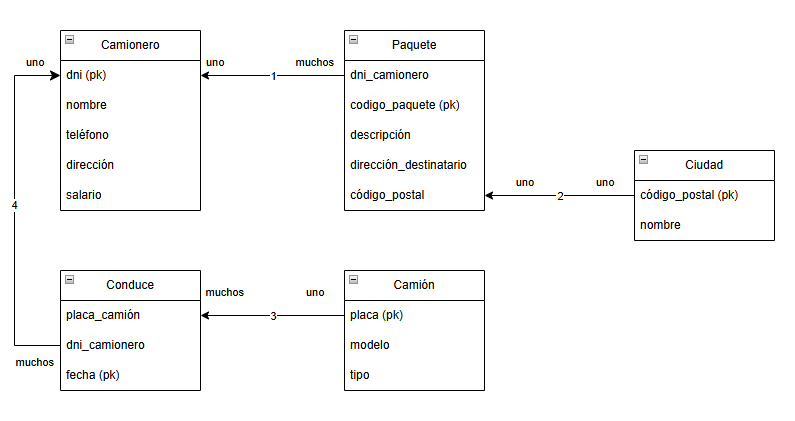
1. **Ejercicio de Cardinalidad con Modelo Lógico**

**Especificaciones del proyecto:**

* La empresa de transporte reparte paquetes por toda Guatemala.
* Los encargados de llevar paquetes son los camioneros y se quiere guardar el dni, nombre, teléfono, dirección y salario.
* De los paquetes transportados interesa conocer el código de paquete, descripción y dirección del destinatario.
* Un camionero distribuye muchos paquetes, y un paquete sólo puede ser distribuido por un camionero.
* De la ciudad a las que llegan los paquetes interesa guardar el código postal y el nombre.
* Un paquete sólo puede llegar a una ciudad. sin embargo, a una ciudad pueden llegar varios paquetes.
* De los camiones que llevan los camioneros, interesa conocer la placa, modelo y tipo.
* Un camionero puede conducir diferentes camiones en fechas diferentes, y un camión puede ser conducido por varios camioneros.



**Entidades:**

**Camionero**: dni (clave primaria), nombre, teléfono, dirección, salario.

**Paquete**: código\_paquete (clave primaria), descripción, dirección\_destinatario, dni\_camionero (clave foránea), codigo\_postal\_ciudad (clave foránea).

**Ciudad**: codigo\_postal (clave primaria), nombre.

**Camión**: placa (clave primaria), modelo, tipo.

**Conduce**: dni\_camionero (clave foránea), placa\_camion (clave foránea), fecha.

**Relaciones**:

Un Camionero distribuye muchos Paquetes, pero un Paquete solo puede ser distribuido por un Camionero (relación uno a muchos entre Camionero y Paquete).

Un Paquete solo puede llegar a una Ciudad, pero a una Ciudad pueden llegar muchos Paquetes (relación uno a muchos entre Ciudad y Paquete).

Un Camionero puede conducir diferentes Camiones en diferentes fechas, y un Camión puede ser conducido por varios Camioneros (relación muchos a muchos entre Camionero y Camión, resuelta con la tabla Conduce).

1. **Investigación e Instalación de Docker para Bases de Datos**

**¿Qué es Docker?**

Docker es una plataforma de código abierto que permite desarrollar, implementar y ejecutar aplicaciones en contenedores. Un contenedor es una unidad ligera y portátil que empaqueta una aplicación junto con todas sus dependencias (bibliotecas, frameworks, configuraciones, etc.) en un entorno aislado. Esto garantiza que la aplicación funcione de manera consistente en cualquier entorno, ya sea en desarrollo, pruebas o producción.

**Características Clave de Docker:**

**Aislamiento**: Los contenedores están aislados entre sí y del sistema anfitrión, lo que evita conflictos de dependencias.

**Portabilidad**: Los contenedores pueden ejecutarse en cualquier sistema que tenga Docker instalado, independientemente del sistema operativo.

**Eficiencia**: Los contenedores comparten el kernel del sistema operativo anfitrión, lo que los hace más ligeros y rápidos que las máquinas virtuales.

**Escalabilidad**: Docker facilita la creación y gestión de múltiples instancias de contenedores, lo que es útil para aplicaciones distribuidas.

**¿Cómo se utiliza Docker para gestionar bases de datos en entornos de desarrollo?**

Docker es especialmente útil para gestionar bases de datos en entornos de desarrollo porque permite:

Crear entornos consistentes: Los desarrolladores pueden usar la misma imagen de la base de datos en todos los entornos (desarrollo, pruebas, producción), evitando problemas de compatibilidad.

**Aislar bases de datos**: Cada contenedor de base de datos funciona de manera independiente, lo que evita conflictos entre proyectos.

**Facilitar la configuración**: Las bases de datos pueden configurarse y desplegarse rápidamente usando imágenes predefinidas (por ejemplo, MySQL, PostgreSQL, MongoDB).

**Reducir el tiempo de configuración**: En lugar de instalar y configurar manualmente una base de datos, los desarrolladores pueden usar contenedores Docker que ya están preconfigurados.

**Instalación exitosa de Docker:**

