

Universidad Simón Bolívar

Departamento de Computación y Tecnología de la Información

CI-3391 Laboratorio de  
Sistemas de Bases de Datos I

# Proyecto 2

## Sistema de Subastas

Estudiantes:

Maria Magallanes 13-10787

José Barrera 15-10123

Sartenejas, diciembre de 2019

# Introducción

La competencia puede ser una forma de estimular la motivación, y es por eso que las subastas tienen tanta relevancia. Las subastas son un sistema organizado de ventas que permite que el comprador que esté dispuesto a pagar más dinero se lleve el producto, y el sistema que utilizan es la competencia.

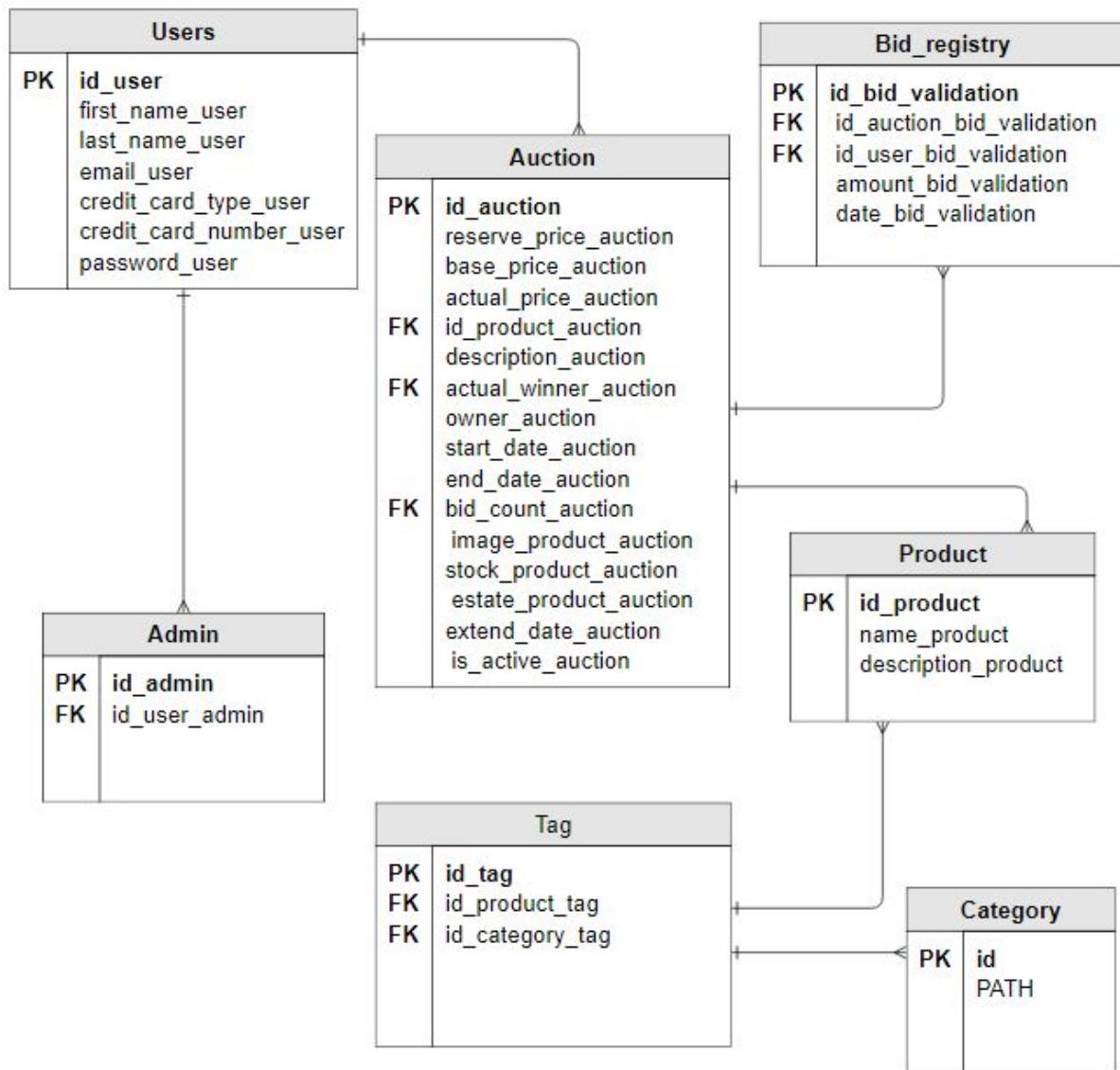
Se ha pedido que se implemente un modelo físico de una subasta. Para ello hay que tomar en cuenta varias características de las subastas y analizarlas para diseñar un modelo entidad relación que permita visualizar mejor las estructuras que proporcionarán la información para representar y manejar los datos necesarios para implementar este modelo.

Algunas de las cosas que hay que tomar en cuenta es que sólo los usuarios pueden crear subastas y estas deben tener fecha de inicio y fin, además de tener un precio base (con el que la subasta va a empezar), tener un precio de reserva (el mínimo monto que el vendedor espera recibir) y el precio actual (que se actualiza cada vez que un interesado propone una suma de dinero que supera el precio actual del producto, a esta operación se le conoce como bid). También es importante tener un registro de los productos que hay en la plataforma, tener categorías de productos y además llevar un control de los bids que se hayan realizando.

En este informe se encuentra el razonamiento detrás de cada parte del proyecto de la creación de este sistema de subasta; desde la realización del diagrama UML hasta la creación de la base de datos y la lógica detrás de las consultas que se realizan en la base de datos.

## Creación de la Base de Datos

Para este sistema de subastas se decidieron crear las relaciones Users, Admin, Auction, Bid\_registry, Category y Products. Todas relacionadas y con los atributos que se muestran en el siguiente diagrama UML:



Se realizó de la siguiente forma para tener un registro del grupo de usuarios (admins) que pueden realizar ciertas acciones especiales .

Con respecto a los productos, se consideró como producto un objeto genérico que puede tener diversas características y que puede ser subastado, y es en esta última relación en donde se colocan todos los detalles de venta de este producto. Se cuenta con tres precios: el base, el de reserva y el actual, tal como se especifica en el problema, se especifica como clave foránea el id del producto que se está subastando, se lleva registro del mayor postor, del dueño de la subasta, de las fechas de inicio y fin, y algunas características relevantes de la subasta, además del tiempo de extensión de la subasta (que se pide en este sistema).

Se tiene una tabla donde se registran todos los bids realizados en el sistema, considerando un bid una oferta en una subasta que sea válida, para ello esta debe ofrecer un precio mayor al precio actual. Sin embargo es importante resaltar que como el precio actual va cambiando se va a tener varias tuplas con el mismo id de subasta, lo que va a permitir que a la hora de que el postor o un admin decida borrar una oferta (un bid), se pueda actualizar la lista con el postor más reciente en la lista de bids que a parte debe ser quién ofrece más dinero por el producto en cuestión.

Otra relación que es importante es la de categoría ya que esta implementación se realizó en forma de árbol n-ario con ayuda de ltree, para así tener una estructura de subcategorías bien organizada, tomando en cuenta que todos los productos deben pertenecer a una categoría que no tenga una subcategoría, esta es llamada categoría hoja. Por otro lado, como los productos pueden tener varias categorías se creó la tabla tag que relaciona cada producto con alguna categoría hoja.

## Implementación de la Base de Datos y Queries

Para la realización de esta base de datos se utilizó PostgreSQL, algunas instrucciones básicas de SQL para crear tablas, hacer consultas y llenado de datos, sin embargo también se usaron Stored Procedures para considerar los casos bordes, como la concurrencia, escalabilidad y manejo de usuarios. Uno de los Stored Procedure es el de undoLastBid, procedimiento que se explicó brevemente en la parte de creación de Bases de Datos, y consta simplemente de eliminar la última oferta de bid que le han hecho a una subasta y colocar la anterior.

Otra cosa importante que nos pidieron fue consultar en promedio ¿cuáles productos tienen precios de venta más altos? ¿los productos que tienen el precio base menor a \$1 o los que empiezan con más de \$1, con la condición de que no puedo comparar productos que sean distintos. Para eso es importante entender que quiero una lista de productos ordenados por el promedio de los precios de venta de las subastas que tienen precio base <\$1 agrupados por el id del product, y sacar algo similar pero esta vez con la condición de que el precio de venta sea mayor e igual a \$1; para luego comparar estos resultados de AVG (Average) y así determinar el producto con el precio de venta más alto. Se obtuve que el precio era mayor para los de >1\$

## Para ejecutar la base de datos de este proyecto

La forma correcta de ejecutar la base de datos es entrando en la terminal y escribir:

```
> ./script_13-10787_15-10123.sh
```

Con este comando ya tendremos la base con todos los datos necesarios para hacer consultas a ella. Es importante destacar que los archivos Nomina Empleados Sartenejas.csv y Censo Empleados.csv deben estar en la misma carpeta donde se encuentre el script\_13-10787\_15-10123.sql y el script.

Los queries del proyecto están en archivos distintos con la forma: query#N\_team5.sql con N igual al número que identifica el query en cuestión.

# Conclusiones

Todo está en constante movimiento, en constante cambio, actualización, y es importante tener una base de datos primero que sea escalable, pero además hay que asegurarnos de que las reglas se cumplan después de hacer los cambios pertinentes, hay que recordar que una base de datos también es considerada como un conjunto de parámetros que se deben cumplir.

Empezando por diseñar una buena base de datos partiendo de entender los conceptos básicos de modelo entidad relación, para luego pasar a ver la base de datos como un modelo relacional y un UML, que a parte de ser sencillo y funcional, debe modelar todas las reglas o instrucciones del sistema y que siempre se cumplan estas restricciones.

Para esta parte del curso es posible que se necesiten procedimientos y funciones como el Stored Procedure, o los Triggers, que me permiten manejar situaciones, y es algo que sucede todos los días, la vida no es estática, y hay sus excepciones.