

David Caro - 202222073
Alejandro Abril - 202224328
Gabriel Martinez - 202214559

Proyecto 1 DPOO - Entrega 2

Descomposición del problema.

Sistema completo: Se debe construir un sistema para controlar una parte de las operaciones de una empresa que se encarga de rentar automóviles.

Primera descomposición:

Inventario	Alquiler	Sedes	Funcionalidades
Debe tener un sistema de inventario, de los carros disponibles en cada una de las sedes.	Debe tener un sistema de alquiler que gestiona el proceso desde que se reserva un vehículo hasta que es regresado, incluyendo el cálculo de tarifas, la gestión de clientes, conductores y pagos.	Manejará la información de las sedes incluyendo la información de sus empleados, y el control de los mantenimientos de los vehículos.	La información debe guardarse en archivos, los cuales deben ser persistentes, dentro de una carpeta, la cual solo puede editar la aplicación. La persistencia no necesariamente debe hacerse en un archivo, puede haber más de un archivo. La carpeta donde se guardan los archivos no debe ser la misma que la del código fuente. Todos los usuarios deben tener un login y un password y usaran la misma app dependiendo del tipo de usuario. Debe ser capaz de generar un archivo de log con el historial de un vehículo.

Segunda descomposición:

ALQUILER		
Controla las reservas y alquileres de vehículos, los únicos que pueden reservar o alquilar vehículos, son clientes registrados.		
Cliente: <ul style="list-style-type: none">1. Nombres2. Datos de contacto3. Fecha de Nacimiento4. Imagen doc de identidad	Tarjeta credito: <ul style="list-style-type: none">1. Número2. Fecha Vencimiento3. CCV4. Banco	Alquileres: <ul style="list-style-type: none">1. Categoría2. Tarifa diaria3. Sede entrega4. Sede devolución5. Seguros6. Conductor adicional7. Cobro final
Licencia: <ul style="list-style-type: none">1. Número2. País exp3. Imagen doc4. Fecha Venc	Reserva: <ul style="list-style-type: none">1. Tipo de carro2. Fecha y hora retorno3. Cobro inicial	Reserva Especial: <ul style="list-style-type: none">1. Sede donde se encuentra el carro

SEDES		
Esta parte es la que maneja las sedes y los empleados		
Sedes <ul style="list-style-type: none">1. Nombre2. Ubicación3. Horarios de atención	Administrador general: <ul style="list-style-type: none">1. Administrar sede2. Agregar sede	Empleado: <ul style="list-style-type: none">1. Actualizar estado carro2. Mantenimiento3. Fecha disponibilidad
Administrador local: <ul style="list-style-type: none">1. Sede2. Nuevo empleado3. Nuevo usuario4. Cambiar empleado		

INVENTARIO
Maneja el inventario de los vehículos que son propiedad de la empresa.

Vehículo: <ol style="list-style-type: none"> 1. Placa 2. Marca 3. Modelo 4. Color 5. Tipo de transmisión 	Categorías: <ol style="list-style-type: none"> 1. Pequeños 2. SUV 3. Vans 4. Lujo 5. Eléctricos
--	---

FUNCIONALIDADES		
Archivo sedes: <ol style="list-style-type: none"> 1. Datos sede (Nombre, Ubicación, Horario atención) 2. Administrador general (Nombre) 	Archivo Empleado: <ol style="list-style-type: none"> 1. Datos empleados(Nombre, sede) 2. Datos administración local (Nombre, sede) 	Archivo pagos: <ol style="list-style-type: none"> 1. Duración alquiler. 2. Seguros 3. Sede entrega 4. Sede devolución 5. Conductor adicional 6. Cobro final
Archivo inventario: <ol style="list-style-type: none"> 1. Vehículo 2. Categoría 3. Donde se encuentra 	Archivo log: <ol style="list-style-type: none"> 1. Vehículo 2. Fecha compra vehiculo 3. Alquileres (Fecha Inicio, Fecha fin) 4. Mantenimientos (Fecha inicio, Fecha fin) 	Login/Password: <ol style="list-style-type: none"> 1. Data base de users y password. 2. Registrarse (usuario, empleado, administrador) 3. Ingresar. 4. Direccionar a la interfaz para usuario, empleado o administrar.

Toma de decisiones:

Después de haber hecho la descomposición del problema se explicará la toma de decisiones propuestas en los diagramas.

Se crearon 4 clases principales, se realizó una clase vehículo que contiene los atributos de un vehículo que puede ser alquilado, dicha clase se relaciona con la clase categoría que contiene atributos relacionados con las tarifas y los tipos de carros, de esta manera podemos manejar diferentes tarifas por categoría y recargos por conductores adicionales, tenemos una clase Alquiler que tiene los atributos y métodos necesarios para que un cliente pueda realizar un alquiler, y una clase Reserva que hereda a la clase alquiler, por ende hereda sus atributos y métodos, esto lo hicimos debido a que el cliente siempre tendrá la opción de reservar o alquilar, y esencialmente es el mismo proceso salvo algunas excepciones que se definen con los atributos y métodos propios de la clase reserva, de igual forma hicimos otra clase llamada reserva especial que hereda a la clase reserva, por ende, se heredan los atributos y métodos de reserva sumados a sus atributos propios. Ahora para realizar una reserva o un alquiler se necesitan clientes registrados, por ende, tenemos una clase cliente con todos los atributos que debe tener un cliente, esa clase se

relaciona con la clase reserva para especificar que pueden haber de 0 a n clientes haciendo una reserva, por otro lado, están las clases datosLicencia y datosTarjetaCredito que se relacionan con la clase clientes, puede haber 1 licencia y 1 tarjeta de crédito por cada cliente y sus atributos referencian la respectiva información de las entidades que se relaciona con los clientes y con los conductores adicionales. Las clases reserva y alquiler se relacionan con la clase categoría para así poder tener la disponibilidad de los tipos de carros.

Se tiene la clase sede, decidimos abstraer de la realidad el nombre para poder distinguir las sedes, la ubicación debido a que se necesita saber en qué lugar se encuentran los vehículos, los horarios de atención son necesarios debido para los clientes saber a qué horas pueden recoger o dejar un automóvil en la sede. Cada sede tiene un administrador local que se encarga de administrar la sede, agregar empleados y expulsarlos, estos también tienen la opción de crear un nuevo cliente, Por otro lado tenemos a los empleados, los cuales hacen el manejo de todos los carros, estos pueden sacar los carros a mantenimiento y recibirlos, por lo que es necesario tener métodos que logren esa función, entre esos están sacarDelInventario, actualizarEstado. Cada empleado debe poder ser reconocido por lo que tiene un atributo para esta función, el cual es IdEmpleado, que logra reconocer a los empleados por su número de identificación, el empleado al manipular los vehículos tiene un vínculo directo con la clase vehículo, por último se tiene un administrador general, el cual se encarga de agregar nuevas sedes o gestionárselas dependiendo si se quiere cambiar ciertos aspectos de la sede, dentro de la empresa solo existe un administrador general por lo que las cardinalidades que este tiene con otras clases sólo puede ser 1.

Atributos alquilado/no alquilado de la clase vehículo

```
Alquilado:    { cliente: *información del cliente*  
                lugarRecogida: "x"  
                fechaRecogida: "#:##:##"  
                rangoHorasRecogida: #-#  
            }
```

```
noAlquilado:  { sede: x  
                disponibilidad: "si"/"no"  
                proximaDisponibilidad: null/"#:##:##"  
            }
```

Esta imagen representa el mapa que vamos a implementar para el caso de los vehículos que se encuentran alquilados y no alquilados, el mapa llamado alquilado contiene una llave llamada cliente y el valor es la información del cliente, las respectivas llaves lugarRecogida, fechaRecogida y rangoHorasRecogida junto con sus respectivos valores. Para el no alquilado se tiene la sede y su valor, la disponibilidad que puede ser representado con un booleano, y la próxima disponibilidad con su respectivo valor que puede ser nulo o no.

Diagramas de secuencia:

