

**Gestión de Datos:**

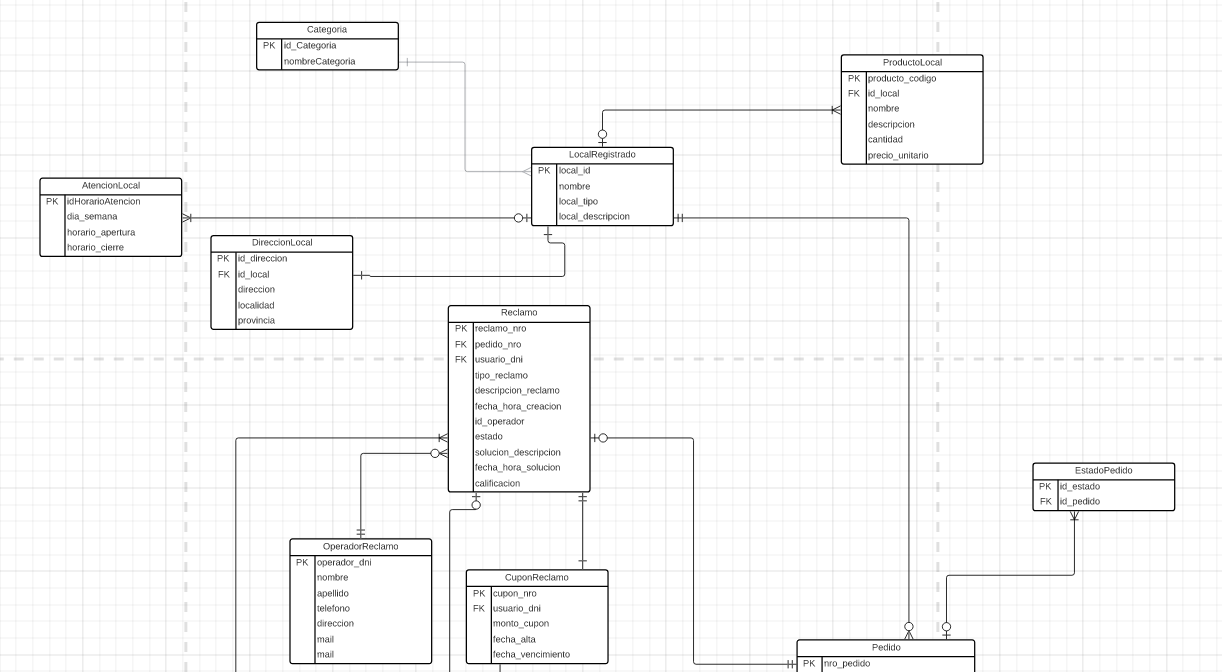
**Trabajo Práctico Delivery**

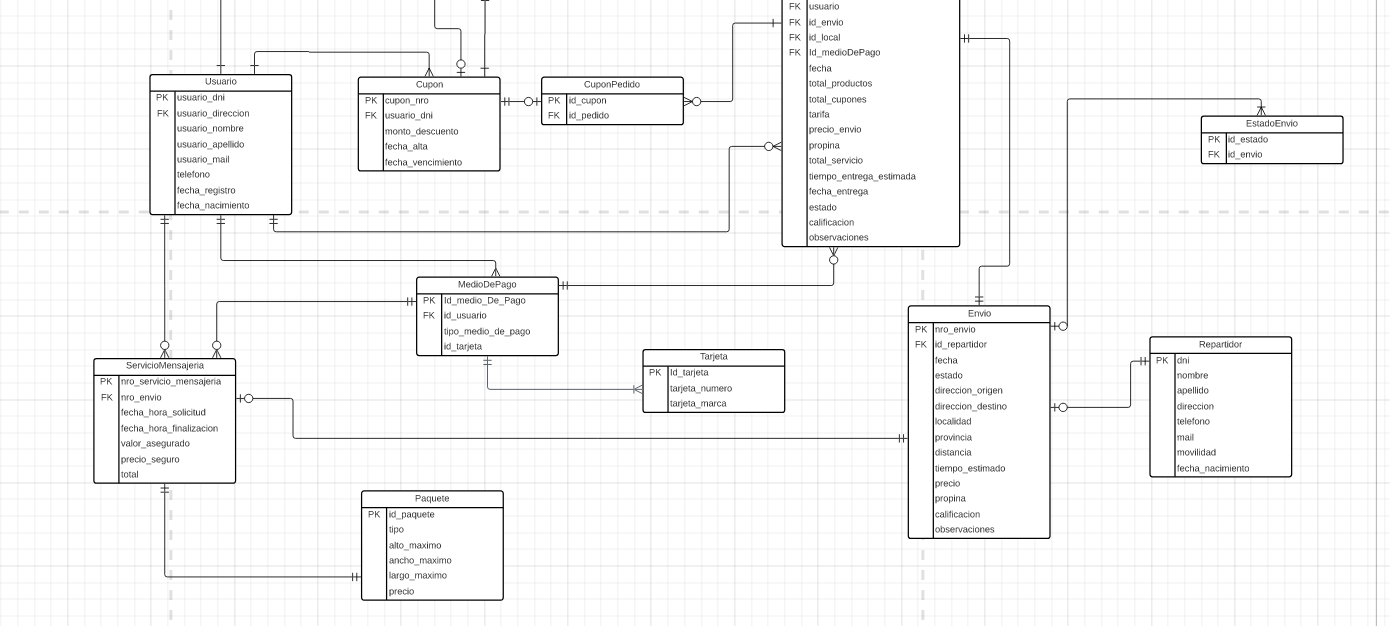
| Alumno | Legajo |
| --- | --- |
| Lucas Patricio Sewrjugin | 1727692 |
| Camila Albite | 1642168 |
| Juan Ignacio Ferro | 1677743 |
| Tomas Zubizarreta | 1725816 |

Trabajo Práctico Cuatrimestral

-2023-

**DER**

****

****

**Decisiones DER**

Decidimos que “LocalRegistrado” esté relacionado con muchas “Categorias” porque varios locales pueden ser de distintas categorías.

También que “LocalRegistrado” tenga muchos “ProductoLocal” asociado ya que un local tiene diferentes productos para ofrecer, a su vez “ProductoLocal” tiene una Foreign Key (FK) asociada al “LocalRegistrado” con el “id\_local”.

“LocalRegistrado” está relacionado con muchas “AtencionLocal” ya que es una entidad de “AtencionLocal” por día de la semana que el local está abierto.

“LocalRegistrado está vinculado únicamente con una “DirecciónLocal” ya que a pesar de que podría haber diferentes franquicias de locales, consideramos que serían otras entidades de “LocalRegistrado” que tengan su propia dirección.

“LocalRegistrado” esta relacionado con muchos “Pedido” ya que un mismo local puede atender varios pedidos, sin embargo cada “Pedido” recibe una FK del “id\_local” para saber a que local pertenece.

“Pedido” esta relacionado uno a muchos con “EstadoPedido”, ya que de los estados que puede adoptar son varios para un mismo pedido (Aceptado, Recibido, etc.), a su vez tiene una Foreign Key que proviene de “Pedido” ya que tiene un Id del pedido asociado al estado. Lo mismo sucede con “Envio” y “EstadoEnvio”.

Un “Pedido” esta relacionado con un unico “Envio” ya que un mismo pedido no puede tener varios envios, a su vez “Pedido” tiene una FK perteneciente de “Envio” para indicar que nro de envio esta relacionado con que pedido.

Un “Envio” esta relacionado con un unico “Repartidor” que se encargara de entregar el pedido, a su vez “Envio” tiene una Foreign Key (FK) asociada a “Repartidor” para saber quien realizara el envio.

Un “Envio” puede ser un “ServicioMensajeria”, por esto estan relacionados uno a uno, a su vez “ServicioMensajeria” tiene una FK asociada al envio que es el “nro\_envio” que esta vinculado con el envio a realizar.

El “ServcicioMensajeria” esta relacionado con un unico “Paquete” que es obligatorio que este para este servicio.

Varios “ServicioMensajeria” pueden ser pagados con un mismo “MedioDePago” ya que se pueden solicitar varios de estos servicios y abonarlos de la misma manera.

Un “MedioDePago” puede tener varias “Tarjeta” asociadas.

Un “MedioDePago” puede estar vinculado con varios “Pedido”, el cual tiene una FK que define el medio de pago a utilizar en el pedido.

Un “Usuario” puede tener muchos “ServicioMensajeria” asociados, ya que puede solicitar esta servicio muchas veces.

“Usuario” esta relacionado con muchos “MedioDePago” ya que este puede optar por pagar con diferentes medios de pago.

Un “Usuario” esta relacionado con muchos “Pedido” ya que este puede solicitar muchos pedidos, a su vez “Pedido” posee una FK asociada al usuario que permite saber que usuario solicito cada pedido.

Un “Usuario” esta relacionado con una “DireccionUsuario” ya que este posee una unica direccion, a su vez “Usuario” recibe una FK de “DireccionUsuario” que permite saber la direccion del usuario.

“Usuario” esta relacionado con muchos “Reclamo” ya que puede realizar muchos reclamos, a su vez “Reclamo” posee una FK proveniente de “Usuario” para saber que usuario realizo el reclamo.

“Usuario” puede estar vinculado a muchos “Cupon” ya que puede haber muchos cupones que pertenezcan a un mismo usuario, a su vez “Cupon” recibe una FK de “Usuario” el cual vincula el cupon con el usuario.

“Reclamo” puede tener muchos “OperadorReclamo” ya que puede ir rotando quien atiende un mismo reclamo.

Un “Reclamo” puede estar relacionado un “Cupon” en el caso de que se haga un reclamo de algun cupon.

“Cupon” esta relaciondado con “CuponReclamo”, que se efectua cuando hay un reclamo por un cupon, tambien recibe una FK de “Cupon” para saber a que usuario esta vinculado el cupon.

Un “Reclamo” esta relacionado con un “CuponReclamo” y verifica que el usuario que posee el cupon sea el mismo que era el que lo reclamo, se verifica la FK de “usuario\_dni” que estan tanto en “Reclamo” como en “CuponReclamo”.

Un “Pedido” puede estar relacionado con muchos “CuponPedido”, el cual recibe una FK de “Pedido” en donde esta el “id\_pedido”

Un “Cupon” esta puede estar relacionado con un “CuponPedido” en el caso que se necesite.

Un “Pedido” puede estar relacionado con un “Reclamo” en el caso que se necesite.

**Decisiones script**

En el procedimiento “migrar\_usuario” habia que especificar que el DNI, nombre, apellido y email no sean “null” ya que definimos en la tabla que no podia ser de esta manera porque seria un usuario sin Primary Key (PK) en el caso del DNI y porque si posee null el resto de campos consideramos que no seria un usuario valido. a su vez, en ese mismo procedimiento decidimos ordenar a los usuario por su DNI.

Algo similar pasa con el procedimiento “migar\_repartidor” con “migar\_usuario” ya que consideramos que si no tiene DNI, nombre, apellido y mail no iba a ser un Repartidor valido.

En el procedimiento “migrar\_local” tuvimos que indicar que “local\_nombre” no sea null ya que era la PK. Tambien utilizamos select distinct para obtener las tablas porque consideramos que no podia haber locales con un mismo id.

En el procedimiento “migrar\_direccion\_local” sucede lo mismo con “migrar\_local” donde “local\_direccion” es la PK.

En el procedimiento “migrar\_atencion\_local” sucede lo mismo con “migrar\_local” donde “horario\_local\_dia” es la PK.

El resto de procedures son simplemente hacer un insert into de las tablas de la base de datos dada (Tabla Maestra) a las tablas que creamos nosotros.