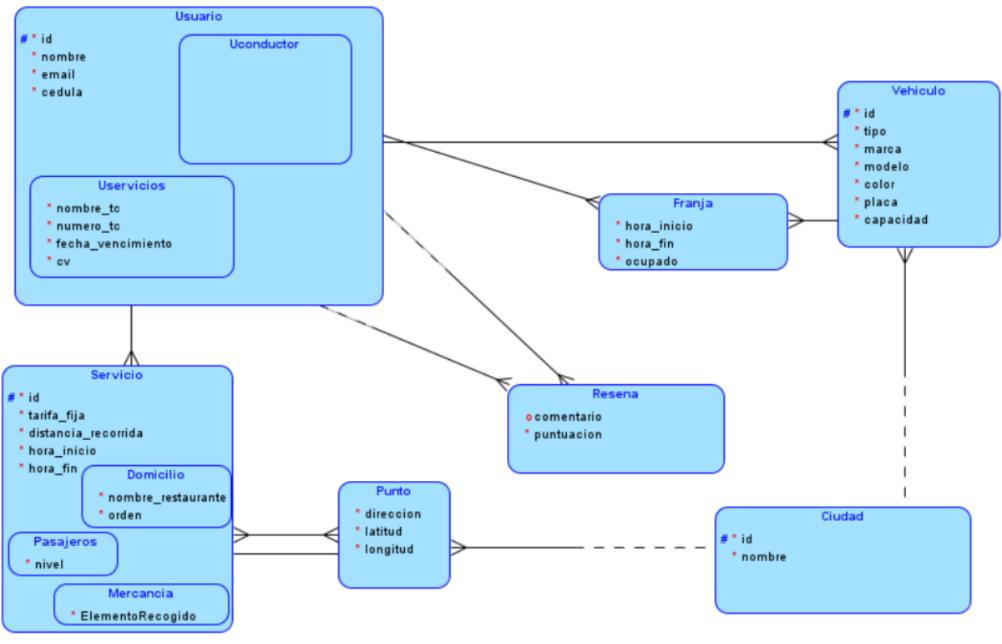


Entrega 3 – Sistemas Transaccionales

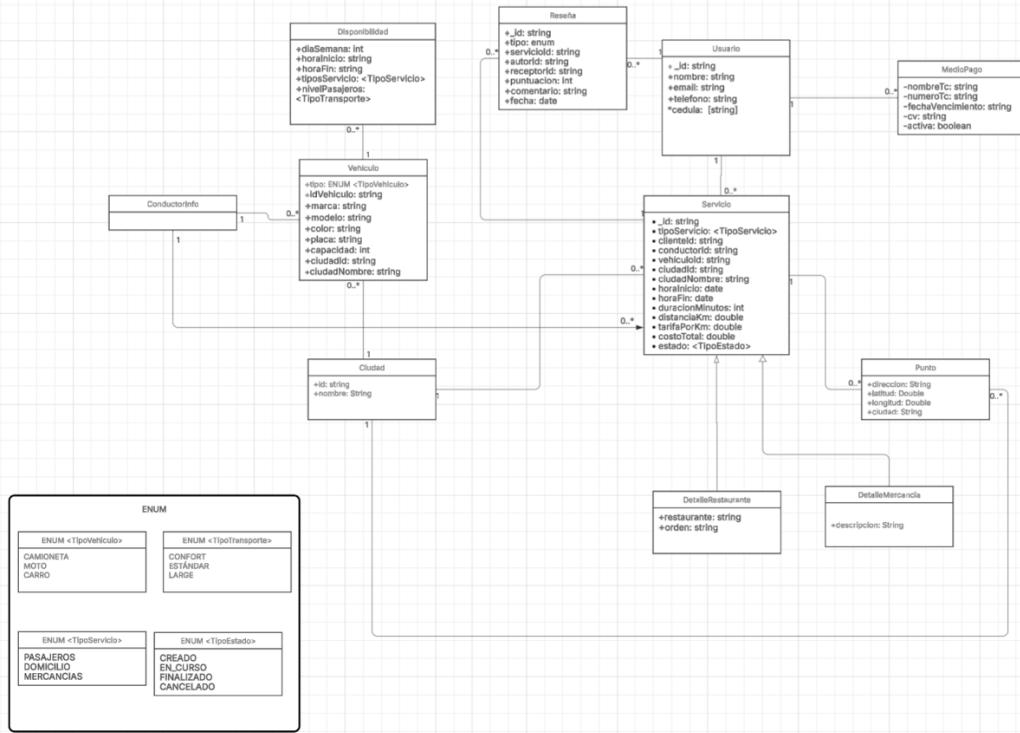
1. Diseño de la base de datos para todos los requerimientos funcionales: Describan las colecciones de datos y las relaciones entre ellas (NoSQL) que corresponden al modelo conceptual UML propuesto. Para ello, presenten lo siguiente:

- Proponga un modelo conceptual en UML o E/R que describa las entidades del modelo de datos para la aplicación que se quiere desarrollar.

El modelo entidad-relación que fue utilizado para las entregas anteriores fue el siguiente:



Sin embargo, este fue cambiado para esta entrega de la siguiente manera:



El nuevo UML diseñado representa de manera estructurada las entidades y las relaciones necesarias para poder cumplir con los requerimientos funcionales y de consulta que son solicitados por la empresa AlpesCab. En este modelo, la entidad Usuario actúa como entidad central, ya que puede tener roles de cliente o conductor, lo que se refleja con la composición con ConductorInfo y la relación con Servicio tanto. ConductorInfo también tiene una relación uno a muchos con Vehículo, ya que cada conductor puede registrar muchos vehículos y cada uno está asociado solamente a un conductor. Cada vehículo se relaciona con varias Disponibilidades, representando los días y franjas horarias en los que puede operar, pero estos no se pueden cruzar entre si. Por otro lado, la entidad Servicio conecta a clientes, conductores, vehículos y ciudades, y también incluye a la entidad Punto que funciona como el origen y los destinos del trayecto, pero también se incluyen DetalleDomicilio o DetalleMercancías, dependiendo del tipo de servicio que sea solicitado. Finalmente, el modelo tiene la entidad Reseña, que mantiene relaciones con Usuario y Servicio, permitiendo que se puedan asignar calificaciones entre clientes y conductores

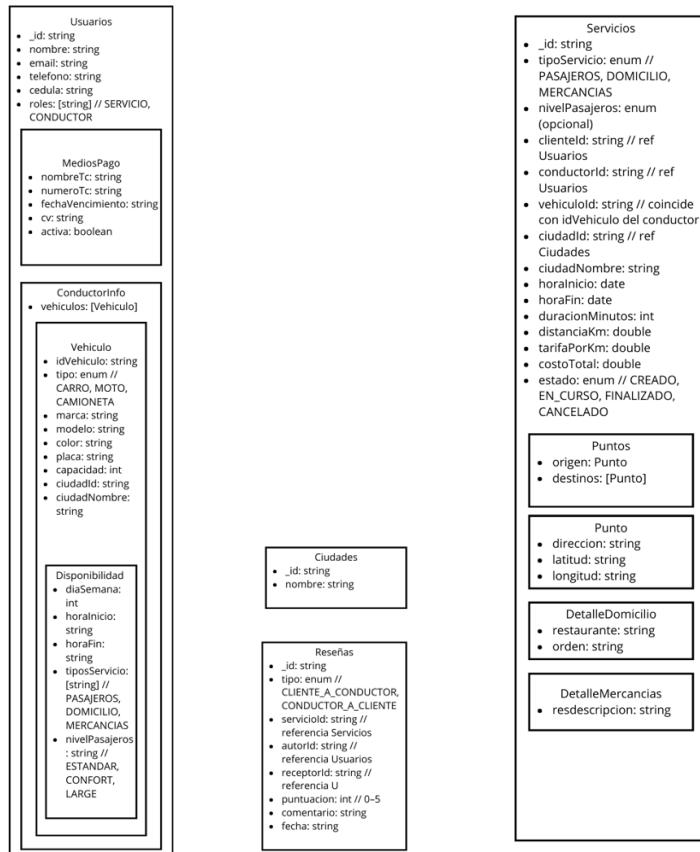
- b. La lista de entidades con la descripción de cada una de ellas, las relaciones entre entidades y su cardinalidad (uno a uno, uno a muchos, o muchos a muchos).

- Usuarios

- Entidad que representa a cualquier persona registrada en AlpesCab y puede ser un usuario de servicios o usuario conductor, o ambos. Los datos obligatorios de esta entidad son el nombre, correo, celular y cédula.
- Cardinalidad:
 - Usuario – MedioPago: Un usuario puede registrar varios métodos de pago
 - Cardinalidad: 1 a N.
 - Usuario – ConductorInfo: Un usuario puede o no tener rol de conductor.
 - Cardinalidad: 1 a 0..1.
 - UsuarioCliente – Servicio: Un usuario puede solicitar múltiples servicios.
 - Cardinalidad: 1 a N.
 - Usuario conductor – Servicio: Un conductor puede prestar muchos servicios.
 - Cardinalidad: 1 a N.
 - Usuario – Reseña: Puede generar o recibir varias reseñas.
 - Cardinalidad: 1 a N.
- MedioPago
 - Subdocumento en el que el usuario puede registrar su tarjeta de crédito: número, nombre, vencimiento y código de seguridad
 - Cardinalidad:
 - Usuario – MedioPago: Un usuario puede tener varias tarjetas, pero cada tarjeta pertenece a un único usuario.
 - Cardinalidad: 1 a N.
- ConductorInfo: Representa la información adicional que un usuario debe proporcionar para ser conductor y también incluye los vehículos que tiene registrados, así como reglas de negocio como que todo conductor debe tener mínimo un vehículo para prestar un servicio.
 - Cardinalidad:
 - ConductorInfo – Vehículo: Un conductor puede registrar varios vehículos, pero cada vehículo solo pertenece a un conductor.
 - Cardinalidad: 1 a N.
 - ConductorInfo – Servicio: El conductor puede quedar vinculado como responsable de múltiples servicios.
Cardinalidad: 1 a N.
- Vehiculo: Entidad que representa los vehículos registrados por un conductor. Tiene como atributos el tipo (carro, camioneta, moto), marca, modelo, color, placa, ciudad de expedición y capacidad. Además, la capacidad y modelo determinan el nivel del servicio (Estándar, Confort o Large).
 - Cardinalidad:
 - Conductor – Vehículo: Un conductor puede tener muchos vehículos, pero un vehículo solo tiene un conductor asociado.
Cardinalidad: 1 a N.
 - Vehículo – Disponibilidad: Cada vehículo tiene múltiples franjas de disponibilidad para distintos tipos de servicio.
 - Cardinalidad: 1 a N.
 - Vehículo – Servicio: Un vehículo puede participar en múltiples servicios.
 - Cardinalidad: 1 a N.

- Ciudad – Vehículo: La ciudad de expedición es única para el vehículo, pero una ciudad puede tener muchos vehículos.
 - Cardinalidad: 1 a N.
- Disponibilidad: Representa las franjas horarias en las que un conductor ofrece servicios con un vehículo y no puede haber disponibilidades que se crucen para un mismo conductor.
 - Cardinalidad:
 - Vehículo – Disponibilidad: Un vehículo puede tener múltiples franjas horarias.
 - Cardinalidad: 1 a N.
- Ciudad: representa ciudades donde opera la empresa
 - Cardinalidad:
 - Ciudad – Vehículo: Una ciudad tiene muchos vehículos, pero un vehículo sólo tiene asociada una ciudad.
 - Cardinalidad: 1 a N.
 - Ciudad – Servicio: Un servicio está asociado a una ciudad, pero esa ciudad puede tener muchos servicios.
 - Cardinalidad: 1 a N.
 - Ciudad – Punto: Un punto geográfico pertenece a una ciudad.
 - Cardinalidad: 1 a N.
- Servicio: Describe las solicitudes de transporte, entrega de comida o mercancías y tiene como atributos el tipo de servicio, puntos de origen y llegada, conductor asignado, vehículo utilizado, costos, tiempos, duración, distancia.
 - Cardinalidad:
 - Usuario cliente – Servicio:
 - Cardinalidad: 1 a N.
 - Usuario conductor – Servicio:
 - Cardinalidad: 1 a N.
 - Vehículo – Servicio:
 - Cardinalidad: 1 a N.
 - Servicio – Punto: Un servicio tiene un punto de origen y varios destinos.
 - Cardinalidad: 1 a N.
 - Servicio – DetalleDomicilio: Solo para servicios de comida.
 - Cardinalidad: 1 a 0..1.
 - Servicio – DetalleMercancias: Solo para transporte de mercancías.
 - Cardinalidad: 1 a 0..1.
 - Servicio – Reseña: Un servicio puede tener varias reseñas asociadas.
 - Cardinalidad: 1 a N.
- Punto: representa una ubicación en el mapa: dirección, coordenadas y ciudad.
 - Cardinalidad:
 - Servicio – Punto:
 - Cardinalidad: 1 a N.
- DetalleDomicilio: Se usa para los servicios de domicilio de comida para incluir los datos del restaurante y la orden hecha por el cliente.
 - Servicio – DetalleDomicilio:

- Cardinalidad: 1 a 0..1.
- DetalleMercancia: Se usa para servicios de transporte de mercancías para tener la descripción del ítem o paquete, peso, observaciones, etc.
 - Servicio – DetalleMercancias:
 - Cardinalidad: 1 a 0..1.
- Reseña: representa las calificaciones entre usuarios tras un servicio,
 - Servicio – Reseña:
 - Cardinalidad: 1 a N.
 - Usuario – Reseña:
 - Cardinalidad: 1 a N.
 - Usuario – Reseña:
 - Cardinalidad: 1 a N.
- c. El análisis de selección de esquema de asociación (modelo normalizado o embebido) para cada relación entre entidades, justificado a partir de los requerimientos de la aplicación y un análisis de la carga de trabajo descritas antes.



El modelo embebido fue hecho con base a los requerimientos que fueron propuestos por AlpesCab y se utilizan modelos embebidos en su mayoría debido a que las entidades del modelo están fuertemente asociadas y se consultan casi siempre de forma conjunta. Primero que todo, los métodos de pago están embebidos dentro de la colección de usuarios debido a que esta información hace parte del perfil del usuario, se actualizan pocas veces al año y son consultadas siempre en

conjunto con el usuario, entonces no tendría sentido tener estas dos entidades por separado. La información del conductor también se embebe dentro de la colección de usuarios dado a que este rol es simplemente una extensión del mismo y no tiene sentido fuera de él. Los vehículos también se embeben dentro del conductor porque cada vehículo pertenece a un solo conductor, no se comparte entre entidades, su número promedio es bajo y sus actualizaciones son poco frecuentes, lo cual cumple con los criterios de simplicidad, atomicidad y ausencia de duplicación. Las disponibilidades de cada vehículo también se embeben dentro de estos porque tienen atributos del vehículo esenciales para su operación y se requieren siempre en conjunto para determinar si un servicio puede ser asignado. Para la colección de Servicios, los puntos de origen y destino se embeben porque solo existen cuando hay un viaje y deben consultarse junto al resto de la información del servicio. DetalleDomicilio y DetalleMercancías también se embeben debido a que solo tienen sentido dentro del servicio que los contiene y no necesitan estar por separado.

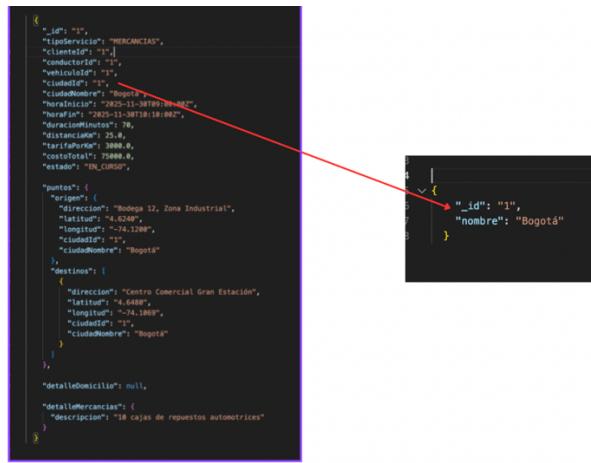
- d. Una descripción gráfica usando Json de cada relación entre entidades en donde presente un ejemplo de datos junto con el esquema de asociación usado (referenciado o embebido).

Para la colección de Usuarios, se ve el siguiente ejemplo del Json en el que se observan las relaciones entre entidades dónde vemos que hay un arreglo embebido de los medios de pagos donde se ve que hay más de dos tarjetas de crédito asociadas a la usuaria 1. Adicionalmente, se puede observar como también está el arreglo embebido de conductorInfo que también tiene embebido respectivamente todos los vehículos que están asociadas al usuario y las disponibilidades que tiene. Como fue mencionado anteriormente, esto se hace para que la consulta sea más fácil y se pueda obtener de manera más eficiente toda la información asociada a una sola usuaria.

```
{  
  "_id": "1",  
  "nombre": "Vale Rueda",  
  "email": "vale.rueda@gmail.com",  
  "telefono": "3001234567",  
  "cedula": "1012345678",  
  "roles": ["SERVICIO", "CONDUCTOR"],  
  
  "mediosPago": [  
    {  
      "nombreTc": "Vale Rueda",  
      "numeroTc": "4111111111111111",  
      "fechaVencimiento": "12/28",  
      "cv": "123",  
      "activa": true  
    },  
    {  
      "nombreTc": "Vale Rueda",  
      "numeroTc": "5500000000000004",  
      "fechaVencimiento": "07/27",  
      "cv": "456",  
      "activa": false  
    }  
  ],  
  
  "conductorInfo": {},  
  "vehiculos": [  
    {  
      "idVehiculo": "VEH001",  
      "tipo": "CARRO",  
      "marca": "Toyota",  
      "modelo": "Corolla 2020",  
      "color": "Blanco",  
      "placa": "ABC123",  
      "capacidad": 4,  
      "ciudadId": "1",  
      "ciudadNombre": "Bogotá",  
  
      "disponibilidad": [  
        {  
          "diaSemana": 1,  
          "horaInicio": "06:00",  
          "horaFin": "10:00",  
          "tiposServicio": ["PASAJEROS", "DOMICILIO"],  
          "nivelPasajeros": "ESTANDAR"  
        },  
        {  
          "diaSemana": 1,  
          "horaInicio": "17:00",  
          "horaFin": "21:00",  
          "tiposServicio": ["PASAJEROS"],  
          "nivelPasajeros": "CONFORT"  
        }  
      ]  
    }  
  ]  
}
```

Lo mismo sucede para la colección de servicios en el que se guarda toda la información asociada a servicio, como el id del vehículo, el id del cliente, el id del conductor, el id de la ciudad, pero estas son referencias a los otros. Sin embargo, ya se tienen embebidos directamente en el Json los diferentes puntos a los que debe llegar el respectivo servicio y el detalle de estas (ya sea si un servicio de mercancía o de domicilio). Esto se hace para facilitar la consulta, pero con las referencias no es necesario embeber toda la información, ya que esto puede causar réplicas de los datos y complejizar la base de datos.

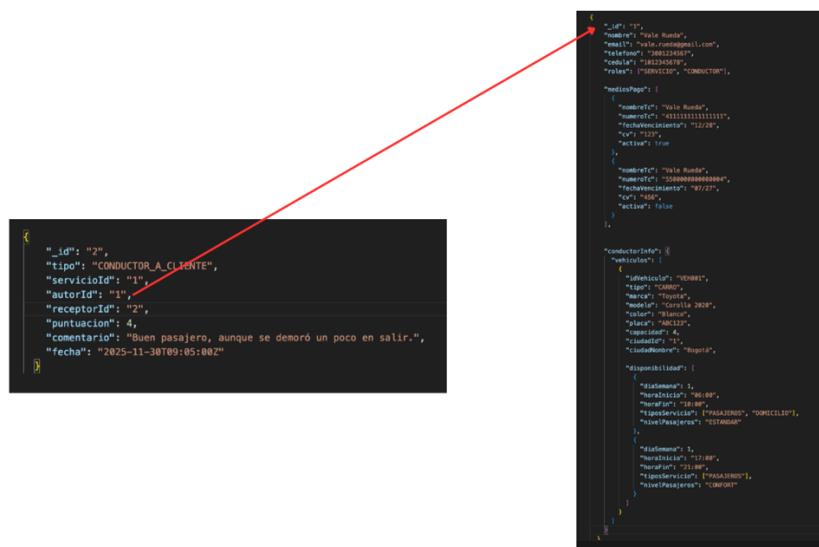
```
{  
  "_id": "1",  
  "tipoServicio": "MERCANCIAS",  
  "clienteId": "1",  
  "conductorId": "1",  
  "vehiculoId": "1",  
  "ciudadId": "1",  
  "ciudadNombre": "Bogotá",  
  "horaInicio": "2025-11-30T09:00:00Z",  
  "horaFin": "2025-11-30T10:10:00Z",  
  "duracionMinutos": 70,  
  "distanciaKm": 25.0,  
  "tarifaPorKm": 3000.0,  
  "costoTotal": 75000.0,  
  "estado": "EN_CURSO",  
  
  "puntos": {  
    "origen": {  
      "direccion": "Bodega 12, Zona Industrial",  
      "latitud": "4.6240",  
      "longitud": "-74.1200",  
      "ciudadId": "1",  
      "ciudadNombre": "Bogotá"  
    },  
    "destinos": [  
      {  
        "direccion": "Centro Comercial Gran Estación",  
        "latitud": "4.6480",  
        "longitud": "-74.1069",  
        "ciudadId": "1",  
        "ciudadNombre": "Bogotá"  
      }  
    ],  
    "detalleDomicilio": null,  
    "detalleMercancias": {  
      "descripcion": "10 cajas de repuestos automotrices"  
    }  
  }  
}
```



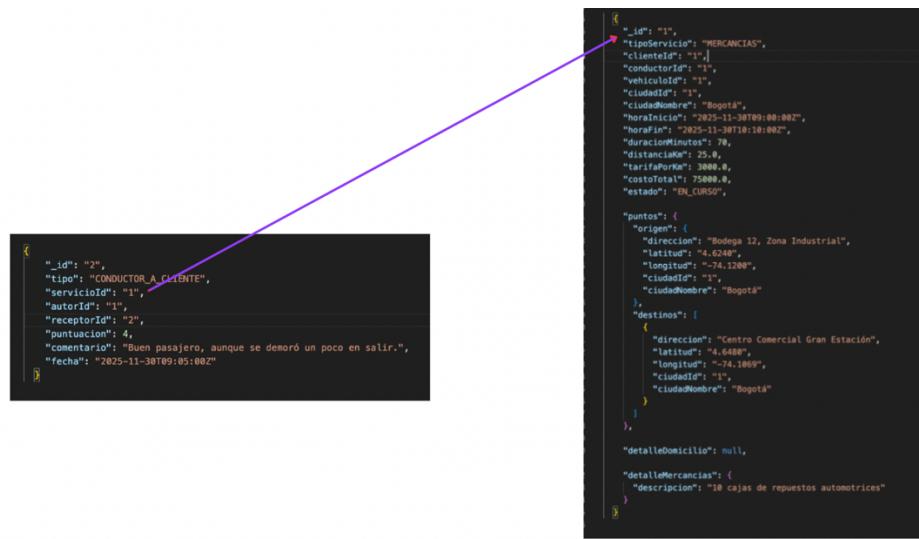
```
{  
  "_id": "1",  
  "tipoServicio": "MERCANCIAS",  
  "clienteId": "1",  
  "conductorId": "1",  
  "vehiculoId": "1",  
  "ciudadId": "1",  
  "ciudadNombre": "Bogotá",  
  "horaInicio": "2025-11-30T09:00:00Z",  
  "horaFin": "2025-11-30T10:10:00Z",  
  "duracionMinutos": 70,  
  "distanciaKm": 25.0,  
  "tarifaPorKm": 3000.0,  
  "costoTotal": 75000.0,  
  "estado": "EN_CURSO",  
  
  "puntos": {  
    "origen": {  
      "direccion": "Bodega 12, Zona Industrial",  
      "latitud": "4.6240",  
      "longitud": "-74.1200",  
      "ciudadId": "1",  
      "ciudadNombre": "Bogotá"  
    },  
    "destinos": [  
      {  
        "direccion": "Centro Comercial Gran Estación",  
        "latitud": "4.6480",  
        "longitud": "-74.1069",  
        "ciudadId": "1",  
        "ciudadNombre": "Bogotá"  
      }  
    ],  
    "detalleDomicilio": null,  
    "detalleMercancias": {  
      "descripcion": "10 cajas de repuestos automotrices"  
    }  
  }  
}
```

En el siguiente Json, se puede observar más claramente cómo se verían los datos cuando se usan netamente referencias. En este caso, para una reseña no es necesario embeber toda la información de un cliente dentro de la reseña, ya que, al fin y al cabo, lo importante de la consulta de una reseña es su contenido. Debido a esto, sólo se hace una simple referencia hacia los usuarios que están haciendo la reseña y las que lo reciben para que queden relacionados entre sí y que se pueda consultar esta información, pero no es necesario incluir toda la información del usuario y su servicio.

```
{  
  "_id": "2",  
  "tipo": "CONDUCTOR_A_CLIENTE",  
  "servicioId": "1",  
  "autorId": "1",  
  "receptorId": "2",  
  "puntuacion": 4,  
  "comentario": "Buen pasajero, aunque se demoró un poco en salir.",  
  "fecha": "2025-11-30T09:05:00Z"  
}
```



```
{  
  "_id": "2",  
  "tipo": "CONDUCTOR_A_CLIENTE",  
  "servicioId": "1",  
  "autorId": "1",  
  "receptorId": "2",  
  "puntuacion": 4,  
  "comentario": "Buen pasajero, aunque se demoró un poco en salir.",  
  "fecha": "2025-11-30T09:05:00Z"  
}  
  
{  
  "_id": "2",  
  "nombre": "Valle Rueda",  
  "email": "valle.rueda@gmail.com",  
  "telefono": "+573223456789",  
  "correo": "valle.rueda@gmail.com",  
  "roles": ["SERVICIO", "CONDUCTOR"],  
  "medioPago": [  
    {"nombre": "Transferencia",  
     "numeroCt": "1111111111111111",  
     "fechaVencimiento": "12/28",  
     "accion": true},  
    {"nombre": "Bogota",  
     "numeroCt": "1111111111111111",  
     "fechaVencimiento": "07/27",  
     "accion": false},  
  ],  
  "condiciones": [  
    {"domicilio": "10001",  
     "tipo": "CALLE",  
     "marca": "Toyota",  
     "modelo": "Corolla 2020",  
     "color": "Blanco",  
     "placa": "ABC123",  
     "id": 1, "ciudadId": "1",  
     "ciudadNombre": "Bogotá"},  
    {"domicilio": 1, "horaInicio": "06:00",  
     "horaFinal": "06:00",  
     "tipoService": "PASAJERO", "domicilio": 1, "ciudad": "Bogotá"},  
    {"domicilio": 1, "horaInicio": "17:00",  
     "horaFinal": "23:00",  
     "tipoService": "PASAJERO", "ciudad": "Bogotá"}  
  ]  
}
```



```
{  
  "_id": "2",  
  "tipo": "CONDUCTOR_A_CLIENTE",  
  "servicioId": "1",  
  "autorId": "1",  
  "receptorId": "2",  
  "puntuacion": 4,  
  "comentario": "Buen pasajero, aunque se demoró un poco en salir.",  
  "fecha": "2025-11-30T09:05:00Z"  
}  
  
{  
  "_id": "1",  
  "tiposervicio": "MERCANCIAS",  
  "clienteId": "1",  
  "conductorId": "1",  
  "vehiculoId": "1",  
  "ciudadId": "1",  
  "ciudadNombre": "Bogotá",  
  "horaInicio": "2025-11-30T09:00:00Z",  
  "horaFinal": "2025-11-30T09:00:00Z",  
  "duracionMaxima": 0, "distanciaMax": 25.0,  
  "tarifaPorKm": 3000.0, "costoTotal": 75000.0,  
  "estado": "EN_CURSO",  
  "puntos: {  
    "origen": {  
      "direccion": "Calle 12, Zona Industrial",  
      "latitud": "4.5234",  
      "longitud": "-74.1208",  
      "ciudadId": "1",  
      "ciudadNombre": "Bogotá"},  
    "destinos": [  
      {"direccion": "Centro Comercial Gran Estación",  
       "latitud": "4.5408",  
       "longitud": "-74.1869",  
       "ciudadId": "1",  
       "ciudadNombre": "Bogotá"}  
    ]  
  },  
  "detalleDomicilio": null,  
  "detalleMercancias": {  
    "descripcion": "10 cajas de repuestos automotrices"  
  }  
}
```

Valentina Rueda 202022052

Juan Eduardo Ballesteros 202221829