

| | |
|------------------------|--------------------------------|
| Compañía: | Necomplus |
| Aspirante: | Luis Ángel Rodríguez Hernández |
| Prueba: | Prueba Ingreso Backend Java |
| Fecha de Presentación: | 20 de agosto de 2024 |

Tecnologías empleadas

| | |
|--------------------------------------|---|
| Motor de base de datos relacionales: | Microsoft SQL Server 2022 |
| Lenguaje de programación: | Java 21 |
| Mapeador objeto-relacional: | Hibernate 6.3.1 |
| Entorno de desarrollo: | IntelliJ IDEA 2024.2.0.1 (Ultimate Edition) |
| Marco de pruebas unitarias: | JUnit 5 |

Estructura de la base de datos

La estructura de la base de datos consiste en dos tablas: *Vehiculo* y *Estancia*, las cuales se encuentran relacionadas con cardinalidad 1 a muchos, es decir que, un vehiculo puede tener varias estancias. Lo anterior, ocurre cuando un vehículo de residentes contiene muchas estancias durante un mes.

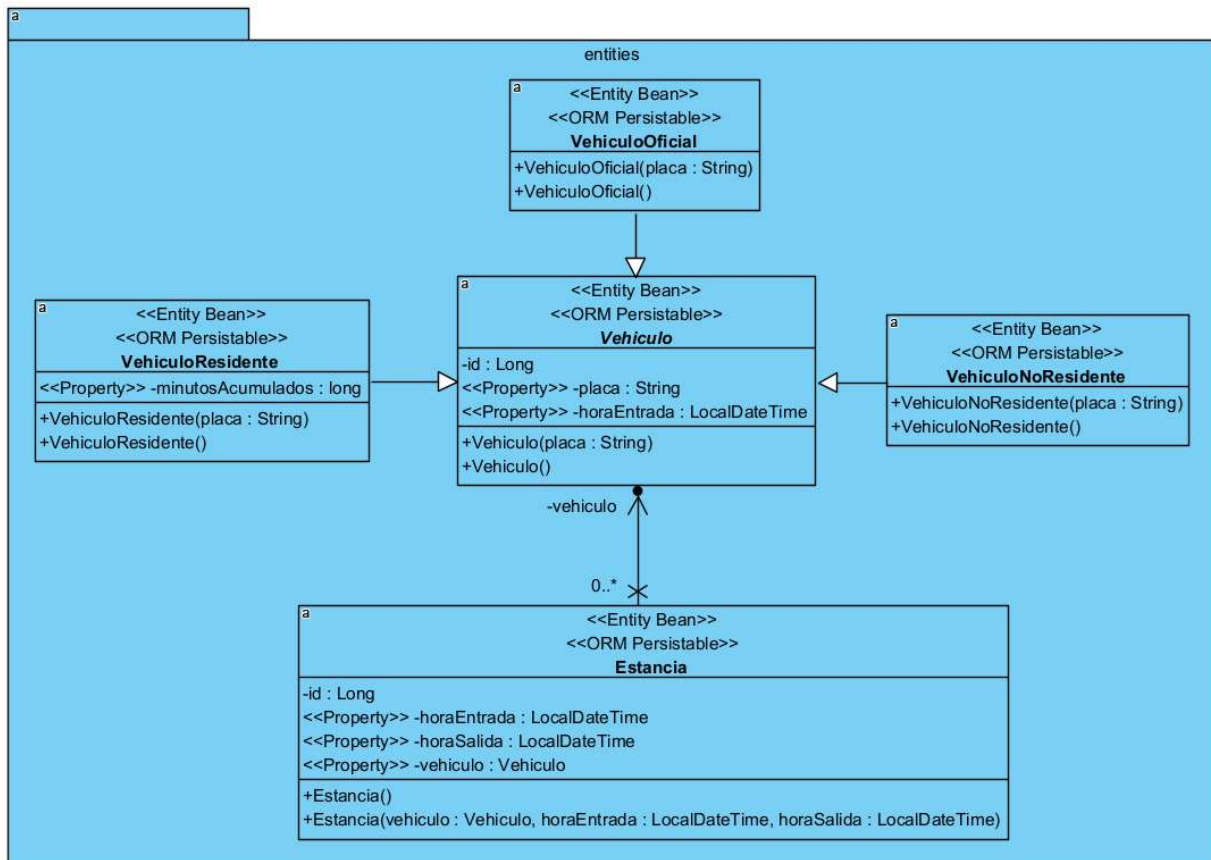
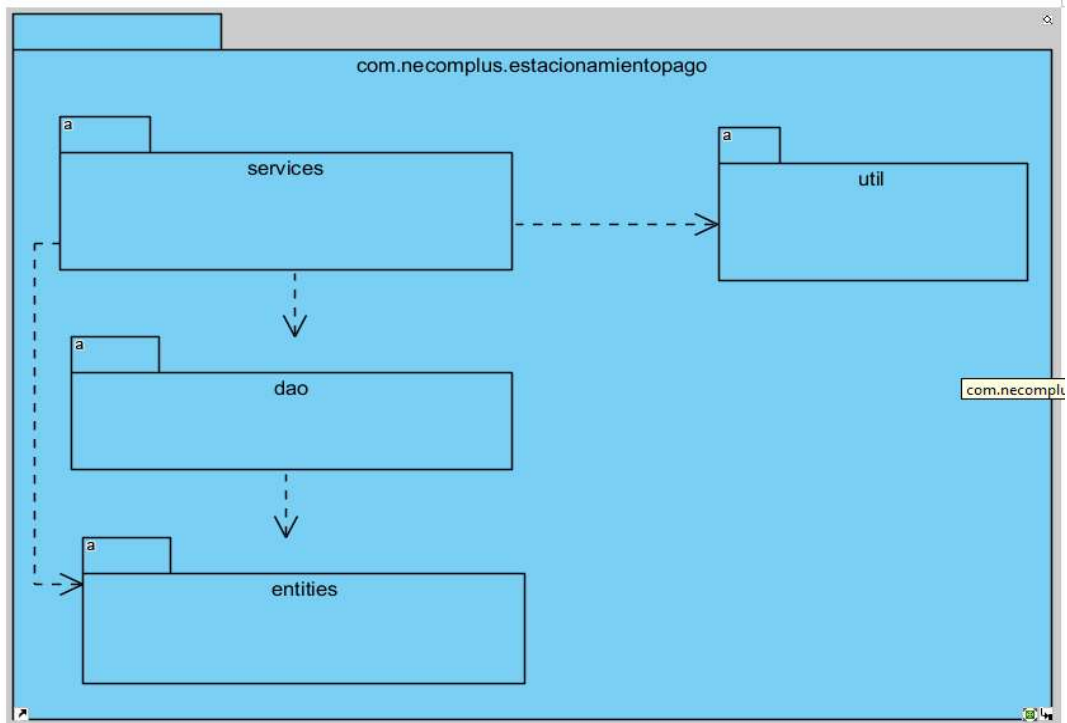
En la tabla *Vehiculo* se guardan los tres tipos de vehiculos (oficiales, residentes y no residentes). En cuanto a los campos podemos decir que: *hora_entrada* contiene la fecha y hora de entrada del vehiculo y se borra cuando se genera la salida. Por otro lado, *minutos_acumulados* aplica para vehiculos residentes.

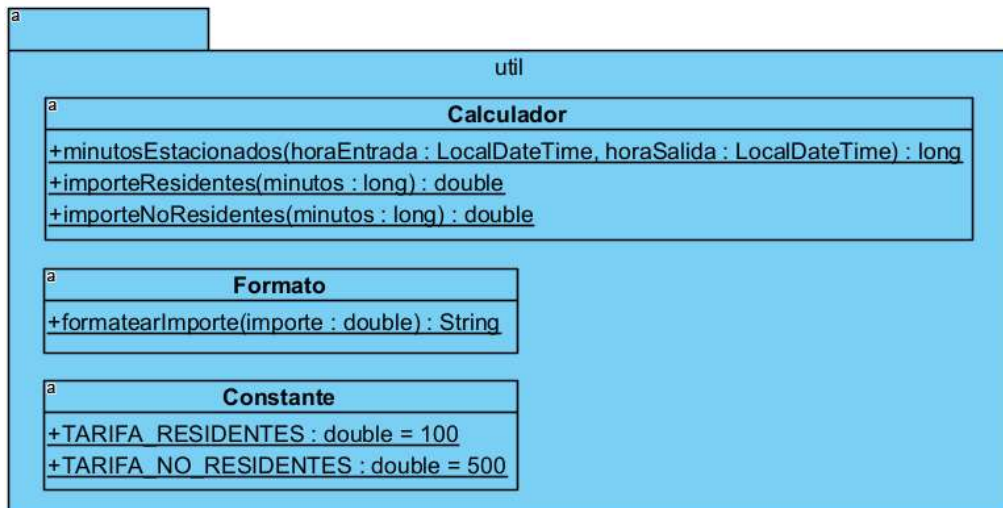
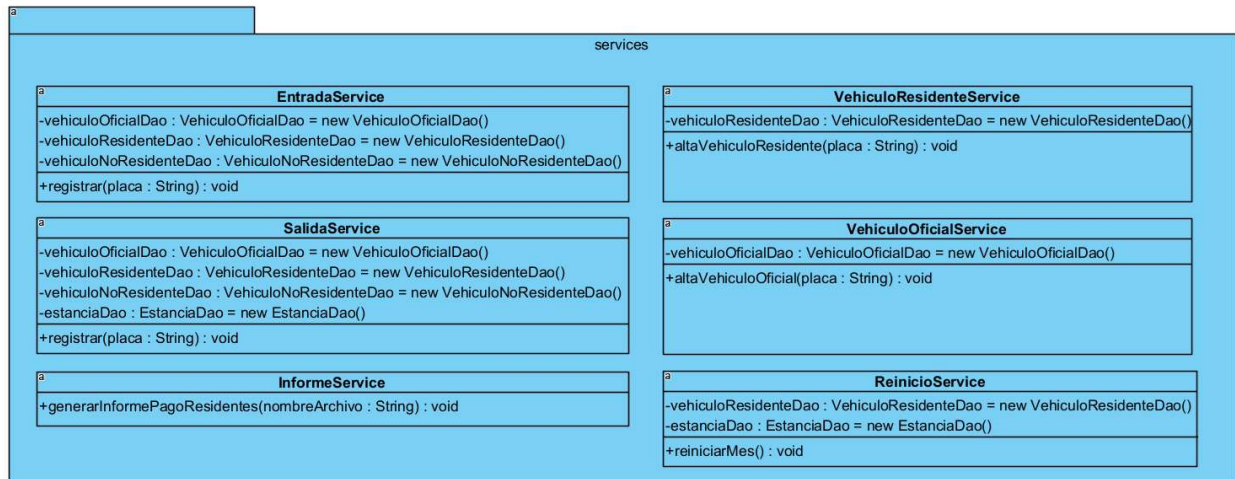
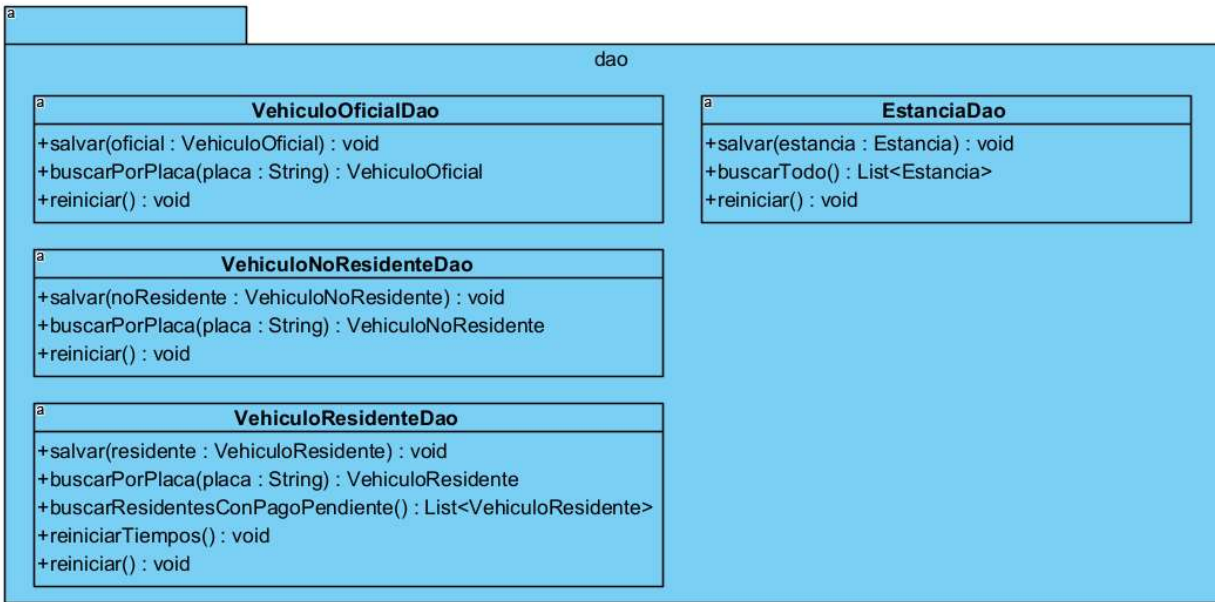
En la tabla *Estancia* se almacenan las estancias (hora de entrada y hora de salida) de los vehículos residentes.

Por último, no se considera importante incorporar disparadores o procedimientos almacenados para mejorar el performance en esta solución, sin embargo, si se crearon índices. Por ejemplo, en la tabla *Vehiculo* se creó uno para garantizar que la *placa* sea única y otro para optimizar las consultas por el campo *tipo_vehiculo*.



Estructura del proyecto





FIN