# Actividad 1 Unidad 3

Ramiro Alfonso España Tejada

Facultad de Ingeniería – Ingeniería de Software

Semestre VII: Estructura de datos

Docente John Carlos Arrieta Arrieta

Universidad de Cartagena

Noviembre, 2024

Contenido

Tabla de contenido

[Actividad 1 Unidad 3 1](#_Toc182925746)

[Introducción 4](#_Toc182925747)

[Objetivos 4](#_Toc182925748)

[De la unidad 4](#_Toc182925749)

[De la actividad 4](#_Toc182925750)

[Justificación 5](#_Toc182925751)

[Desarrollo 5](#_Toc182925752)

[1. Identificar conjunto de entidades y sus atributos: 5](#_Toc182925753)

[2. Identificar interrelaciones entre conjunto de entidades: 6](#_Toc182925754)

[3. Diseñar los respectivos Diagramas Entidad Relación (DER) 9](#_Toc182925755)

[4. Transformar el DER a Modelo Relacional (MR). 9](#_Toc182925756)

[5. Identificar los atributos (Multivaluados o Compuestos) en el DER y transformarlos en Relaciones del MR. 9](#_Toc182925757)

[6. Identificar las llaves candidatas para cada Relación en MR a partir de los atributos identificativos en el DER, luego determinar cual o cuales campos del cada Relacion del MAR serán definidos como Llave primaria 10](#_Toc182925758)

[7. Identificar las llaves foráneas en el MR, a partir de las Interrelaciones de uno a uno en el DER 13](#_Toc182925759)

[8. Identificar las llaves foráneas en el MR, a partir de las Interrelaciones de muchos a muchos en el DER 16](#_Toc182925760)

[9. Identificar las llaves foráneas en el MR, a partir de las Interrelaciones de Generalización |ISA> en DER 19](#_Toc182925761)

[10. Diseñar el respectivo Modelo Relacional (MR) a partir del DER 22](#_Toc182925762)

[11. Instalar 3 motores y servidores de bases de datos relacionales 23](#_Toc182925763)

[a. Entorno: Ambiente virtualizado (VM) sobre kvm en plataforma linux ubuntu 20.04, específicamente máquina virtual sobre ubuntu server 22.04.4 LTS (GNU/Linux 5.15.0-113-generic x86\_64) 23](#_Toc182925764)

[Enlace GIthub 29](#_Toc182925765)

[Conclusiones 29](#_Toc182925766)

[Referencias 30](#_Toc182925767)

# Introducción

En esta unidad, se trabajará en la creación de un sistema de subastas en línea, inspirado en plataformas reconocidas como eBay. El enfoque principal estará en el uso del lenguaje SQL (DDL, DML, DCL y TCL) para implementar y gestionar bases de datos. Este proyecto combina el diseño conceptual y la implementación técnica, permitiendo desarrollar habilidades para transformar un Modelo Entidad-Relación (DER) en un Modelo Relacional (MR) funcional y eficiente.

# Objetivos

## De la unidad

Aplicar el lenguaje SQL en las etapas de diseño, manipulación y control de bases de datos.

Diseñar e implementar estructuras de datos utilizando DDL, garantizando la integridad referencial y las restricciones necesarias.

Administrar la información de las bases de datos mediante comandos DML, asegurando la correcta inserción, modificación y eliminación de registros.

Configurar permisos y accesos con DCL, promoviendo un control seguro y adecuado del sistema.

Manejar transacciones con TCL para garantizar la consistencia y recuperación de datos ante errores.

## De la actividad

Identificar entidades, atributos y relaciones a partir de los requisitos de un sistema de subastas.

Diseñar el diagrama Entidad-Relación (DER) reflejando de forma precisa las interrelaciones del sistema.

Transformar el DER en un Modelo Relacional (MR), ajustando atributos compuestos, multivaluados y generalizaciones.

Implementar el MR en una base de datos utilizando comandos DDL, creando tablas y definiendo claves primarias y foráneas.

Ejecutar operaciones de manipulación de datos con DML, como agregar, actualizar y eliminar información.

Configurar usuarios y permisos mediante DCL, asegurando control y seguridad en el acceso a los datos.

Gestionar transacciones con TCL para mantener la integridad del sistema durante operaciones críticas.

# Justificación

Diseñar e implementar un sistema de subastas en línea es una excelente oportunidad para poner en práctica conocimientos sobre bases de datos y el lenguaje SQL. Este tipo de proyecto involucra entidades y relaciones complejas, como usuarios, artículos, ofertas y transacciones, lo que lo convierte en un ejercicio integral para desarrollar habilidades de análisis, modelado e implementación. Además, permite comprender cómo los conceptos teóricos se aplican a problemas reales, logrando crear sistemas eficientes que cumplen con requisitos específicos. Este enfoque refuerza tanto el dominio técnico como la capacidad de conectar los modelos conceptuales con la gestión práctica de datos en un entorno profesional.

# Desarrollo

## Identificar conjunto de entidades y sus atributos:

En el proceso de Identificar el conjunto de entidades y sus atributos se ha hecho una lectura extensa de las acciones propuestas sobre lo que puede, debe hacer la aplicación de venta en línea de tipo e-commerce usando como referencia algo similar a e-bay, mercadolibre, o cualquiera de sus estilo.

Entidades (*atributos*)

Usuario (*id\_usuario, nombre, apellidos, direccion, correo\_electronico, fecha\_registro, id\_estado\_usuario, id\_tipo\_usuario, contrasena, preferencias*)

TipoUsuario (*id\_tipo\_usuario, tipo\_usuario*)

EstadoUsuario (*id\_estado\_usuario, estado\_usuario*)

Articulo (*id\_articulo, nombre, descripcion, id\_estado\_articulo, precio\_inicial, fecha\_limite, id\_usuario, id\_categoria*)

EstadoArticulo (*id\_estado\_articulo, estado\_articulo*)

Subasta (*id\_subasta, id\_articulo, fecha\_inicio, fecha\_limite, id\_estado\_subasta, id\_usuario*)

EstadoSubasta (*id\_estado\_subasta, estado\_subasta*)

Oferta (*id\_oferta, id\_subasta, id\_usuario, precio\_ofertado, fecha\_oferta*)

Categoria (*id\_categoria, nombre*)

Comentario (*id\_comentario, id\_usuario, id\_articulo, texto\_comentario, fecha\_comentario, id\_comentario\_tipo*)

ComentarioTipo (*id\_comentario\_tipo, comentario\_tipo*)

Operacion (*id\_operacion, id\_subasta, id\_usuario, fecha\_operacion, precio\_final*)

Soporte (*id\_caso, id\_usuario, asunto, descripcion, fecha\_solicitud, id\_estado\_caso*)

EstadoCaso (*id\_estado\_caso, estado\_caso*)

Satisfaccion (*id\_satisfaccion, id\_usuario, puntuacion, comentario, fecha\_satisfaccion*)

## Identificar interrelaciones entre conjunto de entidades:

**Usuario ->---<publica>----<- Articulo:** Un usuario puede publicar varios artículos, pero cada artículo está asociado a un solo usuario.

**Usuario ->---<realiza>----<- Oferta:** Un usuario puede realizar varias ofertas, pero cada oferta está hecha por un solo usuario.

**Usuario ->---<comenta>----<- Comentario:** Un usuario puede hacer varios comentarios, pero cada comentario pertenece a un solo usuario.

**Usuario ->---<contacta>----<- Soporte:** Un usuario puede contactar al soporte varias veces, pero cada caso de soporte está asociado a un solo usuario.

**Usuario ->---<evalúa>----<- Satisfaccion:** Un usuario puede realizar varias evaluaciones de satisfacción, pero cada evaluación está asociada a un solo usuario**.**

**Usuario ->---<compra>----<- Operacion:** Un usuario puede realizar varias operaciones de compra, pero cada operación está asociada a un solo usuario.

**TipoUsuario --|---<clasifica>----<- Usuario:** Un tipo de usuario clasifica a varios usuarios, pero cada usuario tiene un único tipo de usuario.

**EstadoUsuario --|---<representa>----<- Usuario:** Un estado de usuario puede representar a varios usuarios, pero cada usuario tiene un único estado.

**Articulo ->---<partede>----<- Subasta:** Un artículo puede formar parte de una única subasta, pero una subasta puede tener varios artículos.

**Articulo --|---<clasifica>----<- EstadoArticulo:** Un estado de artículo puede clasificar a varios artículos, pero cada artículo tiene un único estado.

**Articulo --|---<pertenece>----<- Categoria:** Una categoría puede tener varios artículos, pero cada artículo pertenece a una única categoría.

**Subasta ->---<tiene>----<- Oferta:** Una subasta puede tener varias ofertas, pero cada oferta está asociada a una sola subasta.

**Subasta ->---<finalizaen>----<- Operacion:** Una subasta puede finalizar en una única operación, pero una operación está vinculada a una sola subasta.

**Subasta --|---<determina>----<- EstadoSubasta:** Un estado de subasta puede determinar varias subastas, pero cada subasta tiene un único estado.

**Subasta ->---<ganada\_por>----<- Usuario:** Una subasta es ganada por un solo usuario, pero un usuario puede ganar varias subastas.

**Oferta ->---<pertenece>----<- Subasta:** Una oferta pertenece a una sola subasta, pero una subasta puede tener varias ofertas.

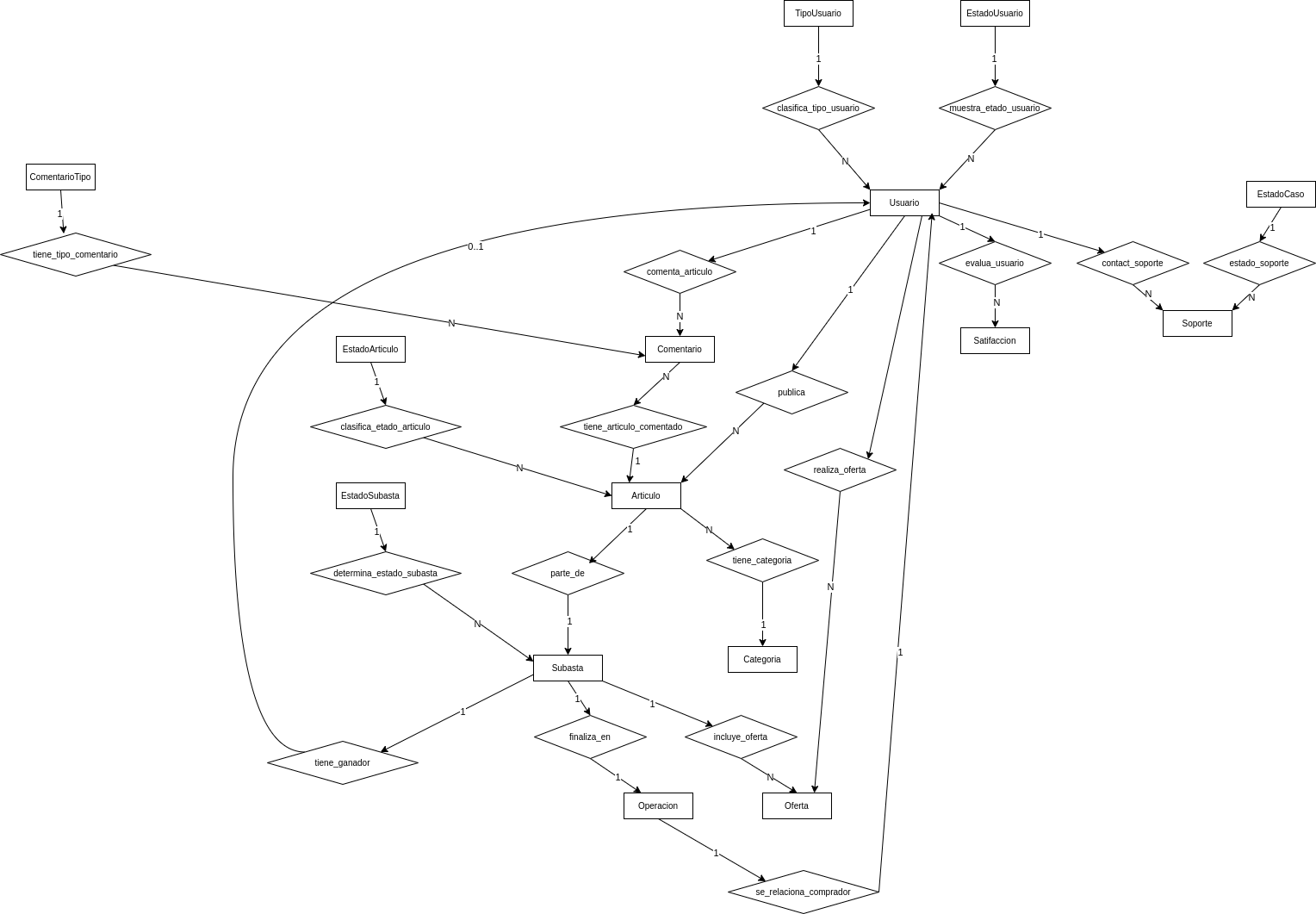
**Oferta ->---<hecha\_por>----<- Usuario:** Una oferta es hecha por un solo usuario, pero un usuario puede realizar varias ofertas.

**Comentario --|---<clasifica>----<- ComentarioTipo:** Un tipo de comentario clasifica varios comentarios, pero cada comentario tiene un único tipo.

**Comentario ->---<realizado\_sobre>----<- Articulo:** Un comentario es realizado sobre un solo artículo, pero un artículo puede tener varios comentarios.

**Soporte --|---<estado>----<- EstadoCaso:** Un estado de caso puede aplicarse a varios casos de soporte, pero cada caso tiene un único estado.

## Diseñar los respectivos Diagramas Entidad Relación (DER)



Enlace público: <https://viewer.diagrams.net/?tags=%7B%7D&lightbox=1&highlight=0000ff&edit=_blank&layers=1&nav=1&title=U1A1_DB1_DER1.drawio#Uhttps%3A%2F%2Fdrive.google.com%2Fuc%3Fid%3D1rIuc6cHLBrxkKV0doDf0zhT1--wWHilV%26export%3Ddownload>

## Transformar el DER a Modelo Relacional (MR).

## Identificar los atributos (Multivaluados o Compuestos) en el DER y transformarlos en Relaciones del MR.

En la búsqueda de los posibles atributos multivaluados o compuestos, se encuentran los siguientes candidatos:

* direccion, porque puede estar compuesto por una calle, ciudad, barrio, localidad, etc.
* preferencias, porque puede ser multivaluado; ya que maneja múltiples por no decir ilimitadas preferencias de usuario.

Para transformarlos en relación del MR, se crean nuevas relaciones para descomponer cada uno de los candidatos:

* Dirección del usuario:
  + Al ser un atributo compuesto, se crea una nueva relación para que se almacene cada parte de la dirección
    - [DIRECCION\_USUARIO] (id\_direccion, id\_usuario, calle, barrio, ciudad, departamento, país)
* Preferencias de usuario:
  + Como es un atributo multivaluado, se crea una nueva relación para que se almacenen las preferencias de cada usuario.
    - [PREFERENCIAS\_USUARIO] (id\_preferencia,id\_usuario, preferencia)

## Identificar las llaves candidatas para cada Relación en MR a partir de los atributos identificativos en el DER, luego determinar cual o cuales campos del cada Relacion del MAR serán definidos como Llave primaria

**[USUARIO] (id\_usuario**, nombre, apellidos, direccion, correo\_electronico, fecha\_registro, id\_estado\_usuario, id\_tipo\_usuario, contrasena, preferencias**)**

[TIPO\_USUARIO] (**id\_tipo\_usuario**, tipo\_usuario)

[ESTADO\_USUARIO] (**id\_estado\_usuario**, estado\_usuario)

[ARTICULO] (**id\_articulo**, nombre, descripcion, id\_estado\_articulo, precio\_inicial, fecha\_limite, id\_usuario, id\_categoria)

[ESTADO\_ARTICULO] (**id\_estado\_articulo**, estado\_articulo)

[SUBASTA] (**id\_subasta**, id\_articulo, fecha\_inicio, fecha\_limite, id\_estado\_subasta, id\_usuario)

[ESTADO\_SUBASTA] (**id\_estado\_subasta**, estado\_subasta)

[OFERTA] (**id\_oferta**, id\_subasta, id\_usuario, precio\_ofertado, fecha\_oferta)

[CATEGORIA] (**id\_categoria**, nombre)

[COMENTARIO] (**id\_comentario**, id\_usuario, id\_articulo, texto\_comentario, fecha\_comentario, id\_comentario\_tipo)

[COMENTARIO\_TIPO] (**id\_comentario\_tipo**, comentario\_tipo)

[OPERACION] (**id\_operacion**, id\_subasta, id\_usuario, fecha\_operacion, precio\_final)

[SOPORTE] (**id\_caso**, id\_usuario, asunto, descripcion, fecha\_solicitud, id\_estado\_caso)

[ESTADO\_CASO] (**id\_estado\_caso**, estado\_caso)

[SATISFACCION] (**id\_satisfaccion**, id\_usuario, puntuacion, comentario, fecha\_satisfaccion)

*Atributo compuesto -* [DIRECCION\_USUARIO] (**id\_direccion**, id\_usuario, calle, ciudad, codigo\_postal, pais)

*Atributo multivaluado -* [PREFERENCIAS\_USUARIO] (**id\_preferencia**, id\_usuario, preferencia)

* **Identificar las llaves foráneas en el MR, a partir de las Interrelaciones de uno a muchos en el DER.**

[TipoUsuario] --|---<clasifica>---|<-- [Usuario]

[EstadoUsuario] --|---<representa>---|<-- [Usuario]

[EstadoArticulo] --|---<clasifica>---|<-- [Articulo]

[Categoria] --|---<pertenece>---|<-- [Articulo]

[EstadoSubasta] --|---<determina>---|<-- [Subasta]

[EstadoCaso] --|---<estado>---|<-- [Soporte]

[ComentarioTipo] --|---<clasifica>---|<-- [Comentario]

* **Conectar la Relacion\_A con la Relacion\_B colocando la misma cardinalidad de la interrelación <IR>**

**Conexión entre TipoUsuario y Usuario**[TipoUsuario]−−∣−−−−−−−−−−−−−−−−∣<−[Usuario]

**Conexión entre EstadoUsuario y Usuario**[EstadoUsuario]−−∣−−−−−−−−−−−−−−−−∣<−[Usuario]

**Conexión entre EstadoArticulo y Articulo**[EstadoArticulo]−−∣−−−−−−−−−−−−−−−−∣<−[Articulo]

**Conexión entre Categoria y Articulo**[Categoria]−−∣−−−−−−−−−−−−−−−−∣<−[Articulo]

**Conexión entre EstadoSubasta y Subasta**[EstadoSubasta]−−∣−−−−−−−−−−−−−−−−∣<−[Subasta]

**Conexión entre EstadoCaso y Soporte**[EstadoCaso]−−∣−−−−−−−−−−−−−−−−∣<−[Soporte]

**Conexión entre ComentarioTipo y Comentario**[ComentarioTipo]−−∣−−−−−−−−−−−−−−−−∣<−[Comentario]

* **Agregar un nuevo campo al final de la lista de campos de la Relacion\_B**
* TipoUsuario y Usuario

[TipoUsuario](**id\_tipo\_usuario**,tipo\_usuario)

[Usuario](id\_usuario,nombre,apellidos,direccion,correo\_electronico,fecha\_registro,id\_estado\_usuario,**id\_tipo\_usuario**,contrasena,preferencias,*tipoUsuario\_id*)

* EstadoUsuario y Usuario

[EstadoUsuario](**id\_estado\_usuario**,estado\_usuario)

[Usuario](id\_usuario,nombre,apellidos,direccion,correo\_electronico,fecha\_registro,**id\_estado\_usuario**,id\_tipo\_usuario,contrasena,preferencias,*estadoUsuario\_id*)

* EstadoArticulo y Articulo

[EstadoArticulo](**id\_estado\_articulo**,estado\_articulo) [Articulo](id\_articulo,nombre,descripcion,**id\_estado\_articulo**,precio\_inicial,fecha\_limite,id\_usuario,id\_categoria,*estadoArticulo\_id*)

* Categoria y Articulo

[Categoria](**id\_categoria**,nombre) [Articulo](id\_articulo,nombre,descripcion,id\_estado\_articulo,precio\_inicial,fecha\_limite,id\_usuario,**id\_categoria**,*categoria\_id*)

* EstadoSubasta y Subasta

[EstadoSubasta](**id\_estado\_subasta**,estado\_subasta) [Subasta](id\_subasta,id\_articulo,fecha\_inicio,fecha\_limite,**id\_estado\_subasta**,id\_usuario,*estadoSubasta\_id*)

* EstadoCaso y Soporte

[EstadoCaso](**id\_estado\_caso**,estado\_caso) [Soporte](id\_caso,id\_usuario,asunto,descripcion,fecha\_solicitud,**id\_estado\_caso**,*estadoCaso\_id*)

* ComentarioTipo y Comentario

[ComentarioTipo](**id\_comentario\_tipo**,comentario\_tipo) [Comentario](id\_comentario,id\_usuario,id\_articulo,texto\_comentario,fecha\_comentario,**id\_comentario\_tipo**,*comentarioTipo\_id*)

## Identificar las llaves foráneas en el MR, a partir de las Interrelaciones de uno a uno en el DER

* Usuario y EstadoUsuario: En Usuario, se agrega una columna id\_estado\_usuario que actúa como llave foránea, conectando cada usuario con su respectivo estado.
  + [Usuario]--|-----<representa>-----|<-[EstadoUsuario]
* Usuario y TipoUsuario: En Usuario, se agrega una columna id\_tipo\_usuario que actúa como llave foránea, conectando cada usuario con su tipo de usuario específico.
  + [Usuario]--|-----<clasifica>-----|<-[TipoUsuario]
* **Conectar la Relacion\_A con la Relacion\_B colocando la misma cardinalidad de la interrelación <IR>**
* **Usuario y EstadoUsuario**
  + [Usuario]--|----------------|--[EstadoUsuario]
* **Usuario y TipoUsuario**
  + [Usuario]--|----------------|--[TipoUsuario]
* **Agregar un nuevo campo al final de la lista de campos de la Relacion\_A**
* Usuario y EstadoUsuario

[Usuario](**id\_usuario**, nombre, apellidos, direccion, correo\_electronico, fecha\_registro, id\_tipo\_usuario, contrasena, preferencias, *estado\_usuario\_id*)

[EstadoUsuario](**id\_estado\_usuario**, estado\_usuario, *usuario\_id*)

* Usuario y TipoUsuario

[Usuario](**id\_usuario**, nombre, apellidos, direccion, correo\_electronico, fecha\_registro, id\_estado\_usuario, contrasena, preferencias, *tipo\_usuario\_id*)

[TipoUsuario](**id\_tipo\_usuario**, tipo\_usuario, *usuario\_id*)

* **repetir el mismo procedimiento en todas las Interrelaciones de uno a uno que existan en el DER.**
* Artículo y EstadoArtículo

[Articulo](**id\_articulo**, nombre, descripcion, precio\_inicial, fecha\_limite, id\_usuario, id\_categoria, *estado\_articulo\_id*)

[EstadoArticulo](id\_estado\_articulo, **estado\_articulo**, *articulo\_id*)

* Subasta y EstadoSubasta

[Subasta](**id\_subasta**, id\_articulo, fecha\_inicio, fecha\_limite, id\_usuario, *estado\_subasta\_id*)

[EstadoSubasta](id\_estado\_subasta, **estado\_subasta**, *subasta\_id*)

* Soporte y EstadoCaso

[Soporte](**id\_caso**, id\_usuario, asunto, descripcion, fecha\_solicitud, *estado\_caso\_id*)

[EstadoCaso](id\_estado\_caso, **estado\_caso**, *caso\_id*)

* **aplicar el mismo procedimiento en todas las Interrelaciones de uno a muchos que existan en el DER.**
* Usuario y Artículo:

[Usuario](**id\_usuario**, nombre, apellidos, direccion, correo\_electronico, fecha\_registro, id\_estado\_usuario, id\_tipo\_usuario, contrasena, preferencias)

[Articulo](id\_articulo, nombre, descripcion, id\_estado\_articulo, precio\_inicial, fecha\_limite, id\_categoria, *usuario\_id*)

* Usuario y Oferta:

[Usuario](**id\_usuario**, nombre, apellidos, direccion, correo\_electronico, fecha\_registro, id\_estado\_usuario, id\_tipo\_usuario, contrasena, preferencias)

[Oferta](id\_oferta, id\_subasta, precio\_ofertado, fecha\_oferta, *usuario\_id*)

* Usuario y Comentario:

[Usuario](**id\_usuario**, nombre, apellidos, direccion, correo\_electronico, fecha\_registro, id\_estado\_usuario, id\_tipo\_usuario, contrasena, preferencias)

[Comentario](id\_comentario, id\_articulo, texto\_comentario, fecha\_comentario, id\_comentario\_tipo, *usuario\_id*)

* Usuario y Soporte:

[Usuario](**id\_usuario**, nombre, apellidos, direccion, correo\_electronico, fecha\_registro, id\_estado\_usuario, id\_tipo\_usuario, contrasena, preferencias)

[Soporte](id\_caso, asunto, descripcion, fecha\_solicitud, id\_estado\_caso, *usuario\_id*)

* Usuario y Satisfaccion:

[Usuario](**id\_usuario**, nombre, apellidos, direccion, correo\_electronico, fecha\_registro, id\_estado\_usuario, id\_tipo\_usuario, contrasena, preferencias)

[Satisfaccion](id\_satisfaccion, puntuacion, comentario, fecha\_satisfaccion, *usuario\_id*)

* Subasta y Oferta:

[Subasta](**id\_subasta**, id\_articulo, fecha\_inicio, fecha\_limite, id\_estado\_subasta, id\_usuario)

[Oferta](id\_oferta, id\_usuario, precio\_ofertado, fecha\_oferta, *subasta\_id*)

* Subasta y Operacion:

[Subasta](**id\_subasta**, id\_articulo, fecha\_inicio, fecha\_limite, id\_estado\_subasta, id\_usuario)

[Operacion](id\_operacion, id\_usuario, fecha\_operacion, precio\_final, *subasta\_id*)

* Artículo y Comentario:

[Articulo](**id\_articulo**, nombre, descripcion, id\_estado\_articulo, precio\_inicial, fecha\_limite, id\_categoria, usuario\_id)

[Comentario](id\_comentario, id\_usuario, texto\_comentario, fecha\_comentario, id\_comentario\_tipo, *articulo\_id*)

## Identificar las llaves foráneas en el MR, a partir de las Interrelaciones de muchos a muchos en el DER

* [Usuario] ->|---<Publica(id\_usuario, id\_articulo)>---|<-[Articulo]
* [Usuario] ->|---<RealizaOferta(id\_usuario, id\_oferta)>---|<-[Oferta]
* [Articulo] ->|---<Clasifica(id\_articulo, id\_categoria)>---|<-[Categoria]
* [Subasta] ->|---<ContieneOferta(id\_subasta, id\_oferta)>---|<-[Oferta]

1. Crear una nueva Relación.

* Publica(id\_usuario, id\_articulo)
* RealizaOferta(id\_usuario, id\_oferta)
* Clasifica(id\_articulo, id\_categoria)
* ContieneOferta(id\_subasta, id\_oferta)

1. Agregar un campo en la [Relacion\_IR] por cada atributo que tenga la Interrelacion <IR>, solo en el caso que <IR> tenga dichos atributos.

* No se han identificado atributos adicionales en estas IR, por lo que los campos permanecen sin modificación.

1. Agregar un nuevo campo al final de la lista de campos de la [Relacion\_IR]. Se recomienda que su nombre inicie con el nombre de la Relacion\_A y termine con la palabra id. El nuevo campo representa la llave foránea que conectará una Tupla de la Relacion\_A con otra Tupla de la Relacion\_IR. [Relacion\_IR](campo1, campo2, relacionA\_id). - Donde [F] es la llave foránea que conecta en la nueva relación

* Relación Publica (entre Usuario y Articulo)

Publica(id\_usuario, id\_articulo, articulo\_id[F])

* Relación RealizaOferta (entre Usuario y Oferta)

RealizaOferta(id\_usuario, id\_oferta, usuario\_id)

* Relación Clasifica (entre Articulo y Categoria)

Clasifica(id\_articulo, id\_categoria, articulo\_id)

* Relación ContieneOferta (entre Subasta y Oferta)

ContieneOferta(id\_subasta, id\_oferta, subasta\_id)

1. Agregar un nuevo campo al final de la lista de campos de la [Relacion\_IR]. Se recomienda que su nombre inicie con el nombre de la Relacion\_B y termine con la palabra id. El nuevo campo representa la llave foránea que conectará una Tupla de la Relacion\_B con otra Tupla de la Relacion\_IR. [Relacion\_IR](campo1, campo2, relacionA\_id, relacionB\_id).

* Relación Publica (entre Usuario y Articulo)

Publica(id\_usuario, id\_articulo, articulo\_id, usuario\_id [F])

* Relación RealizaOferta (entre Usuario y Oferta)

RealizaOferta(id\_usuario, id\_oferta, oferta\_id [F], usuario\_id)

* Relación Clasifica (entre Articulo y Categoria)

Clasifica(id\_articulo, id\_categoria, categoria\_id [F], articulo\_id)

* Relación ContieneOferta (entre Subasta y Oferta)

ContieneOferta(id\_subasta, id\_oferta, oferta\_id [F], subasta\_id)

1. La llave candidata para la [Relacion\_IR] se compone por el relacionA\_id y relacionB\_id

* Publica: (usuario\_id, articulo\_id)
* RealizaOferta: (usuario\_id, oferta\_id)
* Clasifica: (articulo\_id, categoria\_id)
* ContieneOferta: (subasta\_id, oferta\_id)

1. Determinar si la llave primera de [Relacion\_IR] efectivamente corresponde a la llave candidata, o si es necesario identificar una nueva llave candidata para que esta satisfaga el concepto de llave primaria.

La llave candidata identificada se puede usar en la relación como llave primaria de muchos a muchos, entonces las llaves primarias corresponden de manera inequívoca a las llaves primarias.

1. Conectar la Relacion\_A con la Relacion\_IR con cardinalidad de uno a muchos.

* [Usuario]--|----------------|<-[Comentario]
* [Articulo]--|----------------|<-[Comentario]
* [Subasta]--|----------------|<-[Operacion]
* [Usuario]--|----------------|<-[Operacion]
* [Subasta]--|----------------|<-[Oferta]
* [Usuario]--|----------------|<-[Oferta]
* [Subasta]--|----------------|<-[EstadoSubasta]
* [Articulo]--|----------------|<-[EstadoArticulo]
* [Usuario]--|----------------|<-[EstadoUsuario]
* [Usuario]--|----------------|<-[TipoUsuario]

1. Conectar la Relacion\_B con la Relacion\_IR con cardinalidad de uno a muchos.

* [ComentarioTipo]--|----------------|<-[Comentario]
* [Articulo]--|----------------|<-[Subasta]
* [Usuario]--|----------------|<-[Soporte]
* [EstadoSubasta]--|----------------|<-[Subasta]
* [Usuario]--|----------------|<-[Satisfaccion]

## Identificar las llaves foráneas en el MR, a partir de las Interrelaciones de Generalización |ISA> en DER

De acuerdo a la disponibilidad informativa, existen cuatro tipos principales de interrelaciones de generalización (ISA) en el modelo entidad-relación (E-R):

* Exclusiva Total: En este tipo, cada instancia de la superentidad debe pertenecer a una única subentidad, y todas las instancias de la superentidad deben estar asociadas con alguna de las subentidades. Por ejemplo, si tenemos una entidad "Empleado", podría dividirse en "Gerente" y "Asistente", donde cada empleado debe ser uno de estos dos.
* Exclusiva Parcial: Aquí, cada instancia de la superentidad se relaciona solo con una subentidad, pero no todos los ejemplares de la superentidad necesitan estar relacionados con las subentidades. Por ejemplo, en una entidad "Vehículo", podríamos tener "Coche" y "Motocicleta", pero también podrían existir vehículos que no caen en estas categorías.
* Solapada Total: En este caso, las instancias de la superentidad pueden pertenecer a más de una subentidad, y todos los ejemplares de la superentidad deben estar relacionados con al menos una subentidad. Por ejemplo, en una entidad "Persona", una persona puede ser "Estudiante" y "Trabajador" al mismo tiempo.
* Solapada Parcial: Aquí, algunas instancias de la superentidad pueden pertenecer a varias subentidades, pero no todas las instancias de la superentidad están obligadas a estar relacionadas con las subentidades. Por ejemplo, en una entidad "Animal", un animal puede ser "Perro" o "Gato", pero también podría no pertenecer a ninguna de estas categorías si es un animal salvaje.

Haciendo uso de la exclusividad total, encontramos las llaves foráneas:

* Usuario:

id\_estado\_usuario: Llave foránea que referencia a EstadoUsuario(id\_estado\_usuario).

id\_tipo\_usuario: Llave foránea que referencia a TipoUsuario(id\_tipo\_usuario).

* Articulo:

id\_usuario: Llave foránea que referencia a Usuario(id\_usuario).

id\_estado\_articulo: Llave foránea que referencia a EstadoArticulo(id\_estado\_articulo).

id\_categoria: Llave foránea que referencia a Categoria(id\_categoria).

* Subasta:

id\_articulo: Llave foránea que referencia a Articulo(id\_articulo).

id\_usuario: Llave foránea que referencia a Usuario(id\_usuario).

id\_estado\_subasta: Llave foránea que referencia a EstadoSubasta(id\_estado\_subasta).

* Oferta:

id\_subasta: Llave foránea que referencia a Subasta(id\_subasta).

id\_usuario: Llave foránea que referencia a Usuario(id\_usuario).

* Comentario:

id\_usuario: Llave foránea que referencia a Usuario(id\_usuario).

id\_articulo: Llave foránea que referencia a Articulo(id\_articulo).

id\_comentario\_tipo: Llave foránea que referencia a ComentarioTipo(id\_comentario\_tipo).

* Operacion:

id\_subasta: Llave foránea que referencia a Subasta(id\_subasta).

id\_usuario: Llave foránea que referencia a Usuario(id\_usuario).

* Soporte:

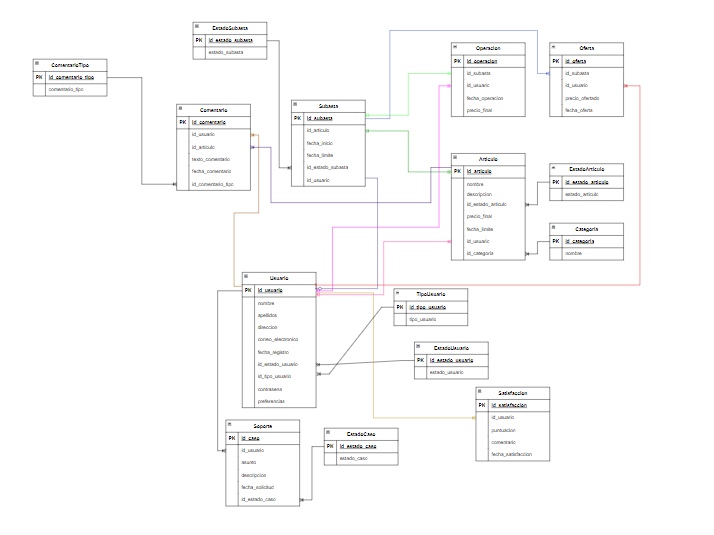
id\_usuario: Llave foránea que referencia a Usuario(id\_usuario).

id\_estado\_caso: Llave foránea que referencia a EstadoCaso(id\_estado\_caso).

* Satisfaccion:

id\_usuario: Llave foránea que referencia a Usuario(id\_usuario).

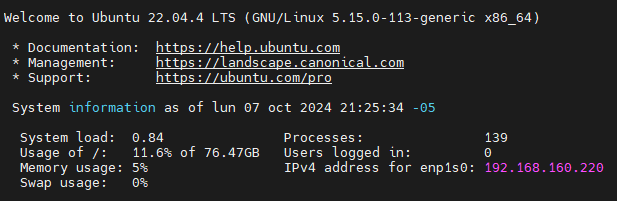
## Diseñar el respectivo Modelo Relacional (MR) a partir del DER



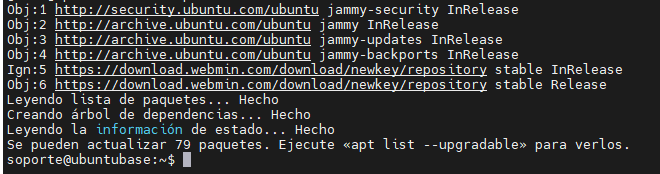
Enlace público: <https://viewer.diagrams.net/?tags=%7B%7D&lightbox=1&highlight=0000ff&edit=_blank&layers=1&nav=1&title=U1A1_DB1_MR.drawio#Uhttps%3A%2F%2Fdrive.google.com%2Fuc%3Fid%3D1e_sjrW7sYUSmSMthcVCHalKVTJcINBFT%26export%3Ddownload>

## Instalar 3 motores y servidores de bases de datos relacionales

## Entorno: Ambiente virtualizado (VM) sobre kvm en plataforma linux ubuntu 20.04, específicamente máquina virtual sobre ubuntu server 22.04.4 LTS (GNU/Linux 5.15.0-113-generic x86\_64)



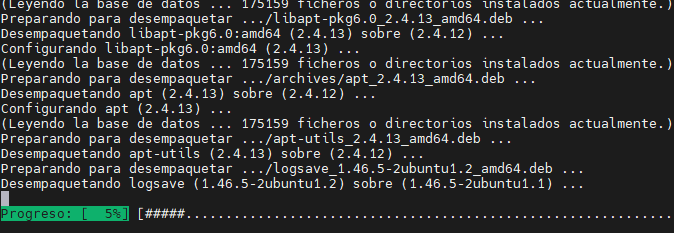
1. Procesos previos para instalación (aplica para cualquier RDBSM)
   1. *Como buena práctica de seguridad no se recomienda mantener abierta la sesión de superadministrador en entornos linux, por lo que los comandos en terminal se deben acompañar de sudo (su en debian, si no se ha configurado el paquete sudo) cuando se necesiten ejecutar comandos con privilegios elevados.*
   2. Se ejecuta la orden para actualizar la lista de actualizables disponibles



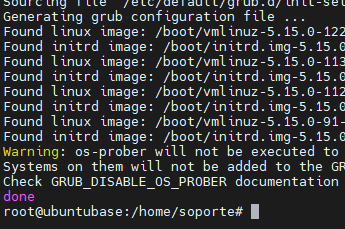
* 1. Se ejecuta la orden de actualizar los paquetes disponibles



* 1. Proceso de actualización



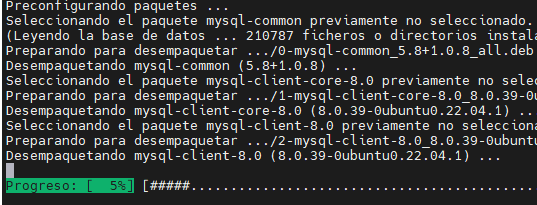
* 1. Proceso de actualización terminado

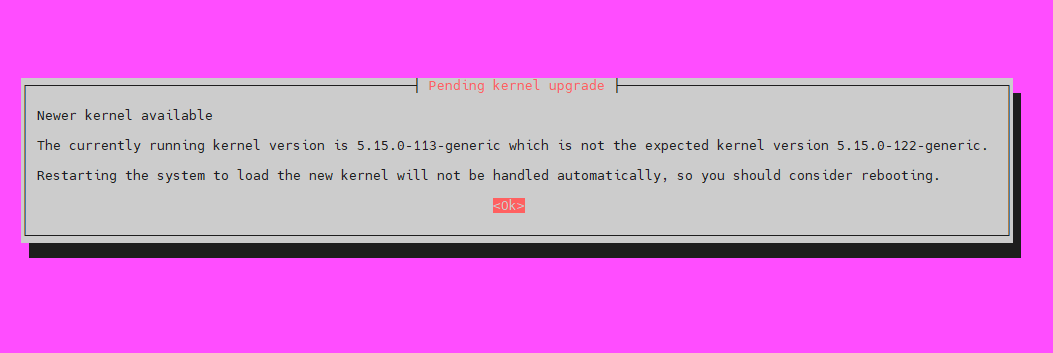


1. Instalación de MySQL
   1. Se instala el el motor junto con el cliente de mysql para efectos de validación de estado de servicios del motor y tareas automáticas

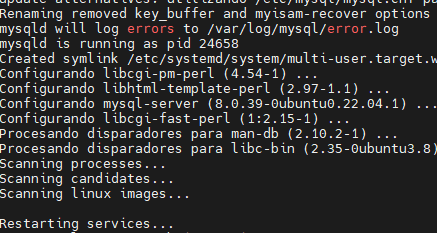


obteniendo:



Aviso de actualización de kernel

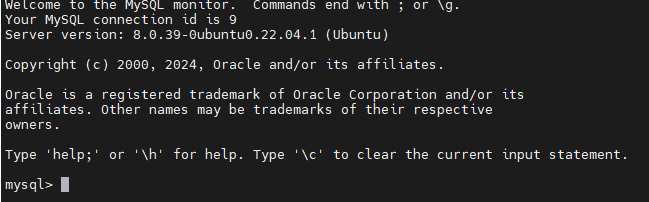
* 1. Se confirma que el motor MySQL ha sido instalado y se está ejecutando bajo el pid 24658



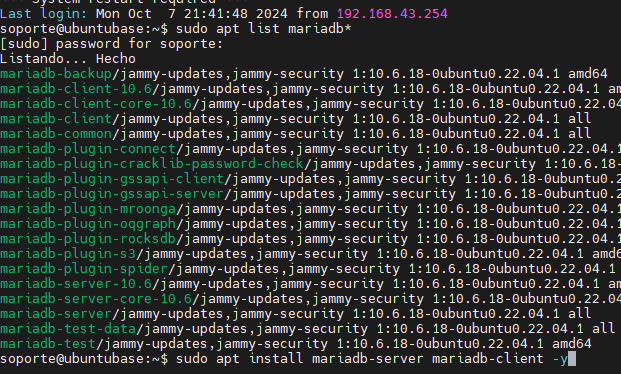
1. Al intentar iniciar mysql (como orden simple) arroja el error de acceso no autorizado, por lo que se inicia la orden mysql -u (usuario) -p (solicita la contraseña del superusuario al iniciar la consola



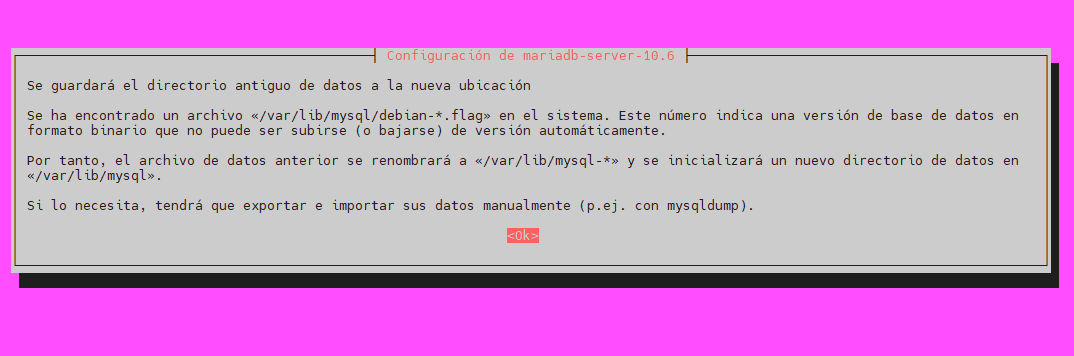
1. Validando los accesos se logra el primer ingreso a la consola del motor y se confirma la ejecución correcta del motor

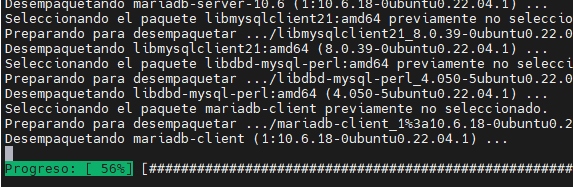


1. Instalar motor MariaDB - al igual que el motor MySQL, se instala con paquetes apt, previa lista de candidatos disponibles

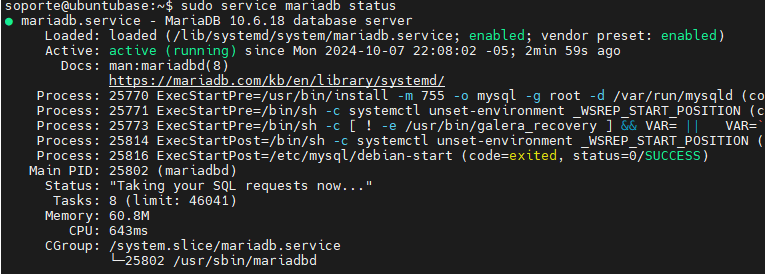


* 1. Configuración del motor post-instalación

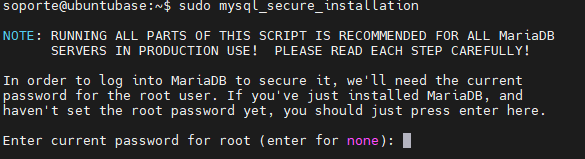




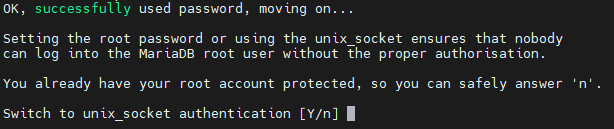
* 1. Ejecutar la orden sudo service mariadb status, nos muestra que el motor está instalado y ejecutando correctamente



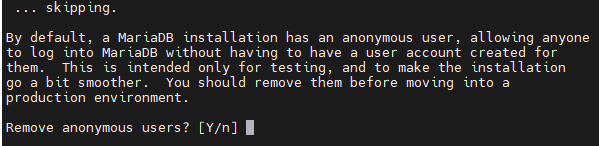
* 1. Se configura el motor para los accesos de seguridad a la consola con la orden sudo mysql\_secure\_installation



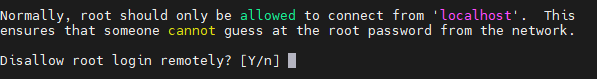
* 1. Se confirma que el usuario super administrador (root) está protegido



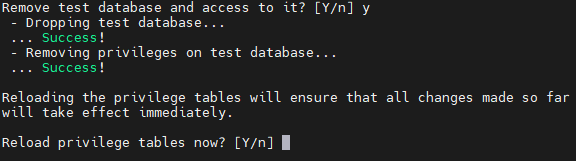
* 1. Aviso sobre usuarios anónimos que no deben estar activos en modo productivo, se procede a quitarlos



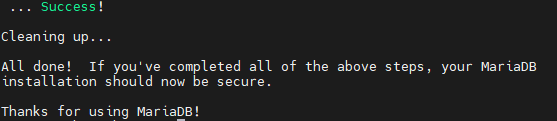
* 1. Se autoriza acceso root remoto



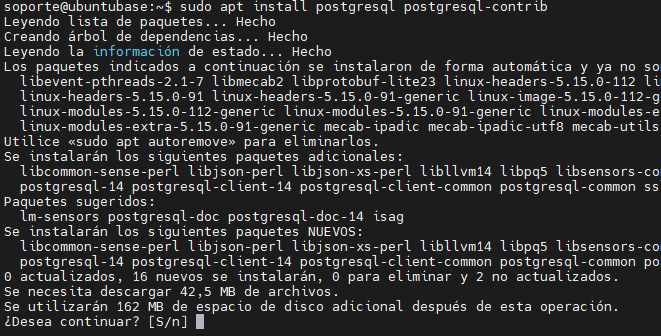
* 1. Se solicita reiniciar o recargar los privilegios para que los cambios hechos se apliquen inmediatamente.



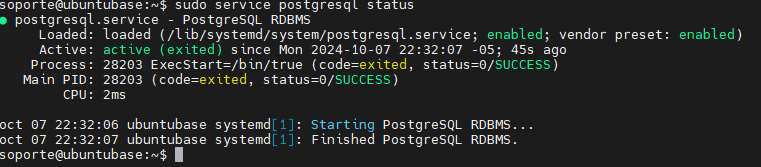
* 1. Motor instalado y configurado correctamente



1. Instalar PostgreSQL
   1. Se ejecuta la orden *sudo apt install postgresql postgresql-contrib*



* 1. Motor instalado y en ejecución



# Enlace Github

<https://github.com/respanat/UdC_Basedatos1_A1U3.git>

# Conclusiones

A lo largo de esta actividad, se han explorado las etapas fundamentales del desarrollo de un sistema de información desde el punto de vista de las bases de datos. Iniciar desde el análisis de requisitos para identificar entidades y relaciones, luego plasmamos esos conceptos en un diagrama Entidad-Relación (DER) y finalmente lo transformamos en un Modelo Relacional (MR). Se pudo evidenciar cómo un buen análisis y modelado son esenciales para construir una base de datos robusta y eficiente que pueda manejar adecuadamente las complejidades de un sistema de subastas en línea. Además, el ejercicio reforzó la importancia de contar con llaves primarias y foráneas bien definidas para garantizar la integridad de los datos y la correcta relación entre las distintas entidades. En conclusión, el desarrollo de este sistema ha permitido comprender el proceso completo, desde la abstracción conceptual hasta la concreción técnica, de un sistema de gestión de datos para una aplicación compleja y de uso cotidiano.

# Referencias

<https://gestionbasesdatos.readthedocs.io/es/latest/Tema2/Teoria.html>

<https://www.aulapc.es/lupa_busquedas_posit.html1accesA~A60.00>

<https://www.innovaciondigital360.com/big-data/rdbms-que-es-una-base-de-datos-relacional-y-cuales-son-sus-ventajas/>

<https://www.marcoteorico.com/curso/31/fundamentos-de-bases-de-datos/419/modelo-e-r-extendido->