

Semana 5

Modelamiento de Bases de Datos (PRY2204)

Formato de respuesta

| **Nombre estudiante: Luciano Rivera** |  |
| --- | --- |
| **Asignatura: Modelamiento de base de datos** | **Carrera: Desarrollador de Aplicaciones** |
| **Profesor: Josué Oteiza** | **Fecha: 09/09/2024** |

**Descripción de la actividad**

En esta quinta semana, realizarás una actividad formativa en pareja con encargo llamada "Representando el MER y sus claves primarias", en donde deberás dar solución al caso planteado desarrollando un Modelo Entidad – Relación (MER) normalizado, el que deberás transformar a modelo físico.

## Instrucciones específicas

Para realizar la actividad formativa de esta semana, continuaremos con el análisis del caso de la cadena de servicios automotrices planteado la semana pasada:

**Contexto:**

Tras identificar y modelar las entidades principales como Clientes, Vehículos, y Órdenes de Servicio, así como las relaciones para la gestión y automatización de los procesos internos de la cadena de servicios automotrices en Santiago, ahora podrás avanzar hacia la etapa de representación detallada del Modelo Entidad-Relación (MER). Este modelo será la base para desarrollar un modelo físico que servirá para la implementación de la base de datos, focalizándose en la optimización y mejora de los procesos operativos existentes.

Para llevar a cabo este nuevo proceso, te recomendamos realizar los siguientes pasos:

**Paso 1: Refinamiento de entidades y atributos**

Revisa y refina las entidades y atributos previamente identificados, asegurando que cada entidad esté claramente definida y que todos los atributos necesarios estén correctamente asignados.

**Paso 2: Revisión y asignación de claves**

Verifica y confirma la asignación de claves primarias y foráneas, centrando especial atención en garantizar la integridad referencial y la correcta relación entre las entidades.

**Paso 3: Aplicación de las Formas Normales**

Aplica de manera rigurosa las formas normales (1FN, 2FN, 3FN) para eliminar redundancias y asegurar la máxima eficiencia y coherencia en el modelo de datos.

**Paso 4: Transformación del MER a Modelo Relacional**

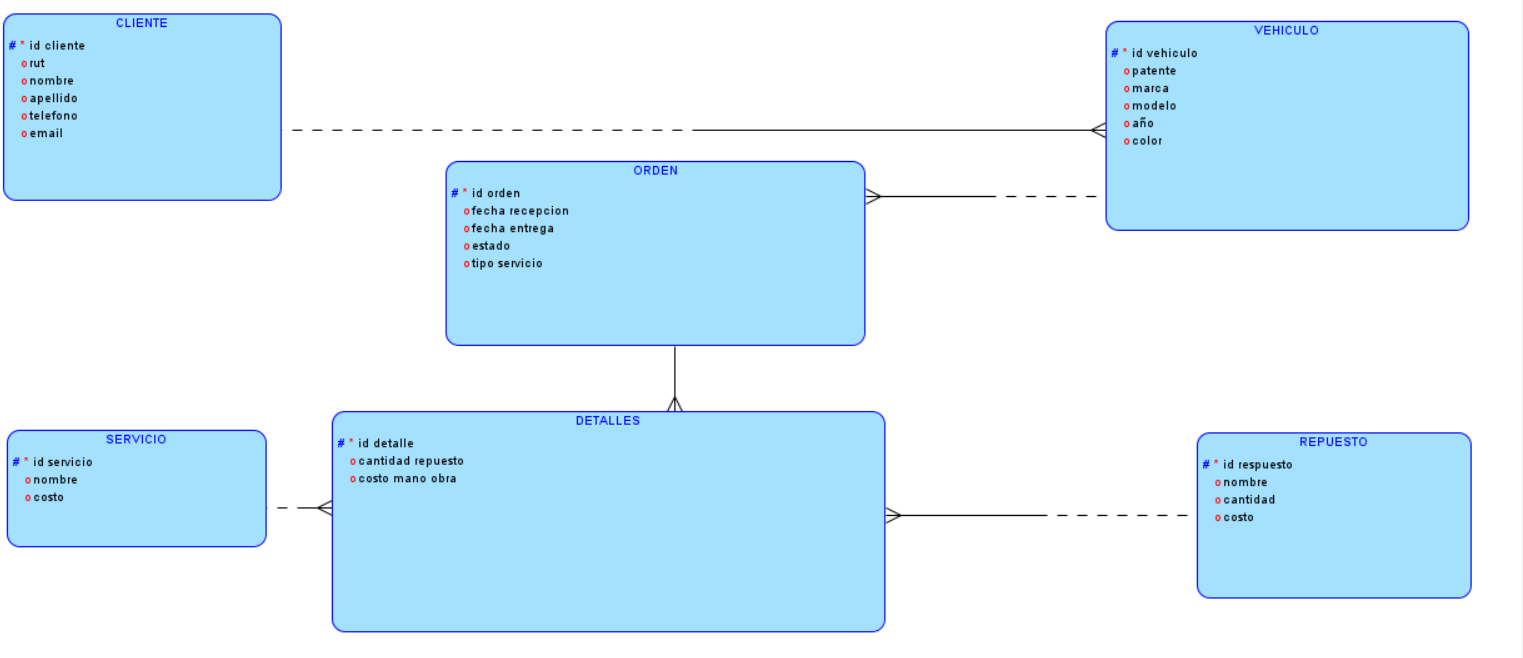
Transforma el Modelo Entidad-Relación refinado en un modelo relacional detallado, preparando la estructura para su implementación en el sistema de gestión de bases de datos.

**Paso 5:** para realizar el ejercicio, tendrás que utilizar la herramienta Oracle SQL Data Modeler, disponible de descarga a través del siguiente enlace:

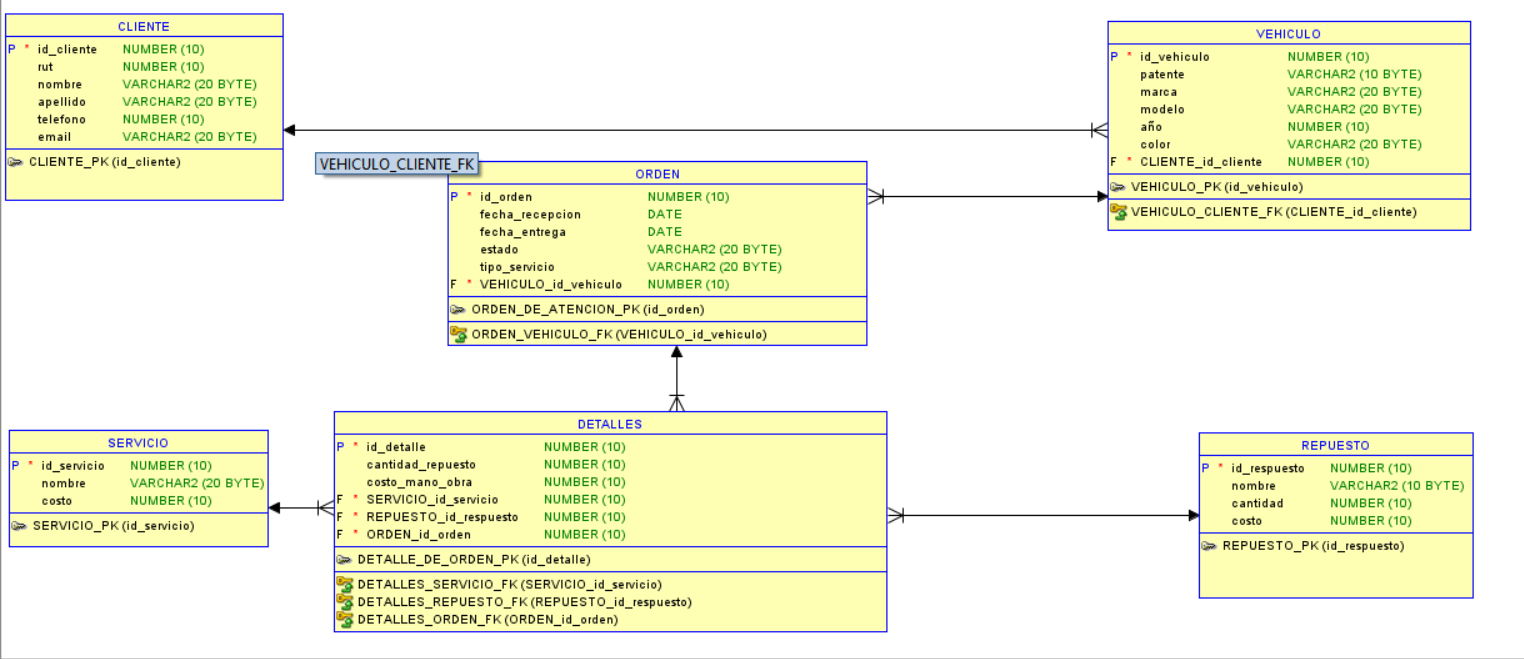
[https://www.oracle.com/database/sqldeveloper/technologies/sql-data-modeler/download/](https://www.oracle.com/database/sqldeveloper/technologies/sql-data-modeler/download/ )

Cuando tengas lista tu solución, adjunta una imagen de tu Modelo Entidad-Relación en esta parte:

Modelo lógico



Modelo físico



Solución del proyecto:

* **Entidades del Proyecto**

° CLIENTES

° VEHICULO

° ORDEN DE ATENCION

° SERVICIO

° REPUESTO

° DETALLE DE ORDEN

* **Atributos de la Entidades**

° CLIENTES

. id\_cliente = (\*) clave primaria

. rut

. nombre

. apellido

. domicilio

. teléfono

. email

° VEHICULO

. id\_vehiculo (\*) clave primaria

. patente

. marca

. modelo

. año

. color

. id\_cliente = (\*\*) clave foránea

° ORDEN\_DE\_ATENCION

. id\_orden = (\*\*) clave primaria

. fecha\_recepcion

. fecha\_entrega

. estado

. Id\_vehiculo = (\*\*) clave foránea

. tipo\_servicio

° SERVICIO

. id\_servicio = (\*) clave primaria

. nombre

. costo

° REPUESTO

. id\_repuesto (\*) clave primaria

. nombre

. cantidad

. costo

° DETALLE\_DE\_ORDEN

. id\_detalle = (\*) clave primaria

. id\_orden = (\*\*) clave foránea

. id\_servicio = (\*\*) clave foránea

. id\_repuesto = (\*\*) clave foránea

. cantidad\_repuesto

. costo\_mano\_obra

* **Asignación Claves Primarias**

. CLIENTE = id\_ cliente

. VEHICULO = id\_vehiculo

. ORDEN\_DE\_ATENCION = id\_orden

. SERVICIO = id\_servicio

. REPUESTO = id\_repuesto

. DETALLE\_DE\_ORDEN = id\_detalle

* **Establecer Relaciones**
* **1. CLIENTE - VEHICULO**
* **Relación**: Un cliente puede tener múltiples vehículos.
* **Cardinalidad**: 1 a N (Un cliente tiene 0 a N vehículos)
* **Implementación**: En la entidad **Vehículo**, el atributo **id\_cliente** es una clave foránea que apunta a **id\_cliente** en la entidad **Cliente**.
* **2. VEHICULO – ORDEN\_DE\_ATENCION**
* **Relación**: Un vehículo puede tener múltiples órdenes de atención.
* **Cardinalidad**: 1 a N (Un vehículo tiene 0 a N órdenes de atención)
* **Implementación**: En la entidad **Orden de Atención**, el atributo **id\_vehiculo** es una clave foránea que apunta a **id\_vehiculo** en la entidad **Vehículo**.
* **3. ORDEN\_DE\_ATENCION – DETALLE\_DE\_ORDEN**
* **Relación**: Cada orden de atención puede tener múltiples detalles (servicios y repuestos utilizados).
* **Cardinalidad**: 1 a N (Una orden de atención tiene 1 a N detalles)
* **Implementación**: En la entidad **Detalle de Orden**, el atributo **id\_orden** es una clave foránea que apunta a **id\_orden** en la entidad **Orden de Atención**.
* **4. SERVICIO – DETALLE\_DE\_ORDEN**
* **Relación**: Un servicio puede estar en múltiples detalles de órdenes.
* **Cardinalidad**: 1 a N (Un servicio puede aparecer en 0 a N detalles)
* **Implementación**: En la entidad **Detalle de Orden**, el atributo **id\_servicio** es una clave foránea que apunta a **id\_servicio** en la entidad **Servicio**.
* **5. REPUESTO – DETALLE\_DE\_ORDEN**
* **Relación**: Un repuesto puede estar en múltiples detalles de órdenes.
* **Cardinalidad**: 1 a N (Un repuesto puede aparecer en 0 a N detalles)
* **Implementación**: En la entidad **Detalle de Orden**, el atributo **id\_repuesto** es una clave foránea que apunta a **id\_repuesto** en la entidad **Repuesto**.
* **Determinar la Cardinalidad De Las Relaciones**

**. CLIENTE – VEHICULO = (1:N)**

**. VEHICULO – ORDEN\_DE\_ATENCION = (1:N)**

**. ORDEN\_DE\_ATENCION – DETALLE\_DE\_ORDEN = (1:N)**

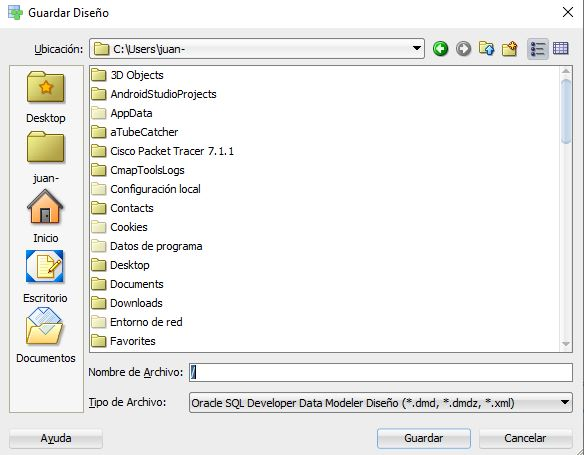
**. SERVICIO – DETALLE\_DE\_ORDEN = (1:N)**

**. REPUESTO – DETALLE\_DE\_ORDEN = (1:N)**

Además, tendrás que descargar el resultado. Para ello, tendrás que hacer clic en la opción Guardar como... del menú Archivo, esto despliega el submenú que se ilustra en la siguiente figura:

**Figura 1**

*Cómo guardar un archivo en SQL*



Nota. Ejemplo de guardado de archivo SQL. Oracle. (s.f.). *Oracle Data Modeler* [Software]. Oracle. <https://www.oracle.com/cl/database/sqldeveloper/technologies/sql-data-modeler/>

**Paso 6:** El archivo descargado desde SQL deberás subirlo al repositorio GitHub. Si no has creado tu cuenta aún, puedes hacerlo a través del siguiente enlace:

<https://github.com/>

Una vez subido el archivo a GitHub, deberás descargar el archivo comprimido .java desde tu repositorio, tal como se muestra en la imagen:

**Figura 2**

*Archivo .raw en GitHub*

Nota. Descarga de archivo desde repositorio GitHub. GitHub (s.f.). GitHub. <https://github.com/>

Posteriormente, desde el repositorio, deberás generar un enlace de tu proyecto:

**Figura 3**

Enlace de proyecto GitHub

Nota. Ejemplo de dónde se extrae un enlace en GitHub. GitHub (s.f.). *GitHub.* <https://github.com/>

Deja en este apartado el enlace de tu repositorio GitHub:

**Paso 7:** una vez adjunta tu respuesta y enlace, no olvides comprimir este documento y el archivo .raw y SQL en un archivo .rar, el cual deberás subir al AVA.



Reservados todos los derechos Fundación Instituto Profesional Duoc UC. No se permite copiar, reproducir, reeditar, descargar, publicar, emitir, difundir, de forma total o parcial la presente obra, ni su incorporación a un sistema informático, ni su transmisión en cualquier forma o por cualquier medio (electrónico, mecánico, fotocopia, grabación u otros) sin autorización previa y por escrito de Fundación Instituto Profesional Duoc UC La infracción de dichos derechos puede constituir un delito contra la propiedad intelectual.