**Materia:** Laboratorio de Software

**Profesor:** Daniel Fernandez

**Alumno:** Leonardo Cáceres

TRABAJO PRÁCTICO INTEGRADOR

DDD (Domain Driven Design)

**SISTEMA:**   **“GESTIÓN AUTOS REMISERIA”**

1. **Historia de Auto**
   1. **Alta de Auto:**

Como un auto nuevo en el sistema, debe poder registrar el mismo para acceder a las funcionalidades disponibles.

**Criterios de Aceptación:**

El nuevo auto debe incluir campos para ingresar la patente, cantidad de pasajeros máximos, precio por kilómetro y si está disponible para hacer viajes (true/false).

El sistema valida que la patente debe tener 6 caracteres. La cantidad de pasajeros y precio por kilómetro debe ser mayor a 0 (cero).

Una vez realizada la carga el sistema mostrará el listado de vehículos cargados.

* 1. **Baja de Auto**

El sistema realiza la baja de vehículos, ingresando la patente del vehículo a eliminar. Si la baja es positiva, el sistema informa la misma. Si es negativa, informa que el vehículo no se encuentra.

* 1. **Modificación de Auto**

El sistema realiza la modificación de vehículos, ingresando la patente del vehículo a modificar y los datos del vehículo modificados. Si la modificación es positiva, el sistema informa la misma. Si es negativa, informa que el vehículo no se encuentra.

* 1. **Cambio de estado**

El sistema permite el cambio de estado del vehículo (disponible o no), mediante el servicio de Dominio correspondiente, y cuya ejecución y persistencia es realizada por un servicio de aplicación.

1. **Características generales**

El sistema trabaja, creando, modificando y eliminando en un memoria (patrón Repository) y posteriormente, mediante el patrón unit of work, se persisten en la base de datos. El sistema permite seleccionar en capa presentación, el acceso a una base SQL (SQL Server) o NOSQL (MongoDB)

1. **Servicios de aplicación (CASOS DE USO)**

|  |  |
| --- | --- |
| **Servicio** | **Caso de uso** |
| Crear auto | Permite crear autos en memoria |
| Eliminar auto | Permite eliminar auto en memoria |
| Modificar auto | Permite modificar auto en memoria |
| Persistir cambios disponibilidad | Permite modificar el estado de un auto a “disponible”(true), “no disponible”(false) y guarda el cambio en memoria |
| Unit of Work | Coordina y persiste los datos desde el contexto a la base de datos |