de datos y significa Extraer, Transformar y Cargar. Es un proceso en el que una herramienta ETL extrae los datos de varios sistemas de origen de datos, los transforma en el área de preparación y, finalmente, los carga en el sistema de almacenamiento de datos. Es un elemento importante cuya función completa el resultado de todo el desarrollo de la cohesión de aplicaciones y sistemas. A continuación, se muestra un ejemplo general del proceso ETL en la figura 2.

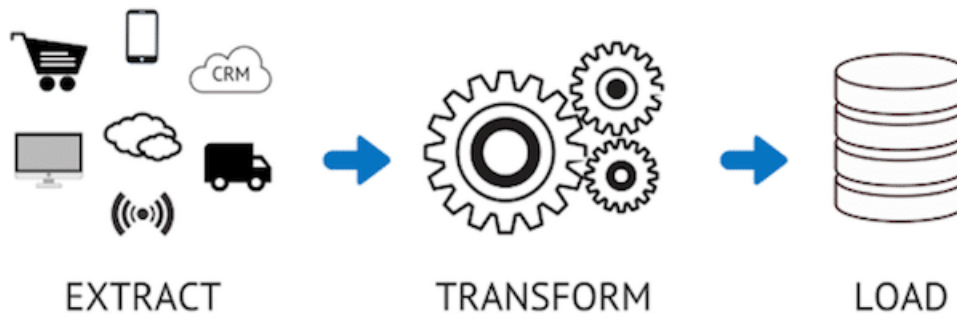


Figura 2: Ejemplo de ETL

7.1.2. ¿Qué es un Data Warehouse?

En la actualidad las empresas han producido una inmensa cantidad de datos las cuales provocaron el surgimiento de una forma para el manejo de estos. El conjuntos de datos o combinaciones de conjuntos de datos cuyo tamaño, complejidad y velocidad de crecimiento dificultan su captura, gestión, procesamiento o análisis mediante tecnologías y herramientas convencionales por lo que a su vez van surgiendo herramientas para ofrecer una mayor facilidad para el manejo de los dato, una de estas herramientas es nombrada como Data Warehouse nacida en 1998 gracias a los investigadores de IBM, Barry Devlin y Paul Murphy los cuales hicieron un gran esfuerzo y trabajaron con gran dedicación creando este gran concepto.

El concepto de Data Warehouse dice que es una base de datos corporativa que se caracteriza por integrar y depurar información de una o más fuentes distintas, para luego procesarla permitiendo su análisis desde infinidad de perspectivas y con grandes velocidades de respuesta, realizando un análisis sobre este concepto podemos decir que nos provee de grandes ventajas para la toma de decisiones basándose en la información integrada y global de la empresa.

En los últimos años, el Data Warehouse ha evolucionado debido a información contextual que ahora se puede adjuntar a los datos no estructurados y que también puede ser almacenada ofreciendo más flexibilidad, hay que tomar en cuenta que el Data Warehouse existe solo para ser leído no editado por lo que no está permitido la edición de los datos por lo que se mantendrán íntegros.

Realizando un análisis profundo a la base de datos del proyecto se nota de una forma muy clara la necesidad de aplicar funciones que son necesarias para obtener datos estadísticos, realizar una relación, entre otras más aplicaciones que se les pueden dar. Por ejemplo: En las ventas se pueden aplicar funciones de tipo fecha para realizar un análisis para ver qué ventas se han realizado entre cierto rango de fechas.

7.2. Implementación de funciones

Para la implementación de las funciones se utilizó el programa Power BI y el motor de base de datos SQL Server 2014 Management Studio, Power BI es un servicio de análisis empresarial de Microsoft, cuyo objetivo es proporcionar visualizaciones interactivas y capacidades de inteligencia de negocios, SQL Server 2014 Management Studio es un sistema de gestión de base de datos relacional desarrollado como un producto de software con la función principal de almacenar y recuperar datos según lo solicitado por otras aplicaciones, el lenguaje M es el lenguaje que utiliza Power Query para poder conectar, combinar y refinar los orígenes de datos y así dejar una base de datos limpia que permita extraer y analizar la información que se desee con ayuda de estos softwares se logró la implementación de las siguientes funciones.

7.3. Transformación de la columna Salario de la tabla Cargos

Para la transformación de la columna salario se utilizó una función de tabla llamada Table.TransformColumnTypes esta función devuelve una tabla de la entrada **table** aplicando la operación de transformación a las columnas especificadas en el parámetro typeTransformations (donde el formato es nombre de columna, nombre de tipo). la tabla antes de modificar se muestra en la figura 3

Syntax : Table.TransformColumnTypes(table as table, typeTransformations as list)

| 1.2 IdCargo | A _C Cargo | \$ Salario | Empleados |
|-------------|----------------------|------------|-----------|
| 1 | 1 cajero | 1,408.04 | Table |
| 2 | 2 repartidor | 1,919.10 | Table |
| 3 | 3 contador | 1,106.12 | Table |
| 4 | 4 panadero | 1,392.38 | Table |
| 5 | 5 pastelero | 1,051.71 | Table |
| 6 | 6 empleado general | 1,870.60 | Table |

Figura 3: Tabla Cargos

al aplicar la función los datos se transforman y se muestran como en la figura 4.

Función : Table.TransformColumnTypes(dbo_Cargos,"Salario", Int64.Type)

Resultado

| 1.2 IdCargo | A _C Cargo | 1 ² Salario | Empleados |
|-------------|----------------------|------------------------|-----------|
| 1 | 1 cajero | 1408 | Table |
| 2 | 2 repartidor | 1919 | Table |
| 3 | 3 contador | 1106 | Table |
| 4 | 4 panadero | 1392 | Table |
| 5 | 5 pastelero | 1052 | Table |
| 6 | 6 empleado general | 1871 | Table |

Figura 4: Tabla Cargos modificada

Se acordó poner la columna de salario en entero para un mejor control de los pagos a los empleados.

Corrección en la conversión del tipo de dato

En la columna salario de la tabla Cargos, anteriormente se había convertido a tipo entero la columna Salario, pero al convertirla se redondearon los datos, esto puede generar pérdidas para la empresa, finalmente se decidió convertir de nuevo la columna Salario a tipo de dato Decimal, como estaba originalmente.

para esto se usó la funcion TransformColumnTypes, el resultado se muestra en la figura 5

Función : Table.TransformColumnTypes(dbo_Cargos,{{"Salario", type number}})

Resultado

| 1.2 IdCargo | A _C Cargo | 1.2 Salario | Empleados |
|-------------|----------------------|-------------|-----------|
| 1 | 1 cajero | 1408.04 | Table |
| 2 | 2 repartidor | 1919.1 | Table |
| 3 | 3 contador | 1106.12 | Table |
| 4 | 4 panadero | 1392.38 | Table |
| 5 | 5 pastelero | 1051.71 | Table |
| 6 | 6 empleado general | 1870.6 | Table |

Figura 5: Tabla Cargos restaurada

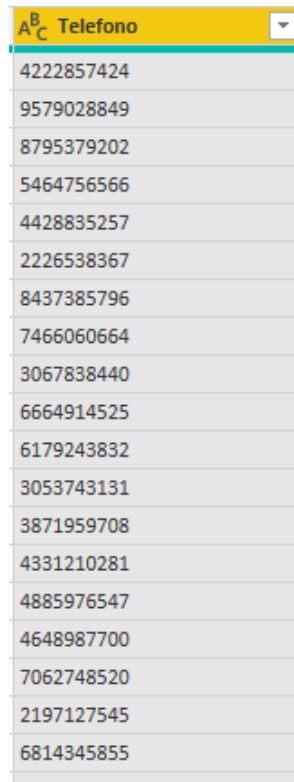
7.4. Modificación de la columna Teléfono de la tabla Clientes

En la carga de datos se introdujo como tipo de dato entero en la columna Teléfono de la tabla Clientes. Esta modificación se realizó porque por regla los números telefónicos se ponen como tipo de dato char, ya que pueden llevar guiones o paréntesis. Además de que un tipo de dato char tiene menor peso que un tipo de dato entero.

Función `Table.TransformColumnTypes`

Se utilizó esta función para transformarla a tipo de dato texto la columna Teléfono de la tabla Clientes ya que era tipo de dato entero. En la figura 6 se muestra la columna transformada a texto

Resultado



| A ^B A ^C Telefono |
|---|
| 4222857424 |
| 9579028849 |
| 8795379202 |
| 5464756566 |
| 4428835257 |
| 2226538367 |
| 8437385796 |
| 7466060664 |
| 3067838440 |
| 6664914525 |
| 6179243832 |
| 3053743131 |
| 3871959708 |
| 4331210281 |
| 4885976547 |
| 4648987700 |
| 7062748520 |
| 2197127545 |
| 6814345855 |

Figura 6: Columna Teléfono Cambiada a tipo texto

Función `Splitter.SplitTextByPositions` y función `Table.SplitColumn`

Después de haber cambiado la columna de tipo de dato entero a char, se procedió a dividir la columna, en el siguiente paso se utilizarán dos funciones en una. al aplicar las funciones en la columna, los datos se muestran de la figura 7

```
Table.SplitColumn(dbo_Clientes, "Telefono", Splitter.SplitTextByPositions({0, 3, 6}), {"Telefono.1", "Telefono.2", "Telefono.3"})
```

Resultado

| A _C ^B Telefono.1 | A _C ^B Telefono.2 | A _C ^B Telefono.3 |
|--|--|--|
| 422 | 285 | 7424 |
| 957 | 902 | 8849 |
| 879 | 537 | 9202 |
| 546 | 475 | 6566 |
| 442 | 883 | 5257 |
| 222 | 653 | 8367 |
| 843 | 738 | 5796 |
| 746 | 606 | 0664 |
| 306 | 783 | 8440 |
| 666 | 491 | 4525 |
| 617 | 924 | 3832 |
| 305 | 374 | 3131 |
| 387 | 195 | 9708 |
| 433 | 121 | 0281 |
| 488 | 597 | 6547 |
| 464 | 898 | 7700 |
| 706 | 274 | 8520 |
| 219 | 712 | 7545 |
| 681 | 434 | 5855 |

Figura 7: Columna Telefono dividida

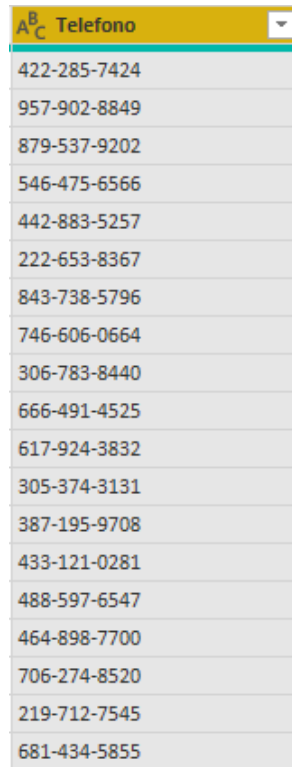
La función `Table.SplitColumn` para dividir la columna y la función `Splitter.SplitTextByPositions` ayuda a separar los datos entre las columnas, la primera tendrá 3, la segunda 3 y la ultima 4.

Función `CombineColumns` y función `Combiner.CombineTextByDelimiter`

Una vez separadas las funciones se prosigue a juntar las columnas en una sola y poner un delimitador entre los datos de cada una de las columnas en este caso será un "-". En la figura 8 se muestra la columna con los datos combinados

`Table.CombineColumns(dbo_Clientes, {"Telefono.1", "Telefono.2", "Telefono.3"}, Combiner.CombineTextByDelimiter("-", QuoteStyle.None), "Telefono")`

Resultado



| Telefono |
|--------------|
| 422-285-7424 |
| 957-902-8849 |
| 879-537-9202 |
| 546-475-6566 |
| 442-883-5257 |
| 222-653-8367 |
| 843-738-5796 |
| 746-606-0664 |
| 306-783-8440 |
| 666-491-4525 |
| 617-924-3832 |
| 305-374-3131 |
| 387-195-9708 |
| 433-121-0281 |
| 488-597-6547 |
| 464-898-7700 |
| 706-274-8520 |
| 219-712-7545 |
| 681-434-5855 |

Figura 8: Columna Telefono combinada

La función `Table.CombineColumns` para unir las columnas y por último `Combiner.CombineTextByDelimiter` que pondrá un separador entre los datos de las columnas combinadas.

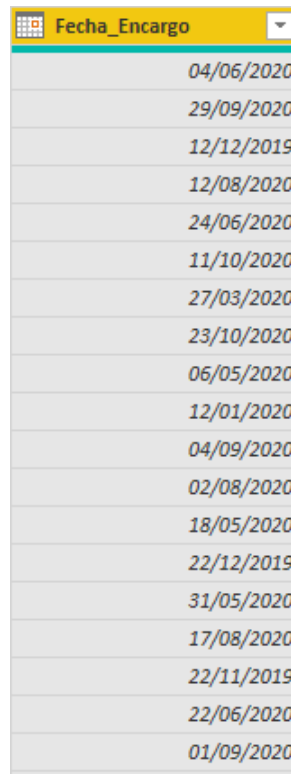
estos mismos procedimientos se aplican en la tabla Empleados, Proveedores.

7.5. Modificación de las columnas Fecha_encargo y Fecha_entrega de la tabla DetalleEncargo

Al revisar la tabla de DetalleEncargo se encontró que la fecha es tomada como tipo de dato texto. En la figura 9 se muestra la columna Fecha con los datos convertidos

```
Table.TransformColumnTypes(dbo_DetalleEncargo,{{"Fecha_Encargo", type date}})
```

Resultado



| Fecha_Encargo |
|---------------|
| 04/06/2020 |
| 29/09/2020 |
| 12/12/2019 |
| 12/08/2020 |
| 24/06/2020 |
| 11/10/2020 |
| 27/03/2020 |
| 23/10/2020 |
| 06/05/2020 |
| 12/01/2020 |
| 04/09/2020 |
| 02/08/2020 |
| 18/05/2020 |
| 22/12/2019 |
| 31/05/2020 |
| 17/08/2020 |
| 22/11/2019 |
| 22/06/2020 |
| 01/09/2020 |

Figura 9: Columnas Fecha_encargo y Fecha_entrega modificadas

Estos mismos procedimientos se aplicaron en las tablas Empleados, Producción, Ventas.

8. Data Warehouse

Teniendo conocimiento de todas las posibilidades y facilidades que nos brinda posteriormente se realizaron investigaciones y en consecuencia encontramos que existían 3 modelos de Data Warehouse, los cuales tienen sus propias características, ventajas y desventajas, se eligió el modelo copo de nieve que es el indicado para nuestro modelo, tomando en cuenta la robustez de la base de datos y realizando análisis previos se observó una dependencia para complementar los datos de algunas tablas. Posteriormente una vez realizado nuestro modelo claramente se nota que está compuesta de dimensiones las cuales están unidas a la tabla de hechos y estas dimensiones en conjunto forman enlaces interesantes que pueden brindar una facilidad a la hora de realizar análisis y obtener datos estadísticos enfocándonos a una o más dimensiones que componen el diagrama.

Observando de una forma detallada da múltiples posibilidades, en este caso la inclinación fue hacia la tabla ventas la cual como su nombre lo dice es la encargada de realizar las uniones para obtener todo lo relacionado a cada venta que es realizada, tomando en cuenta esto se usa como tabla de hechos y las dimensiones serían todas las tablas que complementan su información. Por ejemplo, si queremos saber en qué mes compra un cliente algún producto con mayor frecuencia o alguna posibilidad de que algún vendedor sea preferido viendo las ventas que hace al día. Entonces podemos decir con gran certeza que además de facilitar el análisis de datos también da muchas posibilidades y gran variedad con los datos para obtener una gran cantidad de resultados que pueden formar parte de un gran papel en un negocio.

El modelo copo de nieve nos provee de muchas ventajas entre ellas principalmente que realiza una optimización de algunas herramientas de modelado de base de datos multidimensionales OLAP, también podríamos agregar que la normalización de los atributos reduce el almacenamiento de datos. Otras de las cosas más que aporta el modelo copo de nieve es que no hay redundancia lo cual es algo importante ya que existen reglas en las bases de datos las cuales son muy claras en esta parte de tener datos repetidos o en otras palabras es una mala práctica, también a su vez es fácil su mantenimiento.

A continuación, se muestra el diseño final del modelo copo de nieve del Data Warehouse de la panadería, representado en la figura 10.

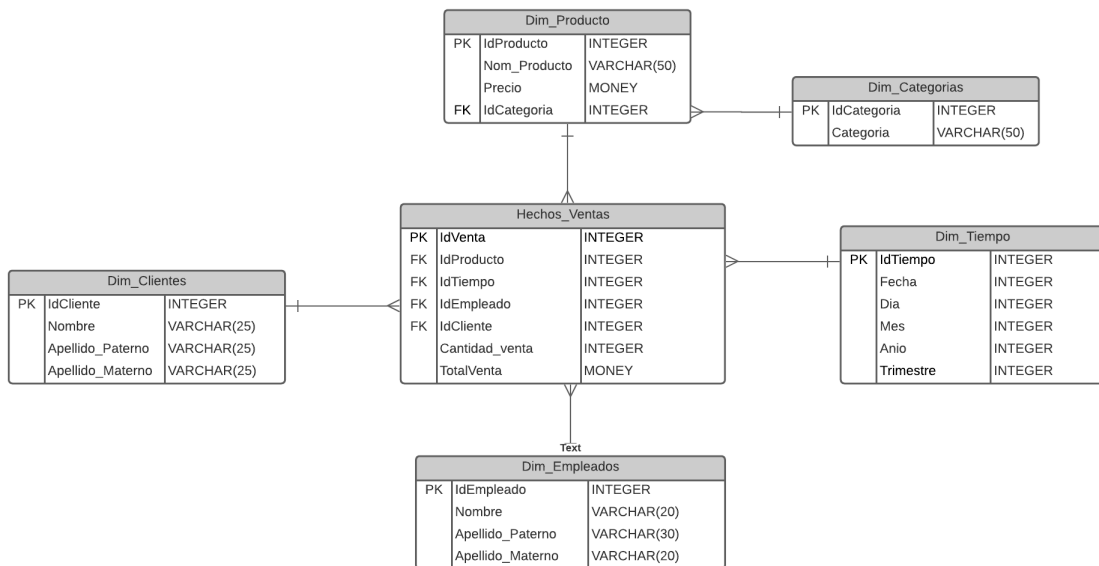


Figura 10: Esquema copo de nieve del Data Warehouse

9. Diccionario de datos de Data Warehouse panadería

9.1. Tabla Dim_Categorías

La tabla Dim_Categorías contiene las categorías del pan con las que cuenta en la panadería. Como se muestra en la tabla 15.

| Categorías | | | |
|--------------|---------|--------------|--|
| Campo | PK o FK | Tipo de dato | Descripción |
| IdCategorías | PK | INT | Almacena el identificador de cada fila de la tabla |
| Nombre | | VARCHAR(50) | Almacena el nombre de la categoría |

Tabla 15: Tabla Dim_Categoría

9.2. Tabla Dim_Clientes

La tabla clientes información sobre el cliente registrado en la panadería. Como se muestra en la tabla 16.

| Clientes | | | |
|------------------|---------|--------------|--|
| Campo | PK o FK | Tipo de dato | Descripción |
| IdCliente | PK | INT | Almacena el identificador de cada fila de la tabla |
| Nombre | | VARCHAR(50) | Almacena el nombre del cliente |
| Apellido_Paterno | | VARCHAR(50) | Almacena el Apellido paterno del cliente |
| Apellido_Materno | | VARCHAR(50) | Almacena el Apellido Materno del cliente |

Tabla 16: Tabla Dim_Clientes

9.3. Tabla Dim_Empleados

La tabla Empleados contendrá la información sobre los empleados con los que cuenta la panadería. Como se muestra en la tabla 17.

| Empleados | | | |
|------------------|---------|--------------|--|
| Campo | PK o FK | Tipo de dato | Descripción |
| IdEmpleado | PK | INT | Almacena el identificador de cada fila de la tabla |
| Nombre | | VARCHAR(50) | Almacena el nombre del empleado |
| Apellido_Paterno | | VARCHAR(50) | Almacena el Apellido paterno del empleado |
| Apellido_Materno | | VARCHAR(50) | Almacena el Apellido Materno del empleado |

Tabla 17: Tabla Dim_Empleados

9.4. Tabla Hechos_Ventas

La tabla Hechos_Ventas es la tabla primaria de este data Warehouse, ya que en ella se encuentran las llaves foráneas hacia las demás tablas, en pocas palabras es la tabla central del data Warehouse. Como se muestra en la tabla 18.

| Empleados | | | |
|----------------|---------|--------------|--|
| Campo | PK o FK | Tipo de dato | Descripción |
| IdVenta | PK | INT | Almacena el identificador de cada fila de la tabla |
| IdProducto | FK | INT | llave foránea que hace referencia a la tabla Productos |
| IdEmpleado | FK | INT | llave foránea que hace referencia a la tabla Empleados |
| IdTiempo | FK | INT | llave foránea que hace referencia a la tabla Tiempo |
| IdCliente | FK | INT | llave foránea que hace referencia a la tabla Clientes |
| Cantidad_Venta | | INT | Almacena la cantidad de piezas de pan compradas |
| Total_Venta | | INT | Almacena la cantidad de dinero pagado por la compra |

Tabla 18: Tabla Hechos_Ventas

9.5. Tabla Dim_Productos

La tabla productos almacena la información de los productos que vende la panadería, así como llevar un control de la existencia del producto, el precio de venta del producto y el precio de venta al mayoreo de cada producto. Como se muestra en la tabla 19.

| Empleados | | | |
|--------------|---------|--------------|---|
| Campo | PK o FK | Tipo de dato | Descripción |
| IdProducto | PK | INT | Almacena el identificador de cada fila de la tabla |
| Nom_Producto | | VARCHAR(50) | Almacena el nombre del producto |
| Existencia | | INT | Almacena la cantidad del producto en existencia |
| Precio | | INT | Almacena el costo individual de un producto |
| IdCategoria | FK | INT | Llave foránea que hace referencia a la tabla Categorías |

Tabla 19: Tabla Dim_Productos

9.6. Tabla Dim_Tiempo

La tabla Tiempo almacena la fecha y la fecha dividida por día, mes, año. Como se muestra en la tabla 20.

| Empleados | | | |
|-----------|---------|--------------|--|
| Campo | PK o FK | Tipo de dato | Descripción |
| IdTiempo | PK | INT | Almacena el identificador de cada fila de la tabla |
| Fecha | | Date | Almacena una fecha |
| Dia | | INT | Almacena un número de día |
| Mes | | INT | Almacena un número de mes |
| Anio | | INT | Almacena un número de año |
| Trimestre | | INT | Almacena un número de trimestre del año |

Tabla 20: Tabla Dimension_tiempo

10. Llenado de las tablas del Data Warehouse

Una vez terminado nuestro Data Warehouse nuestra base de datos se encuentra vacía, además de que algunas tablas y atributos no concuerdan con los de la base de datos original se tuvo que generar unos archivos con procesos necesarios para realizar correctamente la inserción de datos dentro del Data Warehouse desde la base de datos original. Lo cual es uno de los primeros objetivos de utilizar los datos originales de la primera base de datos para poder realizar diferentes consultas las cuales pueden ser solo posibles gracias al Data Warehouse.

A continuación, se muestran las sentencias SQL utilizadas para realizar el llenado de las tablas en el Data warehouse.

```
insert into DW_PanaderiaFinal..Dim_Categorias select * from panaderiaFinal..Categoria;
```

```
insert into DW_PanaderiaFinal..Dim_Productos select (IdProducto,Nom_Producto,Precio,
IdCategoria) from panaderiaFinal..Productos;
```

```
insert into DW_PanaderiaFinal..Dim_Clientes select (IdCliente,Nombre,Apellido_Paterno,
Apellido_Materno) from panaderiaFinal..Clientes;
```

```
insert into DW_PanaderiaFinal..Dim_Tiempo(IdTiempo,Fecha,Dia,Mes,Anio)
select panaderiaFinal..Ventas.IdTiempo,panaderiaFinal..Ventas.Fecha,
Day(panaderiaFinal..Ventas.Fecha),
MONTH(panaderiaFinal..Ventas.Fecha),
YEAR(panaderiaFinal..Ventas.Fecha),
QUARTER(panaderiaFinal..Ventas.Fecha)
from panaderiaFinal..Ventas;
```

```
insert into DW_PanaderiaFinal..Hechos_Ventas (IdVenta,
IdProducto, IdEmpleado, IdTiempo, IdCliente, Cantidad_Venta, Total_Venta)
select panaderiaFinal..Ventas.IdVenta, panaderiaFinal..DetalleVentas.IdProducto,
panaderiaFinal..Ventas.IdEmpleado, panaderiaFinal..Ventas.IdVenta,
panaderiaFinal..Ventas.IdCliente, panaderiaFinal..DetalleVentas.Cantidad,
panaderiaFinal..DetalleVentas.Total_Venta from panaderiaFinal..Ventas inner join
panaderiaFinal..DetalleVentas on Ventas.IdDetalleVenta=DetalleVentas.IdDetalleVenta;
```

Ultima de las sentencias para el llenado de las tablas.

```
insert into DW_PanaderiaFinal..Dim_Empleados (IdEmpleado,Nombre,
Apellido_Paterno,Apellido_Materno,Fecha_Nacimiento)
select panaderiaFinal..Empleados.IdEmpleado,
panaderiaFinal..Empleados.Nombre, panaderiaFinal..Empleados.Apellido_Paterno,
panaderiaFinal..Empleados.Apellido_Materno, panaderiaFinal..Empleados.Telefono,
panaderiaFinal..Empleados.Direccion, panaderiaFinal..Empleados.Fecha_nacimiendo
from panaderiaFinal..Empleados;
```

11. Creación del cubo OLAP

Con la robustez de los datos con el paso del tiempo aparece una gran tecnología llamada cubos OLAP que son una parte importante, ya que permite un uso eficaz de las dimensiones para el análisis de los datos proporcionando respuestas rápidas a consultas analíticas complejas e iterativas, por lo que se vuelve una excelente tecnología que contiene las técnicas de agregación de datos, organizan y resumen grandes volúmenes de datos para que puedan ser evaluados con rapidez. En el momento de la creación del cubo OLAP se notó que existían problemas con algunas columnas de diferentes tablas, debido a los nombres iguales en las columnas de las tablas Dim_Empleado y Dim_Cliente que fueron tomados como duplicidad por lo que hubo la necesidad de hacer correcciones las cuales fueron cambiar el nombre de las columnas de la tabla Dim_Empleado para hacer la diferencia y así sea exitosa la creación. Para la creación de los cubos olap en Power BI se utilizó el siguiente script:

```
Cubo = GROUPBY('Hechos_Ventas',[IdVenta],Dim_Categorias[Categoria],  
Dim_Clientes[Apellido_Materno],Dim_Clientes[Apellido_Paterno],Dim_Clientes[Nombre],  
Dim_Empleados[Apellido_Materno_Emp],Dim_Empleados[Apellido_Paterno_Emp],  
Dim_Empleados[Nombre_Emp],Dim_Productos[Nom_Producto],  
Dim_Productos[Precio],Dim_Tiempo[Fecha],Hechos_Ventas[Cantidad_Venta])
```

Para la generación del cubo, se crea una tabla en la sección de Datos en Power Bi, al crearla aparece una barra con el nombre de la tabla y el símbolo ””, de acuerdo a la sintaxis de la creación de un cubo, se utiliza la función GROUPBY y se abren y cierran corchetes, dentro de los corchetes, en primer lugar y entre comillas simples va el nombre de la tabla de hechos (ejemplo 'hechos_ventas'), en segundo lugar, separado por una ”,” del nombre de la tabla de hechos y encerrado entre corchetes es el id de la tabla de hechos (ejemplo [IdVenta]), una vez teniendo esto se pueden mandar a llamar los datos de todo el diagrama de copo de nieve, ya que Power Bi se encarga de relacionar las tablas del diagrama no es necesario mandar a llamar los id de cada tabla ni hacer inner join, solo con especificar el nombre de la tabla y entre corchetes la columna deseada (ejemplo productos[Precio]) en la figura 11 se muestra un ejemplo de lo explicado.

Nombre: Cubo

Herramientas de tablas

1 Cubo = GROUPBY([hechos_ventas',[IdVenta],hechos_ventas[Cantidad_Venta],productos[Nom_Producto],productos[Precio],clientes[Nombre],clientes[Apellido_Paterno],clientes[Apellido_Materno],empleados[Nombre_Emp],empleados[Apellido_Paterno_Emp],empleados[Apellido_Materno_Emp],tiempo[Anio],tiempo[Dia],tiempo[Fecha],tiempo[Mes],tiempo[Trimestre]])

| IdVenta | Cantidad_Venta | Nom_Producto | Precio | Nombre | Apellido_Paterno | Apellido_Materno | Nombre_Emp | Apellido_Paterno_Emp | Apellido_Materno_Emp |
|---------|----------------|------------------------------------|--------|------------|------------------|------------------|------------|----------------------|----------------------|
| 51 | 2 | Cherries - Fresh | \$14 | Benita | Howatt | Lyles | Gillan | Vequaud | Fawdrie |
| 72 | 2 | Carbonated Water - Blackcherry | \$11 | Carlita | Daid | Tomaszewski | Gillan | Vequaud | Fawdrie |
| 101 | 2 | Pasta - Rotini, Dry | \$10 | Marion | Le Fleming | Haxell | Keir | Hamsson | Tenby |
| 148 | 2 | Compound - Mocha | \$10 | Vevay | Canby | Shugg | Adriaens | Bird | Richardot |
| 214 | 2 | Bread - Frozen Basket Variety | \$6 | Ronna | Howlin | Mc Grath | Hurlee | Mosssdale | Randell |
| 240 | 2 | Rum - Light, Captain Morgan | \$12 | Elsy | Giannazzi | Loton | Alayne | Culshaw | Guillotn |
| 317 | 2 | Chicken - Wings, Tip Off | \$7 | Joelle | Anlay | Chelsom | Hurlee | Mosssdale | Randell |
| 324 | 2 | Papayas | \$7 | Salmon | Sharrock | Walder | Adamo | McGavin | Huison |
| 642 | 2 | Wine - Valpolicella Masi | \$8 | Franklin | Fick | Wilcocke | Adriaens | Bird | Richardot |
| 729 | 2 | Pasta - Fett Alfredo, Single Serve | \$13 | Winona | Halbeard | Pardi | Frederico | MacAscaidh | Clemon |
| 757 | 2 | Browning Caramel Glace | \$14 | Damiano | McSharry | Offlier | Catherin | Dohr | Procter |
| 765 | 2 | Beans - Long, Chinese | \$8 | Olimpia | Buckell | Brunning | Frederico | MacAscaidh | Clemon |
| 897 | 2 | Shrimp - 100 / 200 Cold Water | \$10 | Meg | McCosh | Hadwen | Angelo | Worshall | Jack |
| 903 | 2 | Wine - Cousino Macul Antiguas | \$14 | Esmeralda | D'Ruel | Glawsop | Betsey | Sallier | Hubbert |
| 970 | 2 | Juice - Orange 1.89l | \$10 | Deva | Preddle | Wemm | Aurore | Verdon | Strangward |
| 6 | 7 | Quail - Jumbo Boneless | \$7 | Northrup | Reitenbach | Brackenbury | Keir | Hamsson | Tenby |
| 71 | 7 | Tomatoes | \$12 | Margarethe | Swanne | Soal | Gillan | Vequaud | Fawdrie |
| 106 | 7 | Cleaner - Lime Away | \$14 | Rosana | Myderscough | Tunder | Alayne | Culshaw | Guillotn |
| 133 | 7 | Trout - Rainbow, Frozen | \$11 | Gilbert | Toffalo | Douglass | Angelo | Worshall | Jack |
| 215 | 7 | Mop Head - Cotton, 24 Oz | \$12 | Tarah | Tureville | Whetson | Gillan | Vequaud | Fawdrie |
| 364 | 7 | Lettuce - Mini Greens, Whole | \$10 | Hayes | Chittle | D'Elias | Nan | Hullock | Quilty |

Figura 11: Ejemplo al implementar un Cubo Olap

12. Creación de informes

Una vez creado el data warehouse posteriormente se procedió a realizar el cubo OLAP con el fin de tener la posibilidad de crear los informes que proveen de diseño gráfico para visualizar de una mejor forma los datos los cuales deseamos analizar.

12.1. Informe de Ventas

Lo primero que se realizó fue el informe de las ventas, donde puedes seleccionar ya sea por un nombre de un producto, año, mes, trimestre o día, de acuerdo a lo seleccionado se cargará la información en la gráfica de barras, en pocas palabras se mostrarán las ventas que sean realizado por los datos seleccionados. Para este informe se seleccionaron los datos productos[Nom_Producto], tiempo[Anio],tiempo[Dia],tiempo[Mes],hechos_ventas[Cantidad_Venta] y tiempo[Trimestre] del cubo, en la figura 12 se muestra el informe.

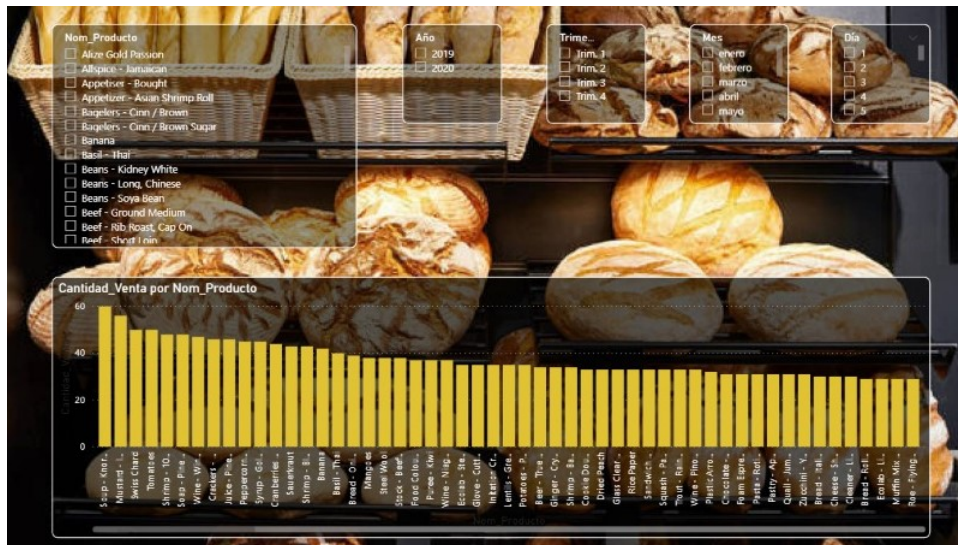


Figura 12: Informe de Ventas

12.2. Informe de Categorías

El segundo informe es sobre categorías, donde puedes seleccionar ya sea por nombre de categoría, mes o año, de acuerdo a lo seleccionado se cargará la información en la gráfica de barras, en pocas palabras se mostrarán las categorías más vendidas que sean realizado de acuerdo a los datos seleccionados. Para este informe se seleccionaron los datos categorías [Categoria], tiempo[Mes], tiempo[Año] y hechos_ventas[Cantidad_Venta], en la figura 13 se muestra el informe.



Figura 13: Informe de Categorías

12.3. Informe de Empleados

El tercer informe es sobre los empleados, donde puedes seleccionar ya sea por nombre de empleado, año o mes, de acuerdo a lo seleccionado se cargará la información en la gráfica de barras, en pocas palabras se mostrarán las ventas que han realizado los empleados ya sea por mes o por año. Para este informe se seleccionaron los datos empleados[Nombre_Emp], tiempo[Mes], tiempo[Año] y hechos_ventas[Cantidad_Venta], en la figura 14 se muestra el informe.

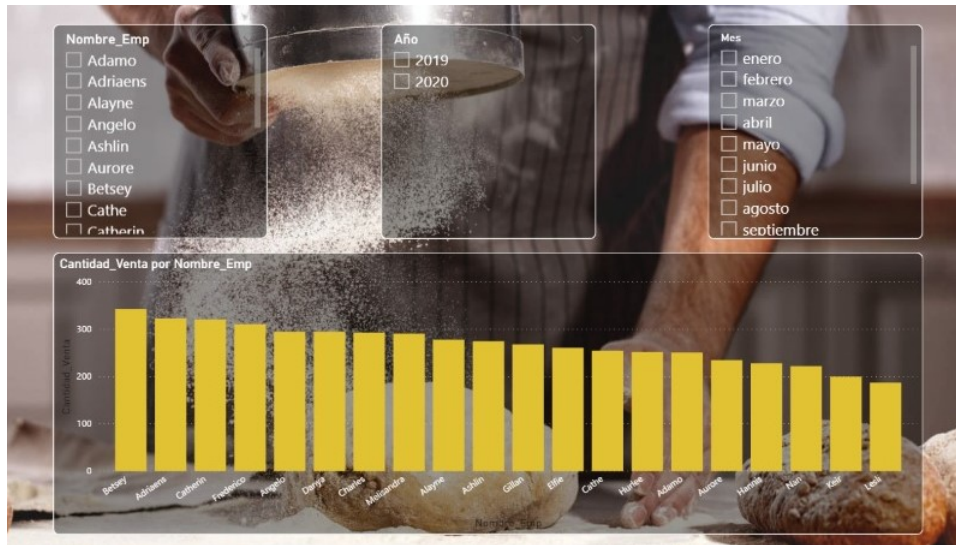


Figura 14: Informe de Empleados

13. Conclusión

Haciendo un recuento de todo lo antes mencionado se comenzó mediante una problemática encontrada en una panadería, con el fin de darle la mejor solución posible se implementó la inteligencia de negocios. El principio de esta implementación fue la creación y aplicación de una base de datos y posteriormente el llenado, la base de datos es la parte fundamental para realizar múltiples aplicaciones y así ofrecer todos los beneficios posibles que nos brinda la inteligencia de negocios, fue tratada por Power BI en el cual mediante funciones se pudo observar un panorama más amplio realizando cambios necesarios en los datos gracias a las funciones que pueden ser aplicadas de una forma sencilla.

Conocimos un concepto muy importante llamado Data Warehouse que nos habla acerca de un modelo de la base de datos el cual es compuesto solo por una parte de la misma para ofrecernos aún más posibilidades para manipular los datos, junto a esto apareció algo que complementa al Data Warehouse llamado Cubo OLAP que nos ayudan a realizar secciones en el Data Warehouse, en secciones se refiere a que se puede seccionar por departamento o un área específica.

Finalizando cada uno de estos procesos se apreció todo lo que puede aportar lo antes mencionado que es "La Inteligencia De Negocios", o también llamada en inglés "business intelligence", incluso a un negocio pequeño al cual lo puede hacer prosperar en armonía compitiendo hasta llegar a un punto muy alto en el mercado, brindando una eficiencia limpia para su manejo de datos y unos análisis precisos con gran relevancia y variedad para tomar en cuenta el mayor número de posibilidades que pueden darse en diferentes situaciones, así qué podemos decir que "La Inteligencia De Negocios", es quizás lo más importante para un negocio y que debe ser utilizada.

14. Referencias

https://troyanx.com/Hefesto/copo_de_nieve.html

https://www.ibm.com/support/knowledgecenter/es/SS9UM9_9.1.2/com.ibm.datatools dimensional.ui.doc/topics/c_dm_snowflake_schemas.html

<https://blog.bi-geek.com/modelo-dimensional/>

<https://www.powerdata.es/data-warehouse#:~:text=Inmon%2C%20el%20cual%20es%20conocido, proceso%20de%20toma%20de%20decisiones.>

<https://datablog.zeus.vision/2017/06/02/que-es-data-warehouse/>

https://www.sinnexus.com/business_intelligence/datawarehouse.aspx

<https://es.wikipedia.org/wiki/OLAP#:~:text=OLAP%20es%20el%20acr%C3%B3nimo%20en, de%20grandes%20cantidades%20de%20datos.>

https://es.wikipedia.org/wiki/Cubo_OLAP

<https://blog.auriboxtraining.com/business-intelligence/los-cubos-olap/>

<https://youtu.be/TXZfkgRhy3s>

<https://docs.microsoft.com/es-es/powerquery-m/m-spec-introduction>

<https://community.powerbi.com/t5/Translated-Spanish-Desktop/C%C3%B3mo-crear-una-tabla-de-td-p/1232662>

https://es.wikipedia.org/wiki/Esquema_en_copo_de_nieve