**EVIDENCIA DE APRENDIZAJE 2. STAGING JARDINERIA**

**OSSMAN MEJÍA GUZMÁN**

**GRUPO PREICA2401B010076**

**OLIMPO DE JESÚS MACEA HERNÁNDEZ**

**GRUPO** [**PREICA2401B010094**](https://iudigital.instructure.com/courses/15609)

**INSTITUCIÓN UNIVERSITARIA DIGITAL DE ANTIOQUIA**

**INGENIERIA DE SOFTWARE Y DATOS**

**BASES DE DATOS II**

**VICTOR HUGO MERCADO**

**MEDELLÍN**

**MAYO DE 2024**

**INTRODUCCIÓN**

Esta evidencia de aprendizaje es la continuación de la evidencia de aprendizaje 1 que responde a los siguientes objetivos de aprendizaje:

1. Da cuenta del diseño de un modelo estrella para un data mart. El modelo estrella es una técnica de modelado de datos que busca, especialmente, optimizar los datos correspondientes a almacenes de datos y data mart. El esquema lógico de un modelo de estrella consta de una tabla central denominada hechos y diversas tablas de dimensiones. La tabla central de hechos, almacena datos susceptibles de análisis, mientras que las tablas de dimensiones almacenan datos utilizados para agrupar, clasificar o filtrar las medidas de la tabla de hechos. El buen diseño de los data mart, asegura la eficiencia de los procesos de inteligencia de negocios, por lo tanto, esta evidencia de aprendizaje permite desarrollar las competencias necesarias para su análisis y diseño dependiendo de las necesidades de la empresa.
2. Incluye el proceso de diseño y creación de la base de datos staging basada en el modelo relacional de la base de datos jardinería. Esto incorpora las siguientes tareas:
   1. Análisis de los datos almacenados en la base de datos Jardinería.
   2. Revisión de los datos almacenados en el Jardinería para identificar cuáles son relevantes y cuáles se deben trasladar a la base de datos Staging.
   3. Construcción de la base de Datos Staging
   4. Diseño de la estructura tablas que estarán en la base de datos Staging.
   5. Construcción de las consultas que permitan traer los registros de Jardinería a la Base de Datos Staging.
   6. Ejecución de las consultas y validar que los datos queden almacenados correctamente en la base de datos Staging.
   7. Construcción del backup de ambas bases de datos.

**OBJETIVO GENERAL**

Analizar y trasladar los datos relevantes de la base de datos de Jardinería a la base de datos Staging, diseñando su estructura, ejecutando consultas de migración y garantizando la integridad de los datos.

**OBJETIVOS ESPEFICOS**

* Realizar la identificación de los datos almacenados en la base de datos de Jardinería, señalando que información es relevante para el proceso de migración a la base de datos Staging.
* Diseñar la estructura de tablas para la base de datos Staging, considerando los tipos de datos, relaciones y restricciones necesarias para el proceso ETL.
* Desarrollar consultas SQL que permitan extraer los datos seleccionados de la base de datos de Jardinería y cargarlos en las tablas correspondientes de la base de datos Staging.
* Ejecutar las consultas de migración y realizar pruebas de los datos almacenados en la base de datos Staging.

**PLANTEAMIENTO DEL PROBLEMA**

En la evidencia de aprendizaje 1 se tomó como caso de estudio, una empresa de jardinería que carece de un sistema que permita realizar el análisis de sus datos de ventas y entender, por ejemplo, las tendencias clave que impulsan su negocio. En dicha evidencia de aprendizaje se planteó el análisis y diseño de un data mart que dé respuesta a las necesidades del negocio para comprender procesos propios de las ventas como el producto más vendido, la categoría con más productos vendidos y el año con más ventas. Esta información ayudará al desarrollo de la inteligencia de negocios en la empresa.

En esta evidencia de aprendizaje 2 surge la necesidad de realizar el análisis y migración de los datos almacenados en la base de datos transaccional hacia una base de datos Staging, esta última definida en pocas palabras como un sistema que se encuentra entre las fuentes de datos y el data mart.

**ANÁLISIS DEL PROBLEMA**

Luego de diseñar el modelo de estrella para el data mart de la empresa de jardinería, en el cual se realizó un análisis detallado de los requisitos y las necesidades del negocio y del modelo relacional es necesario construir la base de datos staging que permita un correcto proceso de ETL hacia el data mart. En este sentido, se llevó a cabo el análisis de los datos almacenados en la base de datos Jardinería con el fin de identificar cuáles son relevantes y cuáles se deben trasladar a la base de datos Staging, según el modelo de estrella planteado.

Posteriormente se realizó la construcción de la base de datos, teniendo en cuenta, la estructura de sus tablas, el diseño de consultas SQL para llevar los registros desde el origen y la ejecución de consultas y validaciones para certificar que los datos queden almacenados correctamente en la Base de datos staging.

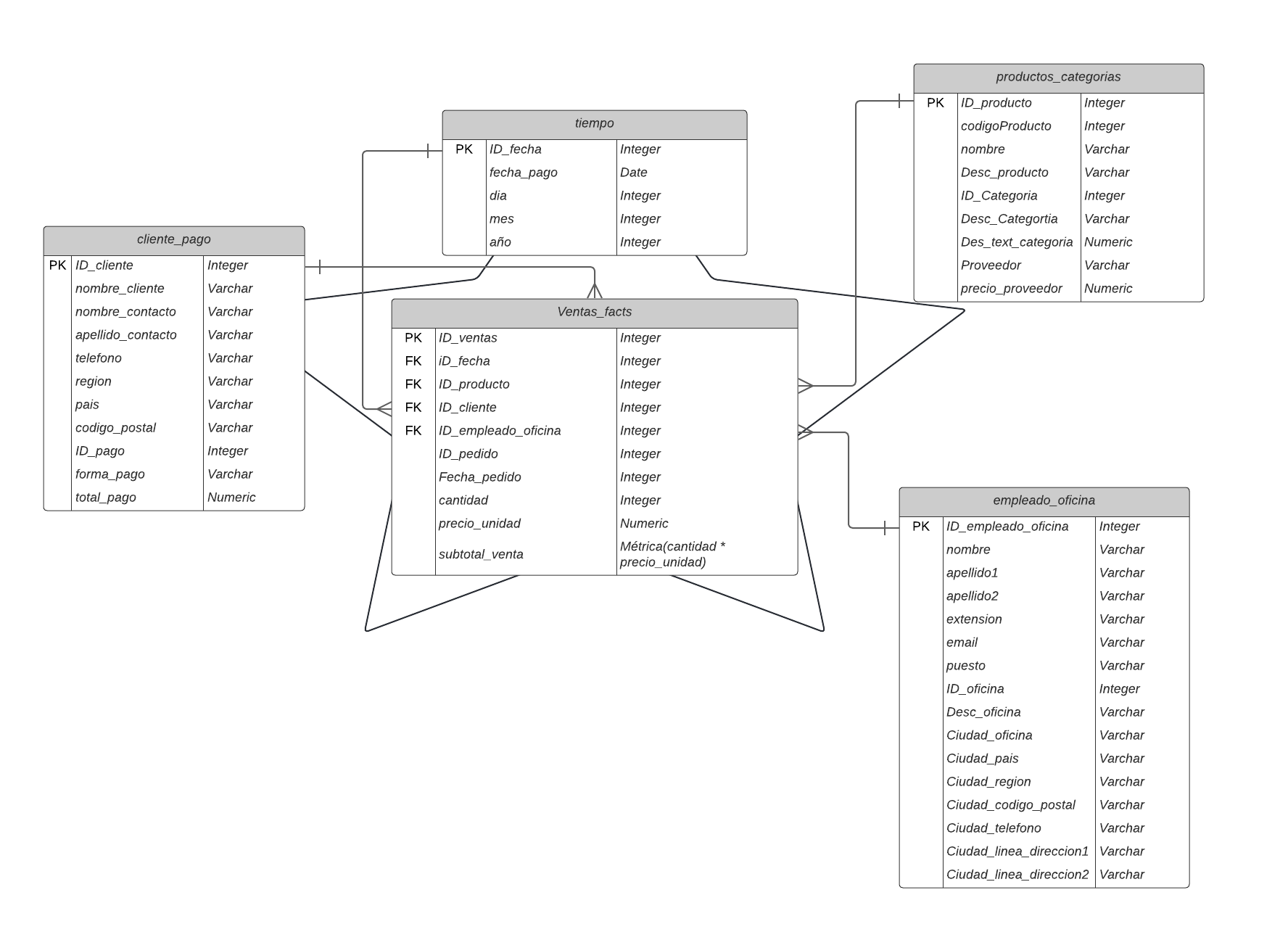
**DESCRIPCIÓN Y AJUSTES AL MODELO DE ESTRELLA**

Este modelo cuenta con una tabla de hechos basada en las tablas *pedidos* y *detalles\_pedidos* del modelo relacional, ya que son las tablas que contienen la información de los pedidos generados con sus diferentes productos y cantidades. Adicional a ello, la tabla de hechos contiene dos campos cuantificables, cantidad y precio unidad para realizar las métricas de productos más vendidos y categoría más vendida.

Las dimensiones del modelo son: *tiempo* que permite consultar las ventas por periodos de tiempo específico, la dimensión *productos\_categoría* que agrupa los datos de las tablas *producto* y *categoría de producto*, la dimensión *cliente\_pago* que facilita la consulta de los clientes a los cuales están asociadas los pedidos y, la dimensión *empleado\_oficina* que agrupa los datos de las tablas *empleado* y *oficina.*

**Esta última tabla se añadió siguiendo la observación que realizó el profesor a la evidencia de aprendizaje 1.**

**DISEÑO DEL MODELO DE ESTRELLA ACTUALIZADO**



**LISTA DE DIMENSIONES PROPUESTAS DEL MODELO ACTUALIZADO**

|  |  |
| --- | --- |
| **Dimensiones** | |
| **Nombre** | **Atributos** |
| Productos\_categoria | ID\_producto  codigoProducto  nombre  Desc\_producto  ID\_Categoria  Desc\_Categoria  Des\_text\_categoria  proveedor  precio\_proveedor |
| Dim\_Tiempo | ID\_Fecha  fecha\_pago  fecha  día  mes  año |
| Cliente\_pago | ID\_cliente  nombre\_cliente  nombre\_contacto  apellido\_contacto  telefono  región  país  codigo\_postal  ID\_pago  forma\_pago  total\_pago |
| Empleado\_oficina | ID\_empleado\_oficina  nombre  apellido1  apellid2  extensión  email  puesto  ID\_oficina  desc\_oficina  ciudad\_oficina  ciudad\_pais  ciudad\_region  ciudad\_codigo\_postal  ciudad\_telefono  ciudad\_linea\_direccion1  ciudad\_linea\_direccion2 |

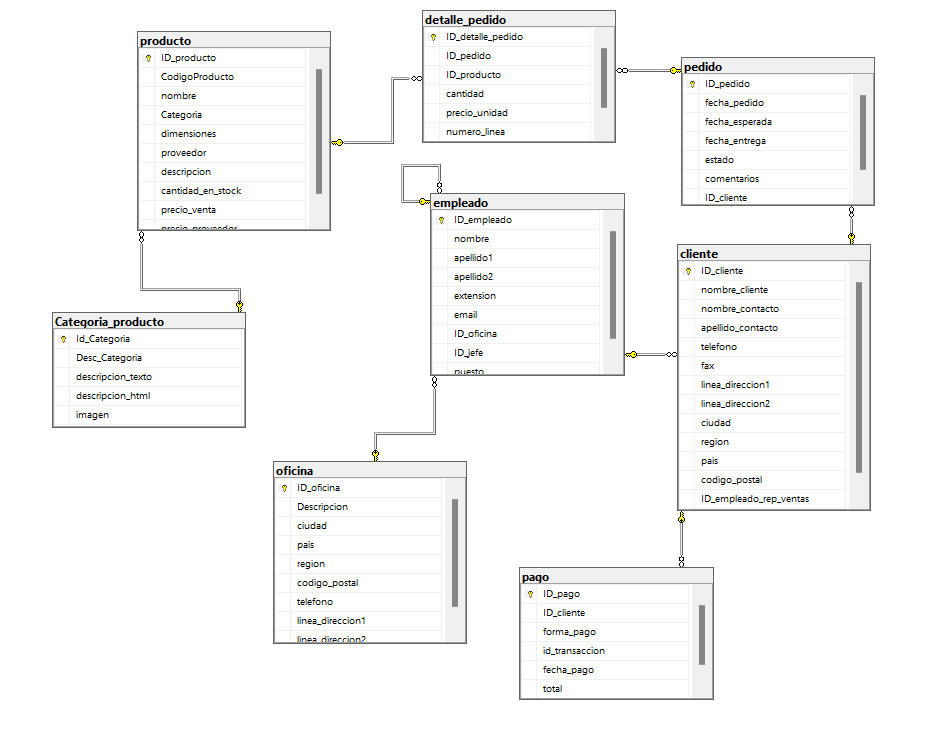
**PROPUESTA DE LA SOLUCIÓN**

Para abordar el problema planteado, se utilizó el diseño de la base de datos Staging mediante Integration Services de Microsoft Visual Community, aprovechando sus características para la extracción, transformación y carga (ETL) de datos. Este proceso se describe a continuación:

**DESCRIPCIÓN DEL ANÁLISIS PRELIMINAR**

La base de datos relacional "jardinería" contiene una serie de tablas que registran información sobre la empresa y abarca la gestión de oficinas, empleados, clientes, pedidos, productos, entre otros. A continuación, se relaciona una descripción de las tablas y los campos más importantes para el análisis:

* **producto**: almacena información sobre los productos ofrecidos, como su nombre, código, descripción, precio, cantidad en stock, etc. También incluye una referencia a la categoría a la que pertenece.
* **categoria\_producto**: esta tabla contiene la información sobre las categorías de productos ofrecidos por la empresa de jardinería.
* **oficina**: esta tabla guarda información sobre las oficinas de la empresa. Registra datos como la descripción, ciudad, país, región, código postal, teléfono y dirección.
* **empleado**: almacena detalles sobre los empleados, incluyendo su nombre, apellidos, extensión, correo electrónico, puesto, y a qué oficina pertenecen.
* **cliente**: la tabla registra los datos de los clientes, como su nombre, contacto, dirección, país, región, etc.
* **pedido**: esta tabla rastrea los pedidos realizados por los clientes, incluyendo la fecha de pedido, fecha esperada de entrega, fecha de entrega (si está disponible), estado del pedido y comentarios adicionales.
* **detalle\_pedido**: registra los detalles específicos de los productos incluidos en cada pedido, como la cantidad y precio unitario.
* **pago**: guarda datos sobre los pagos realizados por los clientes, incluyendo la forma de pago, ID de transacción, fecha de pago y el total pagado.



**CREACIÓN DE BASE DE DATOS STAGING**

El proceso de creación de la base de datos staging se realizó mediante la plataforma para la creación de soluciones de transformaciones de datos SQL Server Integration Services en el IDE Microsoft Visual Studio Community. A continuación, se describe dicho proceso:

1. Creación del proyecto de SQL Server Integration Services
2. Creación de las conexiones de datos para conexiones ADO.NET hacia las bases de datos de jardinería y Staging\_Jardineria, previamente creadas en un servidor de SQL Server.
3. Creación de un flujo de control “Extract\_Jardineria” para el proceso de creación del Staging.
4. Creación de los orígenes y destinos para las tablas y campos del modelo dimensional propuesto, de la siguiente manera:

|  |  |  |
| --- | --- | --- |
| **Dimensión o Hecho esperado** | **Tablas de origen** | **Tablas de destino** |
| Cliente\_pago | **Origen\_Cliente:**  ID\_cliente  nombre\_cliente  nombre\_contacto  apellido\_contacto  telefono  region  pais  codigo\_postal | Crea una nueva tabla llamada "cliente" con una estructura similar a la tabla de origen, pero con una columna adicional para ID\_cliente\_o. |
| **Origen\_Pago:**  ID\_pago  ID\_cliente  forma\_pago  total | Crea una nueva tabla llamada "pago" con una estructura similar a la tabla de origen, pero con una columna adicional para ID\_pago\_o. |
| Producto\_Categoria | **Origen\_Producto:**  ID\_producto  CodigoProducto  nombre  Categoria  proveedor  descripcion  precio\_proveedor | Crea una nueva tabla llamada "producto" con una estructura similar a la tabla de origen, pero con una columna adicional para ID\_producto\_o. |
| **Origen\_Categoria:**  Id\_Categoria  Desc\_Categoria  descripcion\_texto | Crea una nueva tabla llamada "Categoria" con una estructura similar a la tabla de origen, pero con una columna adicional para Id\_Categoria\_o. |
| Empleado\_Oficina | **Origen\_Empleado:**  ID\_empleado  nombre  apellido1  apellido2  extension  email  ID\_oficina  Puesto | Crea una nueva tabla llamada "Empleado" con una estructura similar a la tabla de origen, pero con una columna adicional para ID\_empleado\_o. |
| **Origen\_Oficina:**  ID\_oficina  Descripcion  ciudad  pais  region  codigo\_postal  telefono  linea\_direccion1  linea\_direccion2 | Crea una nueva tabla llamada "Oficina" con una estructura similar a la tabla de origen, pero con una columna adicional para ID\_oficina\_o. |
| Tiempo | **Origen\_Tiempo:**  fecha\_pago | Crea una nueva tabla llamada "Tiempo" para almacenar las fechas de pago únicas. |
| Fact\_Ventas | **Origen\_Pedido:**  ID\_pedido  fecha\_pedido  ID\_cliente | Crea una nueva tabla llamada "Pedidos" con una estructura similar a la tabla de origen, pero con una columna adicional para ID\_pedido\_o. |
| **Origen\_Detalle\_Pedido:**  ID\_detalle\_pedido  ID\_pedido  ID\_producto  cantidad  precio\_unidad | Crea una nueva tabla llamada "Detalle\_Pedidos" con una estructura similar a la tabla de origen, pero con una columna adicional para ID\_detalle\_pedido\_o. |

Finalmente, se presenta una tarea - instrucción SQL TRUNCATE para limpiar todas las tablas en preparación para cargar nuevos datos. Esto eliminará todos los registros, pero mantendrá la estructura de las tablas.

**Los detalles de las consultas SQL que se realizaron para la creación de la base de datos Staging pueden ser consultados en el archivo anexo “QuerysStaging.sql”**

**CONCLUSIONES**

Este ejercicio de evidencia de aprendizaje ha permitido desarrollar las capacidades para la migración de datos hacia una base de datos Staging, este es un proceso que requiere un análisis detallado de los requisitos del negocio, la estructura de los datos de origen y destino y del manejo de las herramientas ETL.

Frente a este último punto, se concluye que la utilización de Integration Services de Microsoft Visual Community para diseñar la base de datos Staging es una actividad que proporciona una solución adecuada y robusta que facilita la extracción, transformación y carga de datos.

Así mismo, es importante que la implementación de los procesos ETL cuenten con un plan bien estructurado, que incluya la identificación de datos relevantes del modelo, el diseño de la estructura de la base de datos staging y la ejecución de pruebas.

Finalmente, es importante destacar que la adopción de una base de datos staging ofrece beneficios como el control de calidad de los datos, la protección ante cambios en el esquema de las bases de datos y la concentración de datos de múltiples fuentes.

**ANEXOS**

Se anexa carpeta comprimida que contiene los siguientes archivos:

* **BK\_Proyecto\_Integration\_Services\_Jardineria:** Carpeta que contiene todo el proyecto de Integration Services de Microsoft Visual Community.
* **Jardineria.bak**: Archivo de respaldo de la base de datos de jardinería.
* **Staging\_jardineria.bak**: Archivo de respaldo de la base de datos staging para el proceso ETL del data mart.
* **QuerysStaging.sql**: Archivo que contiene las consultas SQL que se realizaron para la creación de la base de datos staging.
* **Enlace a repositorio GitHub del proyecto**: <https://github.com/ossmanmejia/BI_Jardineria>