



TRABAJO PRÁCTICO GESTIÓN DE DATOS

FRBA - Inmobiliaria

2do Cuatrimestre 2023

FECHA DE ENTREGA: 27/11/2023

GRUPO: LOS GESTIONADORES

CURSO: K3622

INTEGRANTES DEL GRUPO:

- SCHALLER, GONZALO JORGE
- DEFERRARI, LUCAS ALEJANDRO
- GOLFIERI, ZOE
- NEUSTADT, AGUSTÍN TIAGO

ÍNDICE

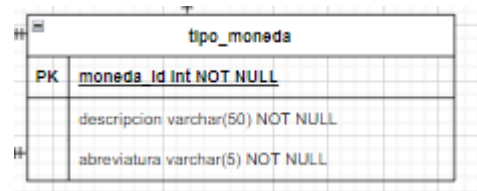
1. <u>Normalización de Tipos de Datos</u>	3
<u>Tipo moneda</u>	
<u>Dirección</u>	
<u>Pago alquiler</u>	
2. <u>Relaciones en la Base de Datos</u>	4
<u>Agente a Anuncio (One to Many)</u>	
<u>Comprador a Venta (Many to One)</u>	
<u>Entidades Separadas para Comprador, Agente y Propietario</u>	
<u>Inmueble a Característica</u>	
3. <u>Business Intelligence (BI)</u>	6
<u>Tabla de Hechos de Venta</u>	
<u>Tabla de Hechos de Alquiler</u>	
<u>Tabla de Hechos de Anuncio</u>	
4. <u>Tablas de Dimensión en BI</u>	8
<u>Tabla de Tiempo</u>	
<u>Tabla de Ubicación</u>	
<u>Tabla bi_pago_alquiler</u>	

Normalización de Tipos de Datos

Tipo moneda

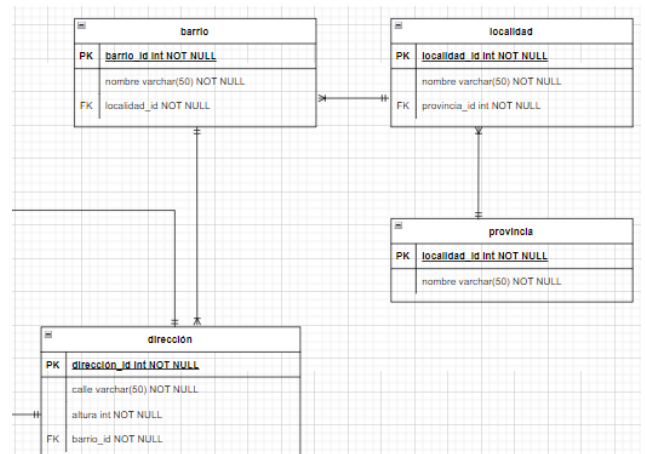
Decidimos utilizar tablas distintas para los campos de operación, moneda, estado del alquiler y tipo periodo ya que son tipos de datos tipificados, y manejarlos de esta manera trae beneficios en términos de normalización, mantenimiento, rendimiento y flexibilidad, mientras que al mismo tiempo ayuda a la escalabilidad.

Todos estas tablas tiene un id y una descripción, en el caso de tipo_moneda también tiene una abreviatura.



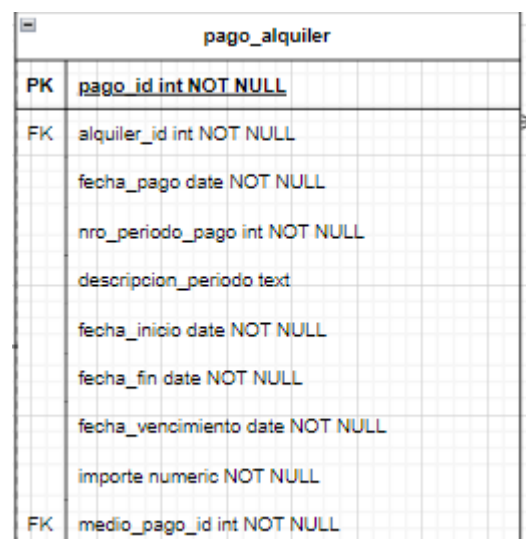
Dirección

Decidimos separar atributos de la dirección como localidad, barrio y provincia ya que al normalizar estos datos flexibiliza nuestro sistema, también ayudando a la mantenibilidad y a la integridad de los datos almacenados. Estamos normalizando debido a que cada atributo depende del otro y no pueden haber atributos no principales dependientes entre ellos.



Pago Alquiler

Decidimos no normalizar el periodo en nuestra base de datos, ya que esta desnormalización aporta flexibilidad a nuestro sistema, facilita su mantenimiento y asegura la integridad de los datos almacenados. Esta decisión se basa en la estabilidad de los campos relacionados con el periodo, los cuales no experimentan cambios frecuentes.

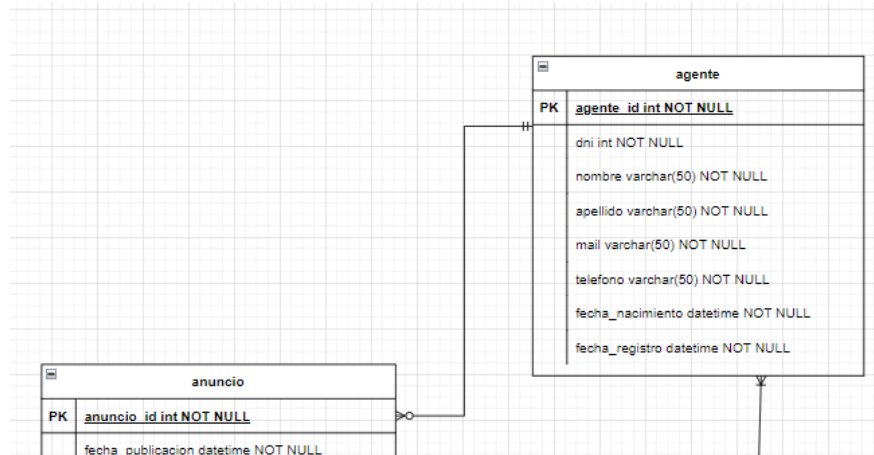


Además creemos que no aporta una cantidad excesiva de atributos a pago alquiler.

Relaciones

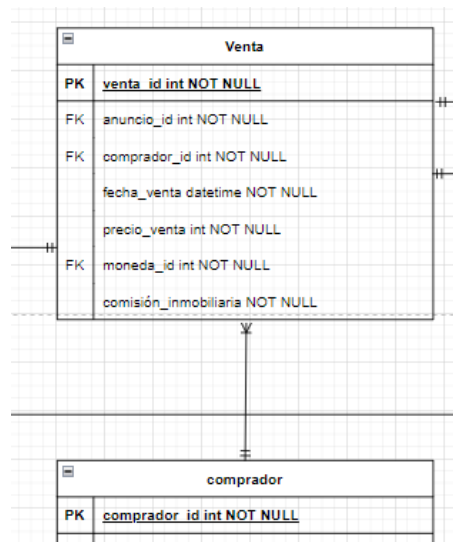
Agente a Anuncio

Decidimos realizar una relación one to many de agente a anuncio ya que un agente podría tener varios anuncios a su nombre, y determinamos que no sea obligatoria esta relación, es decir que un agente podría llegar a existir sin tener ningún anuncio a su nombre. Esta decisión la tomamos ya que pensamos que un agente podría estar inscripto en el sistema hace poco tiempo (en el caso en el que sea nuevo por ejemplo), y por lo tanto no habría concretado todavía ningún acuerdo ya sea de alquiler o venta.



Comprador a venta

Decidimos que sea obligatoria la relación de comprador a venta, pensando que no tendría sentido que exista un comprador cargado en la base de datos el cual no haya comprado nada



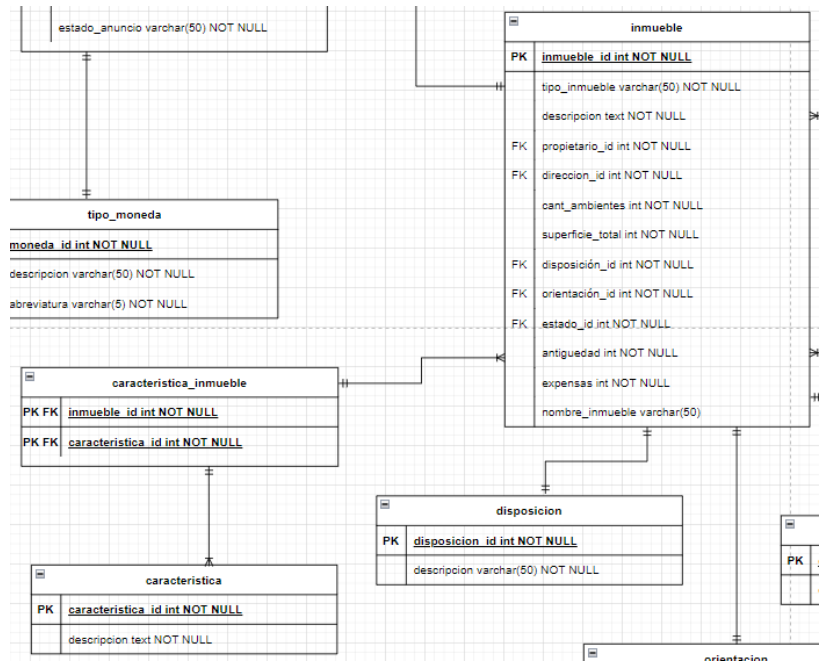
Entidades Separadas para Comprador, Agente y Propietario

Hicimos entidades aparte para comprador, agente y propietario ya que, si bien comparten todos los atributos de las tablas, cada una de ellas representa una idea independiente, por lo que no sería conveniente juntarlas todas para que una persona pueda ser comprador, agente y propietario a la vez.

Inmueble a Característica

Tomamos la decisión de hacer una relación many to many de inmueble a característica, ya que las características de los inmuebles tienen la misma estructura pero solo cambia su descripción.

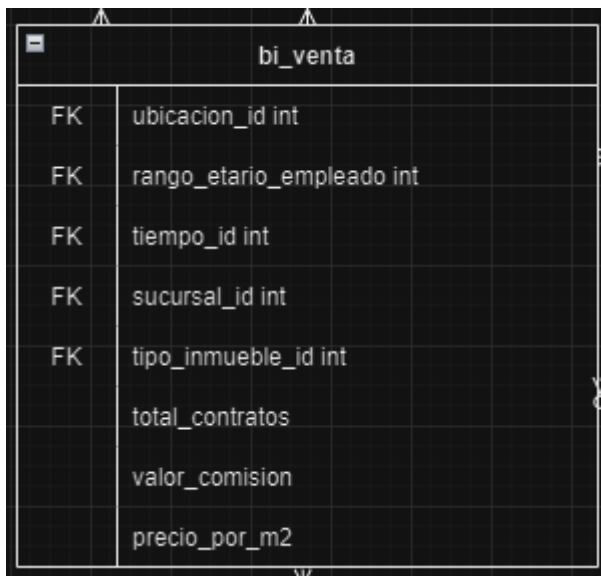
Consideramos que no tiene sentido modelarlo como columnas en inmuebles cuando se puede simplificar el diseño, legibilidad y rendimiento haciendo una relación many to many. Ya que creemos que las columnas pueden tender a confusión, poco orden en inmueble y creímos que si teníamos muchas características inmueble quedaría una entidad excesivamente grande. En este momento eso no pasa pero es posible que se agreguen características y de la forma anterior implicaría agregar muchas columnas por cada inmueble existente. Además elegimos una relación many to many debido a que un inmueble puede tener muchas características y una característica puede estar en muchos inmuebles.



Business Intelligence (BI)

Tablas de hechos

Venta

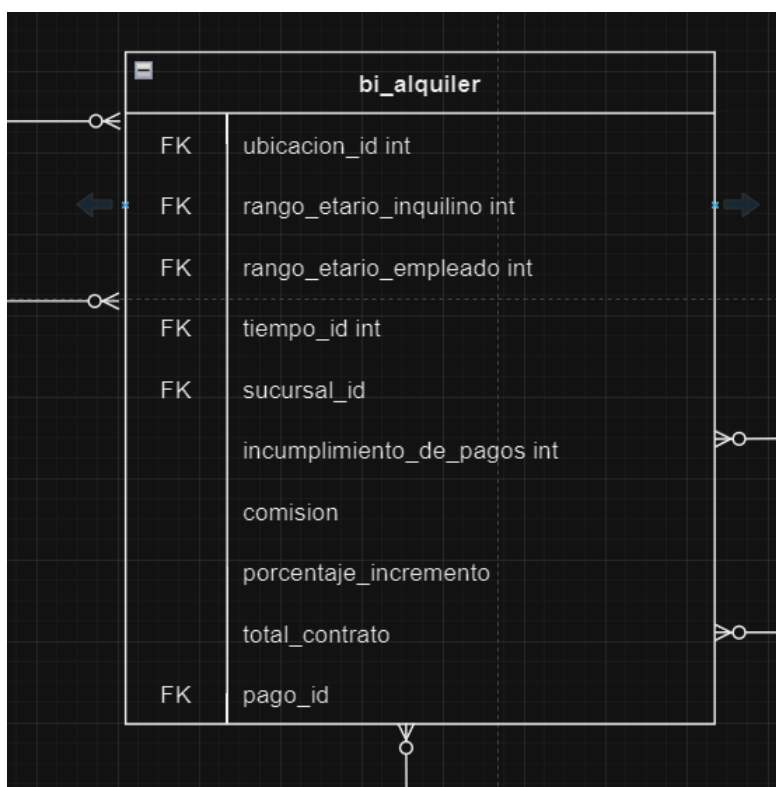


Decidimos realizar una tabla de hechos de venta debido a que se utilizaba en la mayoría de las vistas y creímos que sería necesaria.

Además en sus atributos optamos por agregar los que se utilizaban para ciertos cálculos como el precio por m2 para la vista 6. El mismo es un valor calculado usando la función para calcular el rango m2.

Luego también agregamos las distintas FOREIGN KEY con las tablas de dimensiones que necesitábamos para las vistas como el rango etario empleado para la vista 8.

Alquiler



Decidimos crear una tabla de hechos de alquiler, ya que es necesaria para la creación de las vistas y consideramos que es indispensable. Además, optamos por agregar diversos atributos, como el incumplimiento de pagos, la comisión, el porcentaje de aumento entre distintos pagos en el mismo alquiler y el precio total por el contrato de alquiler. Además de estos campos, la tabla también incluye varias claves foráneas, como la ubicación, el rango etario tanto del inquilino como del agente que realiza el contrato de alquiler,

el tiempo, la sucursal y el método de pago.

Anuncio



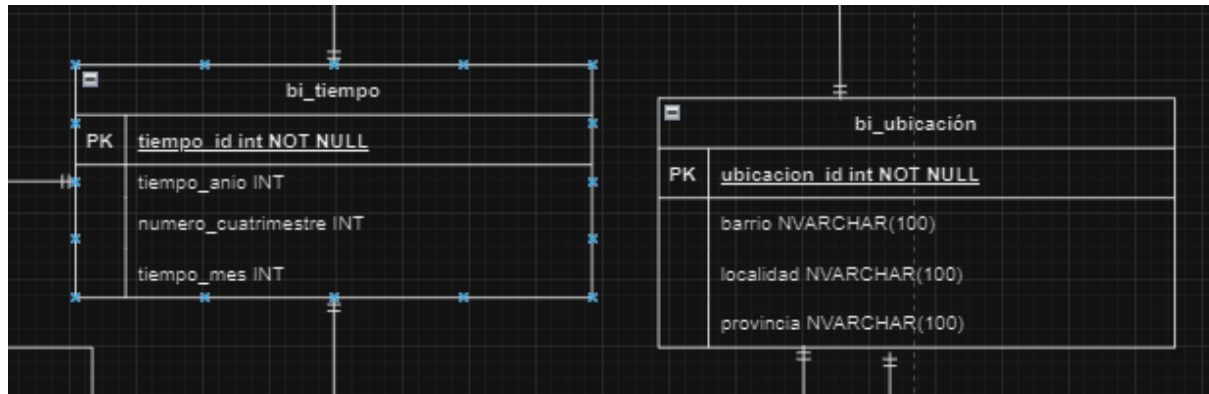
Tomamos la decisión de diseñar una tabla de hechos para la entidad "Anuncio" debido a la necesidad constante de acceder a la información relacionada con los anuncios en diversas vistas. La creación de tablas de dimensión asociadas fue impulsada por la misma razón: la reutilización de datos en varias vistas. Migramos los datos relevantes desde las tablas ya existentes en el script de creación inicial para consolidar la información de manera más eficiente.

En particular, los valores de "duracion_publicacion" y "precio_total" fueron calculados y agregados a la nueva tabla de hechos durante este proceso. Esta medida se tomó con el objetivo de

mejorar la eficiencia en la gestión de datos, permitiendo un acceso más rápido y coherente a la información clave de los anuncios en el contexto de nuestras operaciones de Business Intelligence (BI).

Justificaciones de tablas de dimensiones extra

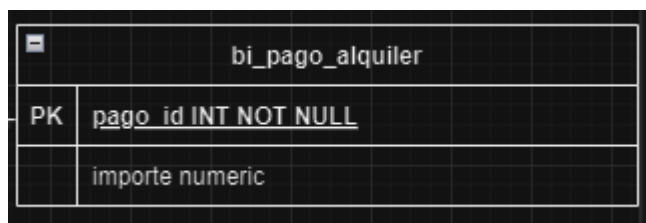
Tiempo y ubicación



Tomamos la decisión de calcular y diseñar las tablas de "Tiempo" y "Ubicación" mediante funciones especializadas. La información temporal se extrajo de la tabla original de anuncios, aprovechando las diversas fechas de publicación asociadas. En el caso de la tabla de ubicación, se llevó a cabo una migración utilizando datos provenientes de las tablas de dirección, barrio, localidad y provincia presentes en el script inicial.

Esta estrategia de diseño nos permite centralizar y organizar de manera eficiente la información temporal y geográfica, facilitando así el acceso y la utilización de estos datos en diversas vistas y análisis dentro del marco de nuestras operaciones de Business Intelligence (BI). La migración de datos desde las tablas originales asegura la consistencia y la coherencia en la representación de la información, contribuyendo a una gestión más efectiva de los datos en el contexto de nuestras necesidades analíticas.

Pago Alquiler



Agregamos a las tablas de dimensiones dadas la tabla bi_pago_alquiler debido a que necesitábamos un total de los pagos para la vista 4.