

## **Proyecto 2 - Entrega 2 : Renta de automóviles**

Grupo 1

Andres Felipe Charry

Ana María Hernández

Julian Mondragón

Diseño y Programación Orientada a Objetos

Ingeniería de Sistemas y Computación

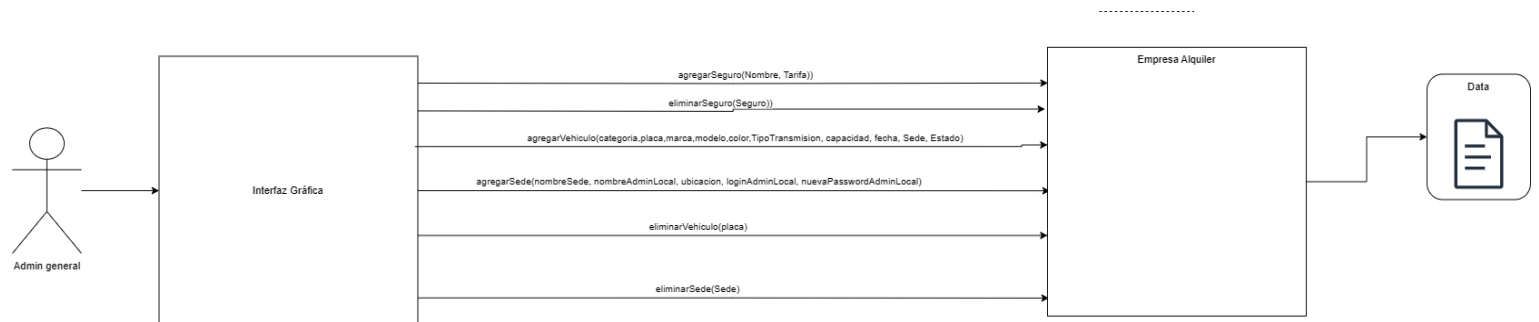
Universidad de los Andes

Bogotá D.C.

Año 2023

## Documento de Diseño.

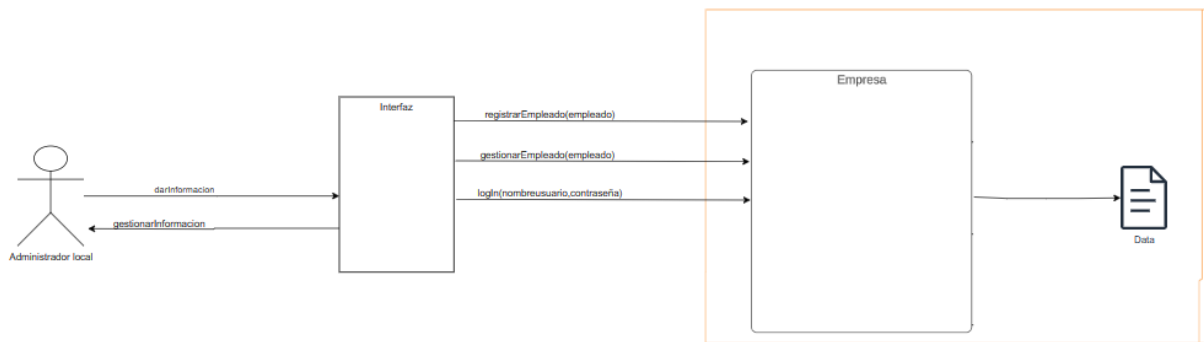
**Contexto del problema:** Antes de dar comienzo al proceso de diseño, se exhibirán las funcionalidades de alto nivel que el sistema debe satisfacer a la hora de interactuar con la interfaz. Esto con el propósito de que independientemente de la implementación, pueda tenerse una representación visual clara del sistema. Con este fin, se mostrarán las interacciones que cada tipo de usuario va a tener con la interfaz del sistema de gestión de vehículos.



**Figura 1: Interacción Administrador global - Interfaz Gráfica**

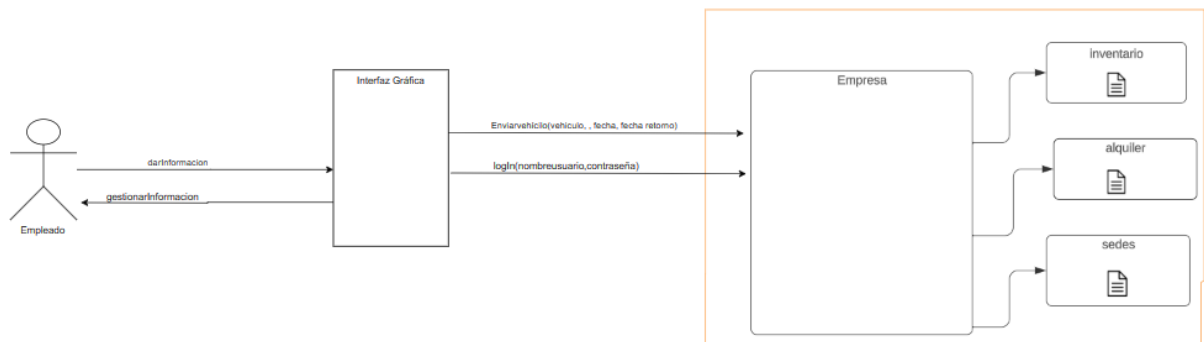
En relación a la interacción del administrador principal con la interfaz, este usuario tiene la responsabilidad de proporcionar la información necesaria para gestionar el inventario de vehículos, ya sea registrando nuevos vehículos o dando de baja aquellos que no puedan seguir siendo alquilados. Además, desempeña un papel vital en la gestión del alquiler de vehículos, ya que tiene la responsabilidad de definir las tarifas por categoría de vehículo, establecer su vigencia y especificar los seguros disponibles para cada automóvil.

Finalmente, se espera que el administrador global gestione la información de las sedes, agregando o eliminando sedes según sea necesario. Ahora bien, respecto a la interfaz, su función principal es autenticar al administrador global, solicitando su nombre de usuario y contraseña, y luego mostrar las opciones adecuadas para cumplir con su rol. También es responsabilidad de la interfaz gestionar toda la información proporcionada por el administrador global y vincularla de manera adecuada con las funcionalidades de inventario, reservas y/o alquiler y gestión de sedes que busca cumplir el sistema.



**Figura 2: Interacción Administrador local - Interfaz gráfica**

Refiriéndose a la interacción del Administrador Local con la interfaz, este usuario tiene la capacidad de gestionar el registro e información de los empleados que forman parte de la sede a la que pertenece. Por su parte, la interfaz debe autenticar al Administrador Local y proporcionarle las opciones necesarias para que pueda cumplir con sus responsabilidades en la sede específica a la que pertenece.



**Figura 2: Interacción empleado - Interfaz gráfica**

En lo que respecta a la interacción del usuario Empleado de Entregas con la interfaz, es fundamental que este usuario tenga la capacidad de cambiar el estado de un vehículo a no disponible. Esto, lo puede hacer mediante el método Enviar Vehículo, el cual le permite enviar el vehículo a mantenimiento o limpieza, según sea el caso.

Por otro lado, la interfaz tiene la responsabilidad de autenticar al Empleado de Entregas de manera precisa y presentar las opciones adecuadas para que pueda cumplir con sus Responsabilidades

## Componentes candidatos y estereotipos

# Roles candidatos: Sistema de reserva de automóviles

### 1. Elementos relacionados con el dominio del problema

- Administrador general
- Administrador local
- Empleado
- Cliente
- Vehículo
- Sede
- Tarifa
- Seguro
- Cliente

### 2. Elementos relacionados con lo que hace el sistema

- Manejador de sesiones
- Reserva

### 3. Cosas que fluyen dentro del sistema

- Inventario

### 4. Agrupadores

- Empresa
- Aplicación

## Refinación de elementos

Al buscar refinar y asignar estereotipos claros en los componentes previamente identificados, llegamos a la siguiente clasificación:

- **Empresa <<Controller>>:**  
Dado que la clase empresa coordina y gestiona información, recibe el estereotipo de controller.
- **Aplicación <<Controller>>:**  
Este componente decide cuál tipo de menú crear ante el usuario según el rol identificado (i.e. empleado, administrador o cliente). Así, controla la vista presentada y los protocolos a continuar.
- **Inventario <<Structurer>>:**  
Este componente, se encarga de cargar y eliminar información para que se pueda acceder a la misma, por lo que recibe el estereotipo de structurer.
- **InterfazAdminGeneral<<Interfacer>>:**  
Dado que este componente se encarga de transformar información y peticiones entre diferentes partes del sistema, para completar las peticiones del Admin General, recibe el estereotipo de interfacer.
- **InterfazAdminLocal<<Interfacer>>:**  
Dado que este componente se encarga de transformar información y peticiones entre diferentes partes del sistema, para completar las peticiones del Admin Local, recibe el estereotipo de interfacer.
- **InterfazCliente<<Interfacer>>:**  
Dado que este componente se encarga de transformar información y peticiones entre diferentes partes del sistema, para completar las peticiones del Cliente, recibe el estereotipo de interfacer.

- **InterfazEmpleado<<Interfacer>>:**

Dado que este componente se encarga de transformar información y peticiones entre diferentes partes del sistema, para completar las peticiones del Empleado, recibe el estereotipo de interfacer.

- **ManejadorSesiones <<Structurer>>:**

Es el encargado de identificar qué tipo de usuario desea usar el programa, obteniendo dicha información y comunicándose con Aplicación.

● **Nota importante:** Damos por entendido que los componentes no mencionados en esta sección, se catalogaron como <<Information holders>>, ya que su rol principal es mantener información para el funcionamiento del sistema en conjunto.

### **Agrupamiento de Elementos**

Una vez que hemos identificado los elementos, podemos distinguir posibles clústers para la implementación:

- **Primer clúster: Aplicación - Manejador Sesiones - InterfazCliente - InterfazEmpleado - InterfazAdminGeneral - InterfazAdminLocal:**

Aplicación es la clase encargada de recibir el usuario y contraseña de la persona que usa el programa, de este modo en el Manejador Sesiones se identifica el rol del usuario y se comunica a Aplicación nuevamente. Allí se decide qué interfaz mostrar por consola (InterfazCliente, InterfazEmpleado, InterfazAdminGeneral o InterfazAdminlocal). En caso de ser la interfaz gráfica se comunica con los menús del paquete Interfaz Gráfica. En conjunto, todas estas clases se encargan de la recolección de datos y del login del usuario.

- **Segundo clúster: Empresa - Inventario**

La clase Empresa instancia los objetos correspondientes de todas las clases (como Sede, Vehículo, entre otras). Para ello colabora con la clase Inventario, la cual carga los datos de los archivos y organiza los objetos de Empresa en estructuras de datos. Todo ello permite un correcto manejo del inventario de la empresa.

### **Asignación de responsabilidades**

<b>N°</b>	<b>Responsabilidad</b>	<b>Componente encargado</b>
<b>1</b>	Gestionar interfaces dependiendo del rol que recibe.	<b>Aplicación</b>
<b>2</b>	Identificar el rol de un login.	<b>Manejador de sesiones</b>

3	Comunicar el rol de un login a Aplicación	
4	Recibir la información necesaria para responder a una petición hecha por la interfaz.	<b>Empresa</b>
5	Modelar la información recibida para crear las clases necesarias para responder la petición hecha por un usuario.	
6	Comunicar al inventario información necesaria para dar persistencia a la información y responder a la petición hecha por el usuario.	
7	Recibir la información que gestionó la empresa.	<b>Inventario</b>
8	Modificar archivos y estructuras de datos correspondientes según la gestión que llevó la empresa previamente.	
9	Ofrecer los métodos disponibles para el usuario Admin General.	<b>InterfazAdminGeneral</b>
10	Ofrecer los métodos disponibles para el usuario Admin Local.	<b>InterfazAdminLocal</b>
11	Ofrecer los métodos disponibles para el usuario Cliente.	<b>InterfazCliente</b>
12	Ofrecer los métodos disponibles para el usuario Admin empleado.	<b>InterfazEmpleado</b>

### Sobre los diagramas

Debido al gran tamaño de los diagramas, estos se encuentran en pdf en la carpeta data del proyecto. Así se permite una mejor visualización de los mismos. Estos son:

- Diagrama UML
- Diagrama de alto nivel de interfaz y de la lógica (por separado, pues así se entienden mejor).

**Aviso importante:** Para ejecutar correctamente la aplicación, es necesario instalar JCalendar, para esto, adjuntamos el siguiente link: <https://toedter.com/jcalendar/> (Esto es importante para poder ingresar las fechas de manera correcta). Esperamos que el uso de esta funcionalidad no afecte nuestra calificación, pues no está restringido en el enunciado del proyecto.