

# **TP Análisis**

**Integrantes:** Barreto Christian, Waigel Agustina

**Asignatura:** Ingeniería del software I

**Fecha de entrega:** 10/06/2024

2do año de ingeniería en sistemas

## Enunciado:

Como resultado del análisis, se espera que los alumnos produzcan un modelo de análisis que incluya:

- Clases de análisis y sus relaciones
- Paquetes de análisis
- Realizaciones de casos de uso
- Cualquier otro artefacto que consideren relevante para representar la estructura y el comportamiento del sistema.

Una empresa de alquiler de vehículos desea desarrollar un sistema para gestionar su flota de automóviles y el proceso de alquiler a sus clientes. El sistema debe permitir:

1. Registrar y mantener información sobre los vehículos, incluyendo marca, modelo, año, número de placa, kilometraje y estado (disponible, alquilado, en mantenimiento).
2. Registrar y mantener información sobre los clientes, incluyendo nombre, dirección, número de teléfono, correo electrónico y número de licencia de conducir.
3. Realizar reservas de alquiler, especificando el cliente, el vehículo deseado, las fechas de inicio y fin del alquiler, y el lugar de recogida y devolución.
4. Calcular el costo del alquiler basándose en el número de días, el tipo de vehículo y cualquier cargo adicional (por ejemplo, seguro, silla de bebé, etc.).
5. Registrar la devolución de un vehículo alquilado, actualizando el estado del vehículo y generando una factura para el cliente.
6. Generar informes, como la lista de vehículos disponibles, los alquileres actuales, los alquileres por cliente y los ingresos por período de tiempo.

Para la entrega:

- Pueden elegir la herramienta de diagramado que les guste (mermaid, lucidchart, excalidraw, etc).
- La entrega se realiza acá mismo, subiendo un archivo comprimido con los documentos que consideren necesarios para el modelo del análisis.
- Puntos extras si hacen la entrega a través de un repositorio de GitHub (en la entrega ponen el link al repositorio).

## REQUERIMIENTOS OBTENIDOS

### Requerimientos Funcionales

**RF1:** El sistema debe permitir dar de alta nuevos vehículos con sus detalles (marca, modelo, año, número de placa, kilometraje y estado).

**RF2:** El sistema debe permitir modificar la información de los vehículos existentes.

**RF3:** El sistema debe permitir dar de baja vehículos existentes.

**RF4:** El sistema debe permitir consultar los vehículos existentes.

**RF5:** El sistema debe permitir cambiar el estado de un vehículo a disponible, alquilado o en mantenimiento.

**RF6:** El sistema debe permitir dar de alta nuevos clientes, incluyendo nombre, dirección, número de teléfono, correo electrónico y número de licencia de conducir.

**RF7:** El sistema debe permitir modificar la información de los clientes existentes.

**RF8:** El sistema debe permitir dar de baja clientes existentes.

**RF9:** El sistema debe permitir consultar los clientes existentes.

**RF10:** El sistema debe permitir dar de alta reservas de alquiler, especificando el cliente, el vehículo deseado, las fechas de inicio y fin del alquiler, y el lugar de recogida y devolución.

**RF11:** El sistema debe permitir actualizar o cancelar reservas existentes.

**RF12:** El sistema debe calcular automáticamente el costo del alquiler basado en el número de días, el tipo de vehículo y cualquier cargo adicional, como seguro o silla de bebé.

**RF13:** El sistema debe permitir registrar la devolución de un vehículo alquilado.

**RF14:** Al registrar la devolución, el sistema debe actualizar el estado del vehículo.

**RF15:** El sistema debe generar una factura para el cliente al devolver el vehículo.

**RF16:** El sistema debe generar informes que muestren la lista de vehículos disponibles.

**RF17:** El sistema debe generar informes de los alquileres en curso.

**RF18:** El sistema debe generar informes de los alquileres realizados por cada cliente.

**RF19:** El sistema debe generar informes de ingresos por período de tiempo especificado.

**RF20:** El sistema no debe permitir dar de baja un cliente si tiene reservas pendientes, deudas, o facturas de tipo "A" registradas.

**RF21:** El sistema no debe permitir dar de baja un vehículo que se encuentra registrado en una reserva.

## **Requerimientos No Funcionales**

**RNF1:** El sistema debe ser capaz de manejar un creciente número de registros de vehículos, clientes y reservas sin que afecte el rendimiento.

**RNF2:** El sistema debe tener un tiempo de respuesta máximo de 3 segundos para las consultas.

**RNF3:** El sistema debe garantizar la seguridad de la información almacenada.

## **CASOS DE USO**

### **Administrador**

#### **1. Gestionar Vehículo**

- 1.1 Extiende Alta Vehículo
- 1.2 Extiende Modificar Vehículo
- 1.3 Extiende Baja Vehículo
- 1.4 Incluye Consultar Vehículo

#### **2. Gestionar Cliente**

- 2.1 Extiende Alta Cliente
- 2.2 Extiende Baja Cliente
- 2.3 Extiende Modificar Cliente
- 2.4 Incluye Consultar Cliente

#### **3. Gestionar Reserva**

- 3.1 Extiende Alta Reserva
  - Incluye 2.4 Consultar Cliente
  - Incluye 1.4 Consultar Vehículo
- 3.2 Extiende Modificar Reserva
- 3.3 Extiende Baja Reserva
- 3.4 Incluye Consultar Reserva

#### **4. Registrar Alquiler**

- 4.1 Extiende Alta Alquiler
  - Incluye 3.4 Consultar Reserva
  - Incluye 4.1.1 Calcular costo alquiler
  - Incluye 3.4 Consultar reserva

#### **5. Registrar Devolución**

- Incluye 1.4 Consultar Vehículo
- Incluye 4.2 Consultar Alquiler
- 6.1 Extiende Generar Factura
  - Incluye 4.1.1 Calcular costo alquiler

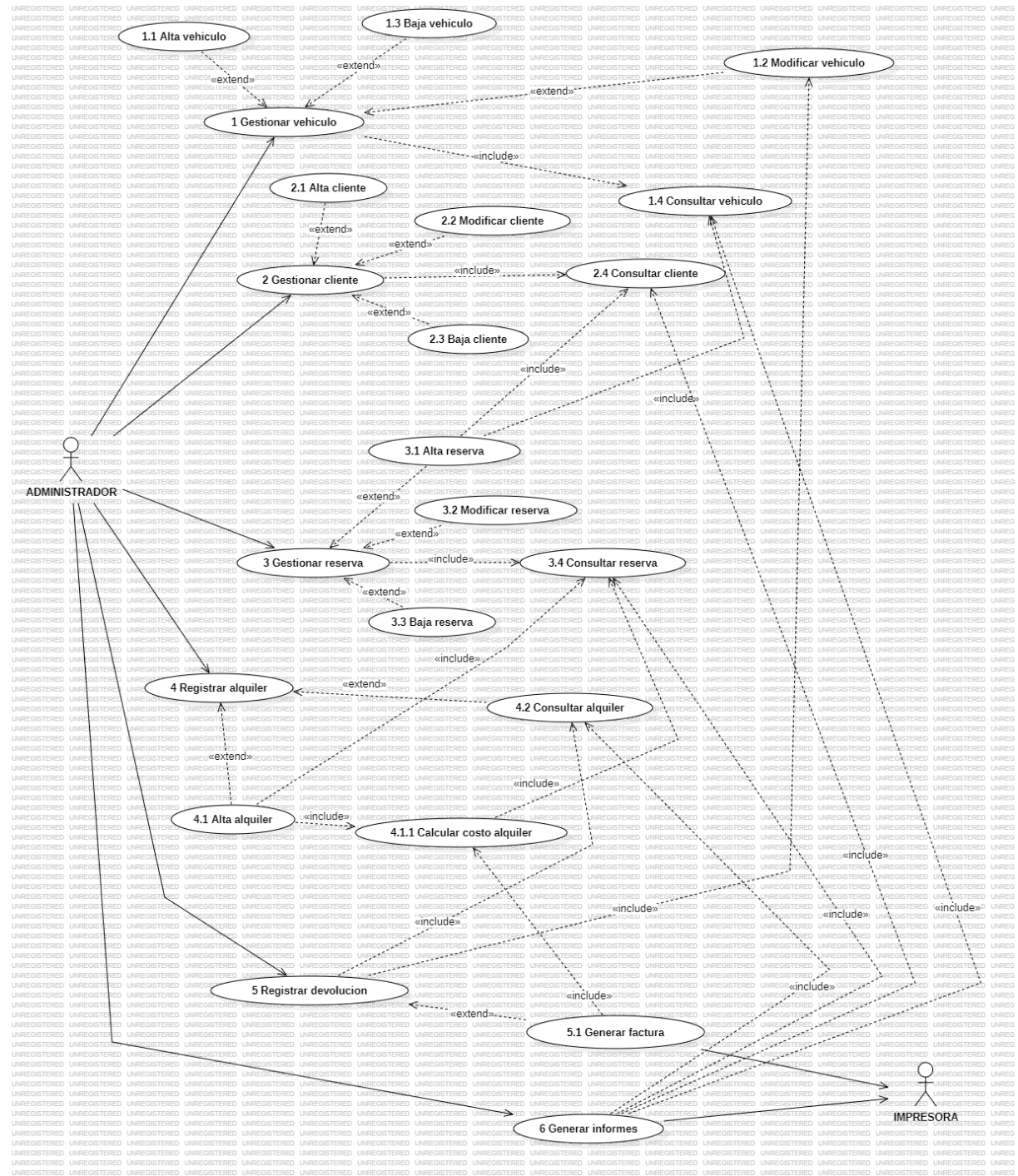
#### **6. Generar Informe**

- Incluye 4.3 Consultar Alquiler
- Incluye 3.4 Consultar Reserva
- Incluye 2.4 Consultar Cliente
- Incluye 1.4 Consultar Vehículo

# Impresora

## 5.1. Generar Factura 6 Generar informe

### DIAGRAMA CASOS DE USO



### REALIZACIÓN CASO DE USO

#### 1. Gestionar Vehículo

El actor **Administrador** utiliza la clase **Vehículo** para realizar las acciones que desee correspondientemente a los métodos de esta clase.

- **1.1 Alta Vehículo** es un método que le permite al administrador dar de alta un vehículo, registrar sus datos e integrarlos en el sistema en la forma de atributos.
- **1.2 Modificar Vehículo** permite al administrador modificar los datos de un vehículo ingresado en caso de que alguno de estos hayan quedado obsoletos por error o modificaciones, tales como el cambio en el estado de un vehículo o un error escribiendo la patente de este.
- **1.3 Baja Vehículo** toma los datos dados de alta previamente en el sistema y luego los elimina de la vista del administrador de tal manera que no se muestren a la hora de realizar una consulta.
- **1.4 Consultar Vehículo** consulta los atributos de un vehículo en cuestión mediante su **ID** y los retorna en un **print()** que muestra al administrador los datos previamente consultados.

## 2. Gestionar Cliente

El **administrador** gestiona los datos de un cliente utilizando los métodos de esta clase.

- **2.1 Alta Cliente** es un método que le permite al administrador dar de alta a un cliente, registrar sus datos e integrarlos en el sistema en la forma de atributos.
- **2.2 Baja Cliente** toma los datos dados de alta previamente en el sistema y luego los elimina de la vista del administrador de tal manera que no se muestren a la hora de realizar una consulta.
- **2.3 Modificar Cliente** permite al administrador modificar los datos de un cliente ingresado en caso de que alguno de estos hayan quedado obsoletos por error o modificaciones, tales como el cambio de nombre o teléfono de este.
- **2.4 Consultar Cliente** consulta los atributos de un cliente en cuestión mediante su **ID** y los retorna en un **print()** que muestra al administrador los datos previamente consultados.

## 3. Gestionar Reserva

Para gestionar las reservas asociadas a un cliente, un **administrador** utiliza varios métodos de esta clase.

- **3.1 Alta Reserva** es un método que le permite al administrador dar de alta una reserva a nombre de un cliente, registrar sus datos e integrarlos en el sistema en la forma de atributos.
- **3.2 Modificar Reserva** permite al administrador modificar los datos de una reserva ingresada en caso de que alguno de estos hayan quedado obsoletos por error o modificaciones, tales como el cambio de la persona a cargo de esta o la fecha de inicio y fin.
- **3.3 Baja Reserva** toma los datos dados de alta previamente en el sistema y luego los elimina de la vista del administrador de tal manera que no se muestren a la hora de realizar una consulta. En este caso, la baja reserva además elimina el hecho de que un cliente pueda retirar los vehículos o generar alquileres por estos, liberando los vehículos de esta reserva.
- **3.4 Consultar Reserva** consulta los atributos de una reserva en cuestión mediante su **ID** y los retorna en un **print()** que muestra al administrador los datos previamente consultados.

#### 4. Registrar Alquiler

Un **administrador** registra alquileres utilizando los métodos propios de esta clase.

- **4.1 Alta Alquiler** es un método que le permite al administrador dar de alta un alquiler asociado a una reserva, registrar sus datos e integrarlos en el sistema en la forma de atributos.
- **4.1.1 Calcular Costo Alquiler** es un método parte de la clase alquiler que se encarga de calcular el costo de este teniendo en cuenta la fecha de inicio y fin del alquiler, condición del vehículo, modelo del vehículo y demás costos.
- **4.2 Consultar Alquiler** mediante este método podemos consultar los atributos de un alquiler mediante su **ID** y retornarlo en la forma de un **print()** que muestre al administrador los datos previamente consultados.

#### 5. Registrar Devolución

En este caso de uso el **administrador** registra una devolución, realizando una consulta del estado actual del vehículo y de la reserva, retornando al cliente una factura con los detalles de la devolución.

#### 6. Generar Informe

Mediante una consulta general de las diferentes clases, un **administrador** genera un informe de los detalles de las reservas.

#### 7. Generar Factura

En el caso de uso **generar factura**, un **administrador** hace uso de los varios métodos de la clase factura con tal de otorgarle al cliente una factura correspondiente a su reserva.

## ETAPA DE ANÁLISIS

#### 1. Diagramas de Clases:

a. **Clases identificadas:** Vehículo, Cliente, Alquiler, Informe, Reserva

b. **Atributos y Métodos de cada clase:**

- **Vehículo**
  - **Atributos:**
    - **Marca**
    - **Patente**
    - **Color**
    - **Modelo**
    - **Estado**
  - **Métodos:**
    - **Vehiculo.AltaVehiculo()**
    - **Vehiculo.BajaVehiculo()**
    - **Vehiculo.ModificarVehiculo()**
    - **Vehiculo.Consultar.Vehiculo()**
- **Cliente**
  - **Atributos:**
    - **Nombre**
    - **Apellido**
    - **DNI**
    - **Dirección**

- Teléfono
  - Email
  - Licencia
- Métodos:
  - Cliente.AltaCliente()
  - Cliente.BajaCliente()
  - Cliente.ModificarCliente()
  - Cliente.ConsultarCliente()
- Reserva
  - Atributos
    - Fecha Inicio
    - Fecha Fin
    - Lugar Recogida
  - Métodos
    - Reserva.AltaReserva()
    - Reserva.BajaReserva()
    - Reserva.ModificarReserva()
    - Reserva.ConsultarReserva()
- Alquiler
  - Atributos
    - Fecha inicio
    - Fecha fin
    - Costo
  - Métodos
    - Alquiler.AltaAlquiler()
    - Alquiler.ConsultarAlquiler()
- Informe
  - Atributos
    -
  - Métodos
    - Informe.RealizarInforme()
- Factura
  - Atributos
    - Número factura
    - Fecha factura
    - Cliente
    - Monto
    - Cargos extras
    - Total
  - Métodos
    - Factura.calcularTotal()

**c. Relaciones entre clases:**

- i. Un cliente puede reservar muchos vehículos, muchos vehículos pueden ser reservados por un cliente.
- ii. Un vehículo pertenece a un alquiler, un alquiler puede tener muchos vehículos.



- iii. Una reserva tiene muchos alquileres, muchos alquileres pertenecen a una reserva.
- iv. Un cliente puede tener muchas reservas, una reserva pertenece a un cliente.

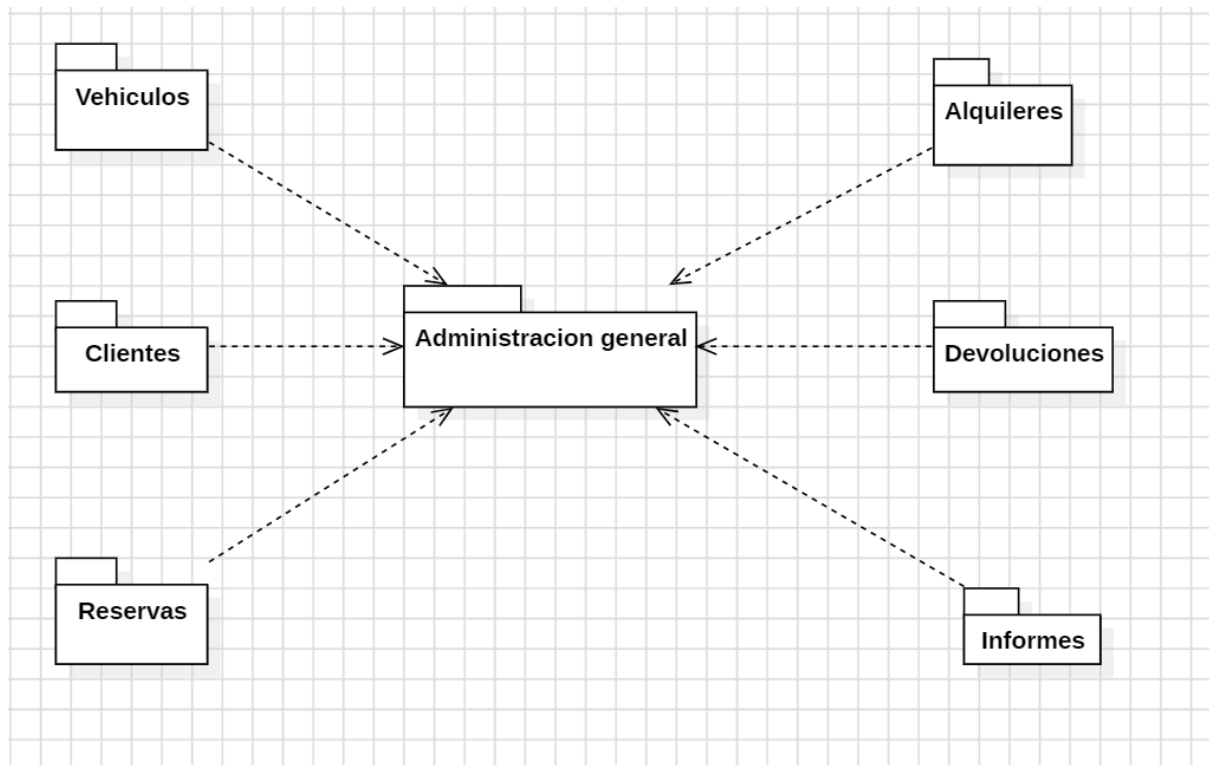
**2. Diagrama de Secuencia:**

- DS1: Registrar una nueva reserva
- DS2: Actualizar información reservas
- DS3: Dar de baja a los vehículos reservados
- DS4: Consultar vehículos disponibles para la reserva
- DS5: Generar orden de alquiler
- DS6: Añadir detalles al informe

**3. Diagrama de Estados:**

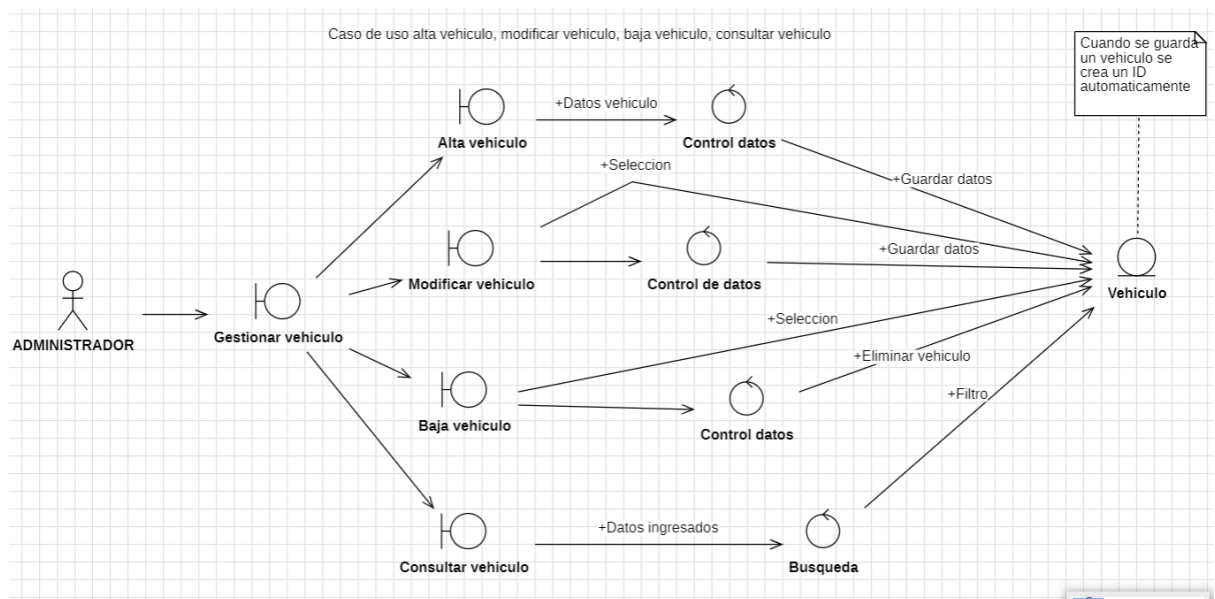
- Estados de un vehículo: Reservado, sin reservas, como nuevo, roto, con detalles, deteriorado
- Estados de un alquiler: Activo, inactivo
- Estados de reserva: activa, inactiva
- Transiciones en los estados en base al estado del vehículo y la duración de los alquileres y reservas (inicio de una reserva, fin de una reserva, estados de los vehículos)

## DIAGRAMA PAQUETES DE ANÁLISIS

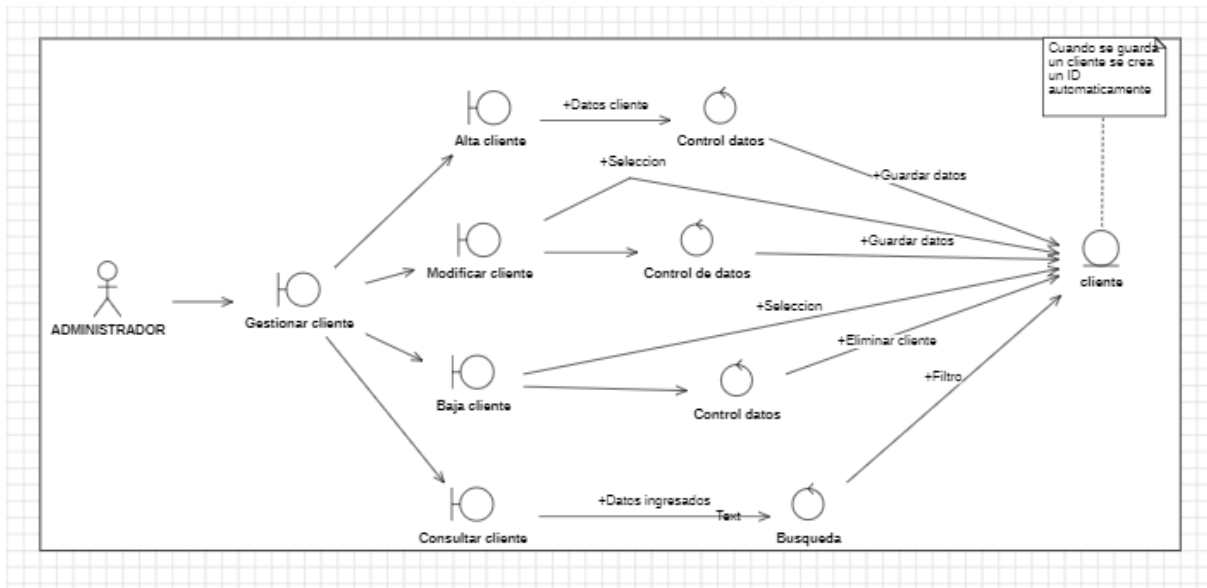


## PAQUETES DE ANÁLISIS

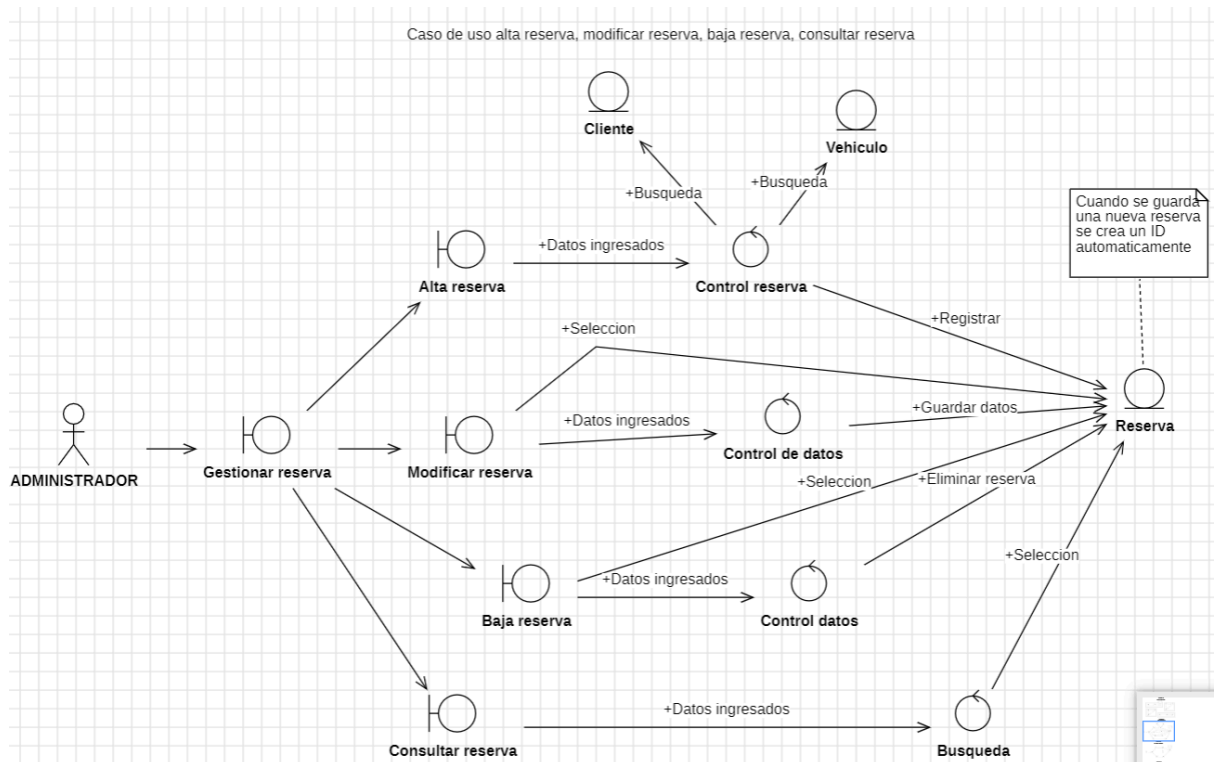
### 1. PAQUETE VEHICULOS



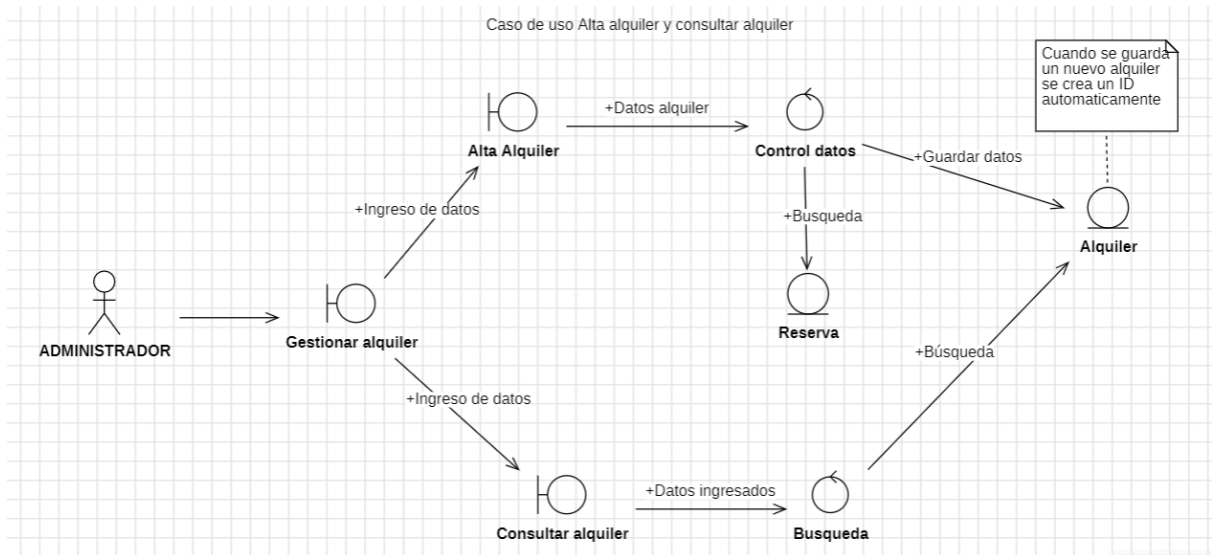
### 2. PAQUETE CLIENTES



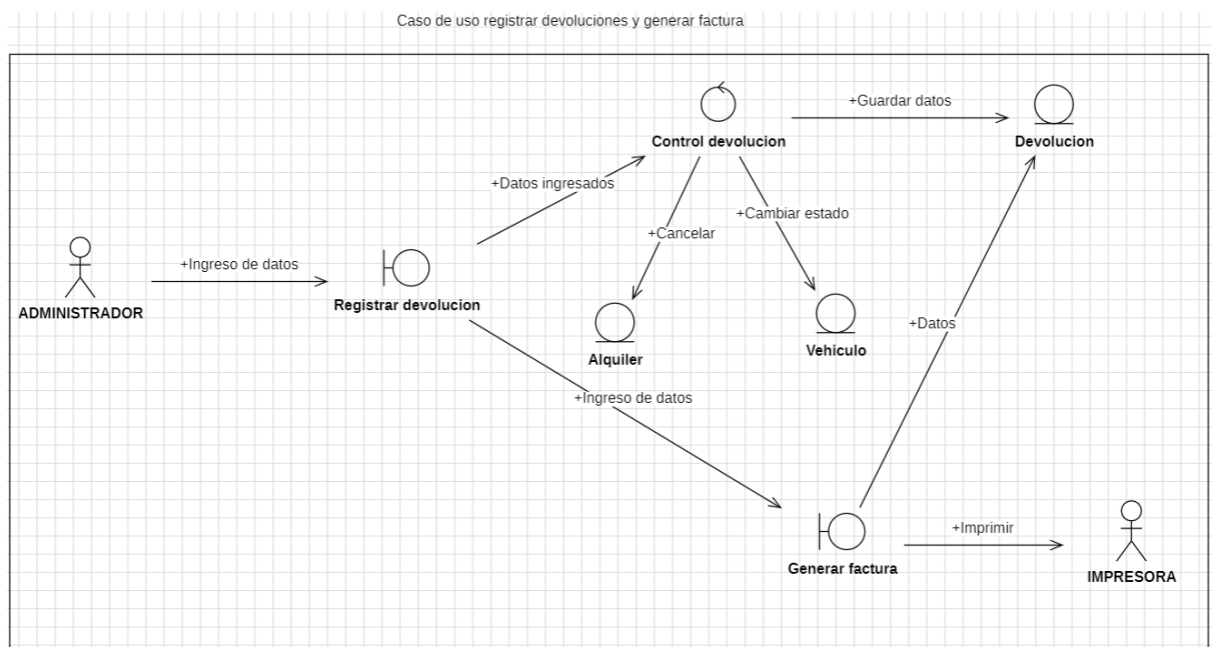
### 3. PAQUETE RESERVAS



### 4. PAQUETE ALQUILERES



## 5. PAQUETE DEVOLUCIONES



## 6. PAQUETE INFORMES

Caso de uso Generar informe

