



Universidad Tecnológica Nacional
Facultad Regional Buenos Aires

Gestión de Datos

Trabajo Práctico

1° Cuatrimestre 2023

FRBA – Delivery

ESTRATEGIA

Grupo N° 13

FUSECHUDA

Apellido, Nombre	Legajo
D'Amario Tomasello, Ivan	1645419
Fukushima, Federico	1284939
Chura Flores, Belen	1558432
Segesso, Guido	1687580

Índice

Contenido

Modelo Relacional	2
Aclaraciones	2
Provincia y Localidad	2
Movilidad / Días	2
Productos	2
Productos por Pedidos	2
Productos por Local	2
Usuario / Repartidor / Operador	2
Envio Pedido / Envio Mensajería	3
Medios de pago	3
Stored Procedures	3
Errores Detectados	3
Pedido Total	3
Medios de Pago	3
DER	4
Modelo Inteligencia de Negocios	5
DER BI	Error! Bookmark not defined.

Modelo Relacional

Aclaraciones

Provincia y Localidad

Se decidió separar la información de los datos de Localidad y Provincia en tablas separadas, para así eliminar la redundancia de datos que podría existir en las tablas Local, Usuario y Repartidor. Además, al tener una tabla separada para cada entidad, se puede actualizar la información en un solo lugar, lo que aumenta la integridad de los datos.

Movilidad / Días

Para estos casos, creímos prudente tener una tabla separada para cada entidad. Si bien estos datos no se repiten en distintas entidades, quizá en un futuro sí. Pensamos que era más apropiado abstraer los datos e individualizarlos con un id.

Productos

Decidimos crear una tabla productos, la cual solo incluiría información exclusiva de los productos existentes. De esta manera, al ser el código de producto la clave primaria de esta tabla, el resto de las tablas que conozcan o utilicen algún producto, podrán referenciar al código del producto

Productos por Pedidos

En este caso decidimos generar una tabla intermedia Producto_Pedido ya que es una manera de implementar la relación muchos a muchos entre las tablas Pedido y Producto. Permite que un pedido tenga varios productos y que un producto pueda ser incluido en varios pedidos. Esto mejora la eficiencia de la base de datos y evita la duplicación de datos.

Además, la tabla intermedia Producto_Pedido almacena información adicional, por ejemplo: la cantidad de productos que se incluyen en cada pedido, el precio de cada producto en un pedido específico y el total por producto. Decidimos crear este campo para que figure este dato histórico, ya que los precios de los productos pueden modificarse en un futuro.

Productos por Local

Aquí también hemos decidido generar una tabla intermedia Producto_Local por los mismos motivos previamente mencionados para el caso de Producto_Pedido, ya que es una relación de muchos a muchos.

Usuario / Repartidor / Operador

Observamos en este caso que, si bien las 3 entidades tienen atributos similares, era conveniente mantenerlos como entidades distintas, ya que cada uno de ellos tiene atributos y relaciones que son específicos y únicos, y habría que implementar una entidad mucho mas grande, la cual tenga los atributos de todos. Pero en este caso habría muchos registros con atributos en NULL.

Además, separar las tablas también nos permite establecer restricciones y validaciones específicas para cada uno

Envio Pedido / Envio Mensajeria

Como observamos que para la entidad Pedido había una entidad Envio, decidimos utilizar el mismo concepto para el Servicio_mensajeria. Creamos la entidad Envio_Mensajeria, que seria el envío propio de la entidad Servicio_Mensajeria. Analizamos usar una sola entidad llamada Envio, que contenga lo necesario para combinar ambas entidades de Envio, pero por las mismas razones mencionadas anteriormente, creímos conveniente mantenerlas como entidades separadas. Y de este modo, la entidad Servicio_Mensajeria no quedaría tan cargada de atributos.

Medios de pago

Para el caso de los Medios de Pago, decidimos crear una tabla que contuviera únicamente las Marcas/Emisores de las distintas tarjetas, y otra tabla que contendría los tipos de medio de pago existentes, para poder relacionar cada uno de estos con un ID. Luego estas entidades se relacionan con la entidad MdeP_Usuario, la cual contiene los distintos medios de pagos que decida guardar cada usuario. Decidimos que su clave primaria sería un ID auto incremental, y sus datos harían referencia al ID de Usuario, al ID de Medio de Pago, y al ID de Marca/Emisor de la tarjeta. También contiene el dato de Número de tarjeta.

Stored Procedures

Pensamos que sería más práctico crear un Stored Procedure para los Insert a cada entidad. Por ello se encontrará un Stored Procedure con el nombre Migrar_”Nombre_Tabla” por cada tabla que hemos creado. Debajo de la creación de estos se encontrarán todos los execute de dichos Procedures para, ahora sí, ejecutar los Inserts y realizar la migración.

Errores Detectados

Pedido Total

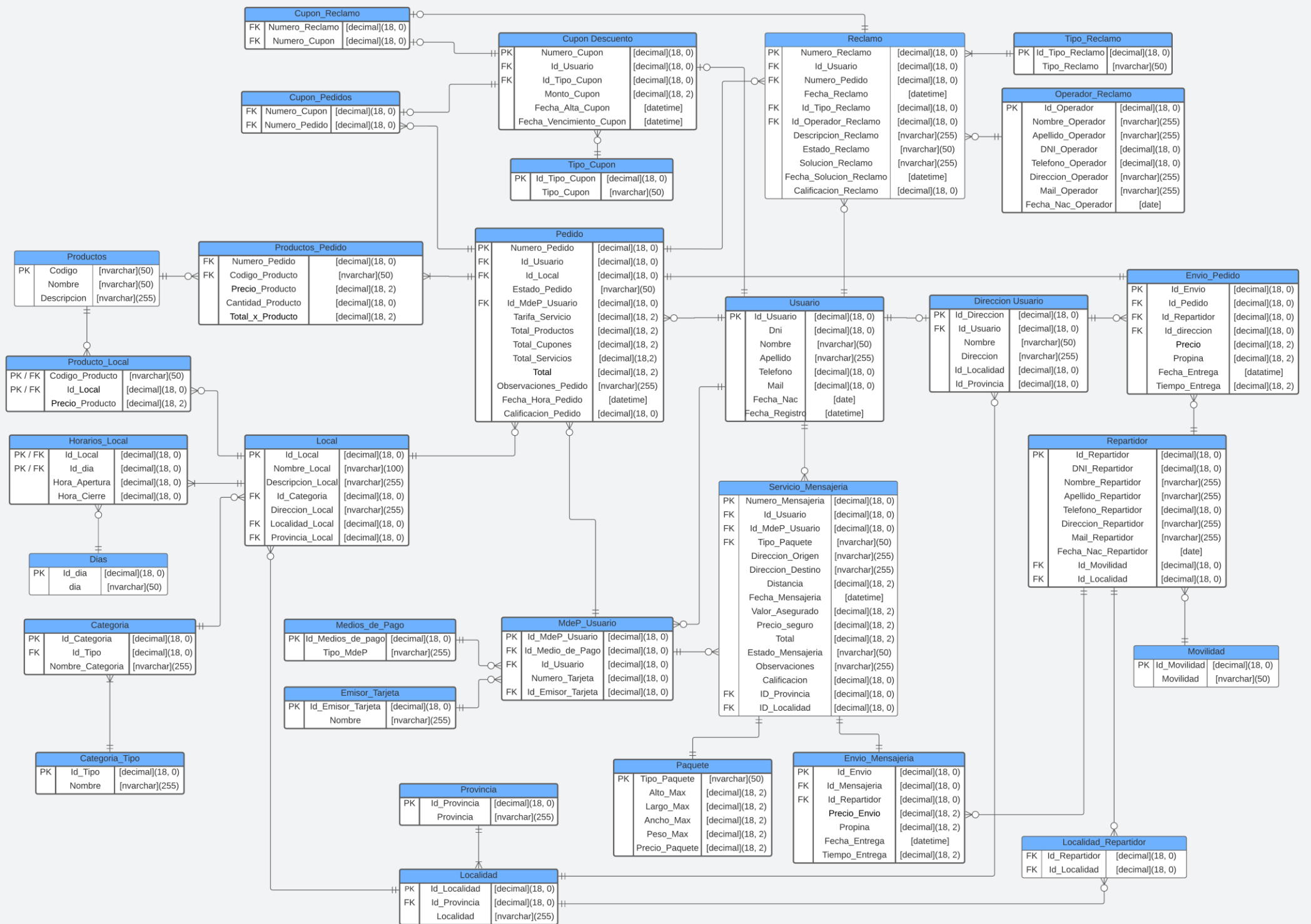
Observamos que, al migrar los datos de los pedidos, y agregar la columna TOTAL, existe una diferencia entre estas en muchos de los casos. De todos modos, por más que difieran, los datos se migraran igual. La columna TOTAL está contemplada de la siguiente manera:

$$\text{TOTAL_PRODUCTOS} + \text{PRECIO_ENVIO} + \text{PROPINA} + \text{TARIFA_SERVICIO} - \text{TOTAL_CUPON}$$

Medios de Pago

Realizando la migración, detectamos que para los casos del medio de pago de tipo “Efectivo”, figuraba un numero de tarjeta junto con su Emisor.

Decidimos migrar estos datos de todos modos teniendo en cuenta que no se utilizaran, ya que este medio de pago no necesita estos datos. Pero respetamos la consigna de migrar todos los datos existentes.



Modelo Inteligencia de Negocios

Comenzamos realizando la creación del esquema y de las dimensiones para después completarlas con los respectivos datos de las tablas correspondientes.

Para las dimensiones consideramos no solo las indicadas en el enunciado del trabajo práctico, sino que además creamos las dimensiones de Usuario, Cupones Reclamos y Cupones Pedidos ya que las consideramos necesarias para ciertas vistas.

Las tablas de hechos que consideramos fueron las tablas de Pedidos, Reclamos y Mensajería ya que son las 3 tablas que representan los eventos principales o transacciones que se desean analizar y que contienen los datos clave que permiten obtener información útil sobre los pedidos, la mensajería y los reclamos en el contexto del modelo de inteligencia de negocios.

Vistas

Para la vista "FUSECHUDA.MayorCantidadPedidos" la creamos para identificar el día de la semana y la franja horaria con la mayor cantidad de pedidos, según la localidad y categoría del local, para cada mes de cada año. Para ello realizamos una consulta para obtener la cantidad de pedidos agrupados por diferentes dimensiones, como el tiempo (mes y año), el día de la semana, la franja horaria, la localidad y la categoría del local. Usamos la función ROW_NUMBER() para asignar un número de fila a cada combinación única de localidad, provincia, categoría de local y tiempo (mes y año), ordenando los registros por la cantidad de pedidos en orden descendente. Después, contamos la cantidad de pedidos para cada combinación de estos criterios. Utilizamos una subconsulta para filtrar y obtener solo las combinaciones de tiempo, localidad, categoría, día y franja horaria que tengan la mayor cantidad de pedidos y después seleccionamos únicamente la fila con el número de fila igual a 1, lo que representa la combinación con la mayor cantidad de pedidos para cada localidad, categoría y mes. La vista final devuelve el día de la semana, la franja horaria, la localidad, la categoría del local, el tiempo (mes y año) y la cantidad máxima de pedidos para cada combinación.

La vista "FUSECHUDA.TotalNoCobrado" se creó para obtener el monto total no cobrado por cada local en función de los pedidos cancelados, considerando el día de la semana y la franja horaria. Realizamos un JOIN con las tablas "FUSECHUDA.[LOCAL]" y "FUSECHUDA.DIAS" para obtener los nombres de los locales y los nombres de los días correspondientes a los identificadores utilizados en la consulta. Aplicamos un filtro WHERE para seleccionar únicamente los pedidos que tienen un estado de cancelado (EstadoPedido = 2). Agrupamos los datos para poder calcular el monto total no cobrado para cada combinación de local, día y franja horaria y por último usamos la función SUM para calcular el total sumando todos los montos de los pedidos cancelados.

Para la vista "FUSECHUDA.ValorEnvioPromedio" se realizó un JOIN con las tablas "FUSECHUDA.BI_Provincia_Localidad" y "FUSECHUDA.BI_Tiempo" para obtener los nombres de las localidades y los identificadores de tiempo correspondientes a los utilizados en la consulta. Luego, se utilizó un GROUP BY para agrupar los datos según el tiempo (mes) y el nombre de la localidad y así calcular el valor promedio mensual de los envíos de pedidos para cada localidad. Finalmente, usamos la función AVG para calcular el valor promedio de los precios de envío en cada combinación de tiempo (mes) y localidad.

La vista "FUSECHUDA.DesvioPromedioDeEntrega" se construye mediante una consulta que realiza un UNION ALL entre dos subconsultas. La primera subconsulta selecciona los datos relevantes de la tabla "FUSECHUDA.BI_Pedidos", mientras que la segunda subconsulta selecciona los datos de la tabla "FUSECHUDA.BI_Mensajeria". Utilizamos un LEFT JOIN para combinar las tablas de rangos horarios y tipos de movilidad, de manera que se obtenga la descripción correspondiente en lugar de los identificadores. Luego agrupamos los datos para calcular el desvío promedio en el tiempo de entrega para cada combinación de estas variables y por último usamos la función AVG para calcular el valor buscado.

Para la vista "FUSECHUDA.MontoTotalCupones" se utiliza la tabla "FUSECHUDA.BI_CuponesPedidos" para obtener los datos de los cupones utilizados. Se realiza un JOIN con la tabla "FUSECHUDA.BI_Pedidos" para obtener información adicional relacionada con cada cupón, como el número de pedido. Luego, se realiza otra combinación para unir la tabla "FUSECHUDA.BI_Rango_Etario" y obtener la descripción correspondiente del rango etario del usuario. Por último, usamos el GROUP BY para agrupar los datos según el mes (obtenido de la tabla "FUSECHUDA.BI_Tiempo"), y el rango etario de los usuarios. Esto permite calcular el monto total de los cupones utilizados en cada combinación de mes y rango etario.

Para la vista "FUSECHUDA.PedidosYMensajeriasEntregados" se realizó un JOIN para obtener la descripción del rango etario de los repartidores, y otra combinación para obtener la localidad relacionada con cada pedido o mensajería. La consulta se divide en dos partes: una para los pedidos y otra para las mensajerías. Dentro de cada parte, se agrupan los datos según el mes, la descripción del rango etario, la localidad y el estado (entregado o no entregado) del pedido o mensajería. Se utiliza la función COUNT para contar los pedidos/mensajerías entregadas y el total de pedidos/mensajerías en cada combinación.

Realizamos un cálculo para obtener el porcentaje de entregados en función del total de dos formas ya que no sabíamos que pretendía bien el enunciado, si en base a la localidad y en base al general. Se utiliza la cláusula GROUP BY para agrupar los resultados según el mes, la localidad y la descripción del rango etario.

La vista "FUSECHUDA.PromedioMensualAsegurado" se creó con una consulta que obtiene los datos de los envíos de paquetes a través del servicio de mensajería. La consulta agrupa los datos según el mes y el tipo de paquete. Utiliza la función AVG para calcular el promedio del valor asegurado de los paquetes en cada combinación. Se obtiene el promedio mensual del valor asegurado de los paquetes en función del tipo y se utiliza el GROUP BY para agrupar los resultados según el mes y el tipo de paquete.

La vista "FUSECHUDA.ReclamosMensuales" se creó para calcular la cantidad de reclamos mensuales recibidos por cada local en función del día de la semana y rango horario. Utilizamos la tabla "FUSECHUDA.BI_Reclamos" para obtener los reclamos registrados. Después, se realiza una serie de JOINS con otras tablas como "FUSECHUDA.BI_Local", "FUSECHUDA.BI_Rango_Horario", "FUSECHUDA.BI_Tiempo" y "FUSECHUDA.DIAS" para obtener la descripción del tiempo, el nombre del local, el día de la semana y la descripción del rango horario correspondiente a cada reclamo, después agrupamos los datos y contamos la cantidad de reclamos para cada combinación para así poder obtener la cantidad de reclamos mensuales por local, día de la semana y rango horario.

Para la vista "FUSECHUDA.PromedioResolucionReclamo" utilizamos la tabla "FUSECHUDA.BI_Reclamos" para obtener la información de los reclamos, y realiza un JOIN con la tabla "FUSECHUDA.BI_Rango_Etario" para obtener la descripción del rango etario de los operadores asociados a cada reclamo. Luego, realizamos un GROUP BY para poder calcular el promedio del tiempo de resolución de reclamos en minutos. Esto se logra restando la fecha de solución del reclamo (FechaSolucion) a la fecha en que se registró el reclamo (Fecha), y luego se calcula el promedio de estos tiempos para cada combinación de tipo de reclamo y rango etario de operador.

Para la vista "FUSECHUDA.MontoCuponesReclamos" utilizamos la tabla "FUSECHUDA.BI_Reclamos" para obtener la información de los reclamos y realiza un JOIN con la tabla "FUSECHUDA.BI_CuponesReclamos" utilizando el número de reclamo RECLAMO_NRO como referencia. Esto permite obtener los cupones asociados a cada reclamo. Después realizamos un JOIN adicional con la tabla "FUSECHUDA.BI_Tiempo" utilizando el campo de tiempo Tiempo presente en la tabla de reclamos permitiendonos asociar cada reclamo a su respectivo periodo de tiempo. Por ultimo, realizamos un GROUP BY para agrupar los datos por mes y calcular la suma del monto de los cupones asociados a los reclamos para cada mes.

