

**MEMORIA Marcos De Desarrollo  
VENTA ONLINE: TIENDA DE  
RECAMBIO PARA TALLERES**

Por: Juan Villaverde  
Daniel Ferreiro

1. Arquitectura global
2. Modelo
  - 2.1. Clases persistentes
  - 2.2. Interfaces de servicios
  - 2.3. Diseño de un DAO
  - 2.4. Diseño de un servicio
  - 2.5. Otros aspectos
3. Interfaz Gráfica
4. Compilación e instalación

## Arquitectura global

El proyecto está conformado por 3 subproyectos siendo estos Modelo Test y Web.

### -Modelo

El modelo contiene a su vez 3 subcarpetas principales, los DAOs, los servicios (que a su vez contiene DTOs, excepciones y los propios servicios) y la carpeta sql que contiene los scripts para generar la base de datos, crear tablas y rellenar la base de datos

El resto de archivos .cs que se encuentran fuera de estas carpetas son las clases autogeneradas por entityFramework mediante el database-first y el archivo de Model1.tt

### -Test

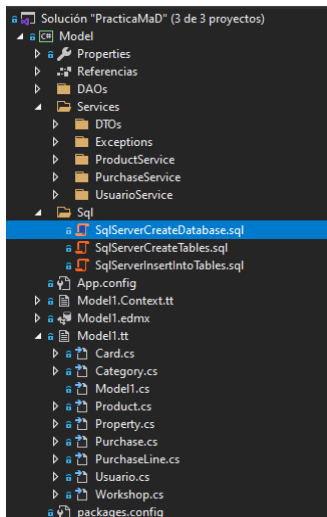
El test contiene las carpetas de:

ServiceTest, la cual contiene los tests de los distintos servicios

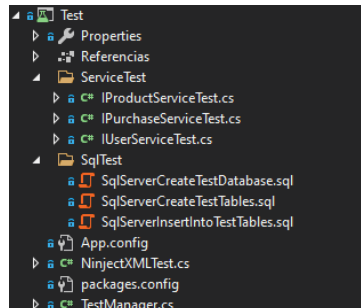
SqlTest, que de igual manera a la carpeta sql del modelo contiene los scripts para inicializar y rellenar las bases de datos donde se llevarán a cabo los tests

### -Web

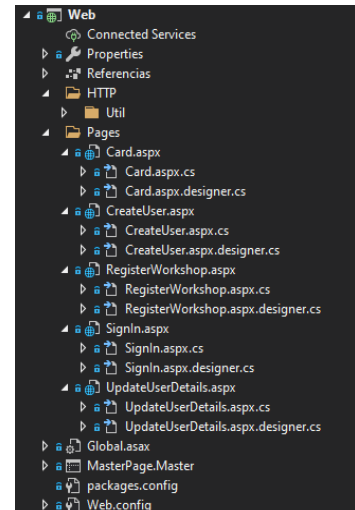
El WEB contiene las páginas necesarias para navegar por la aplicación, además de una carpeta con las páginas de ERROR (404, 500...)



vista solución Modelo



vista solución Test



vista solución Web

## Modelo

### Clases persistentes

Dentro de las clases persistentes tenemos:

Category: categoría del producto i.e neumáticos...

Product: El producto en sí con sus características generales a todos los productos.

Property: Las características más específicas de cada producto

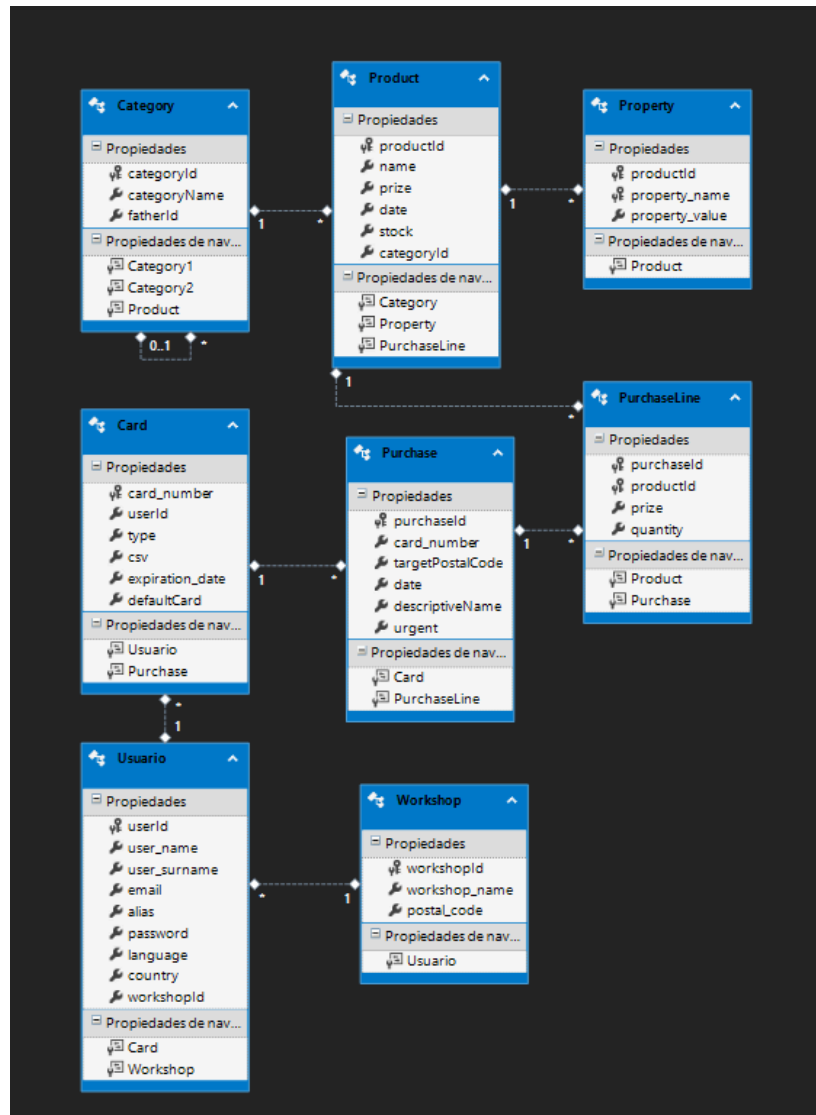
Card: Las tarjetas de crédito pertenecientes a los usuarios

Purchase: Los pedidos

PurchaseLine: El conjunto de productos pertenecientes a un pedido concreto.

Usuario: Un Usuario

Workshop: El taller donde trabajan uno o más usuarios.

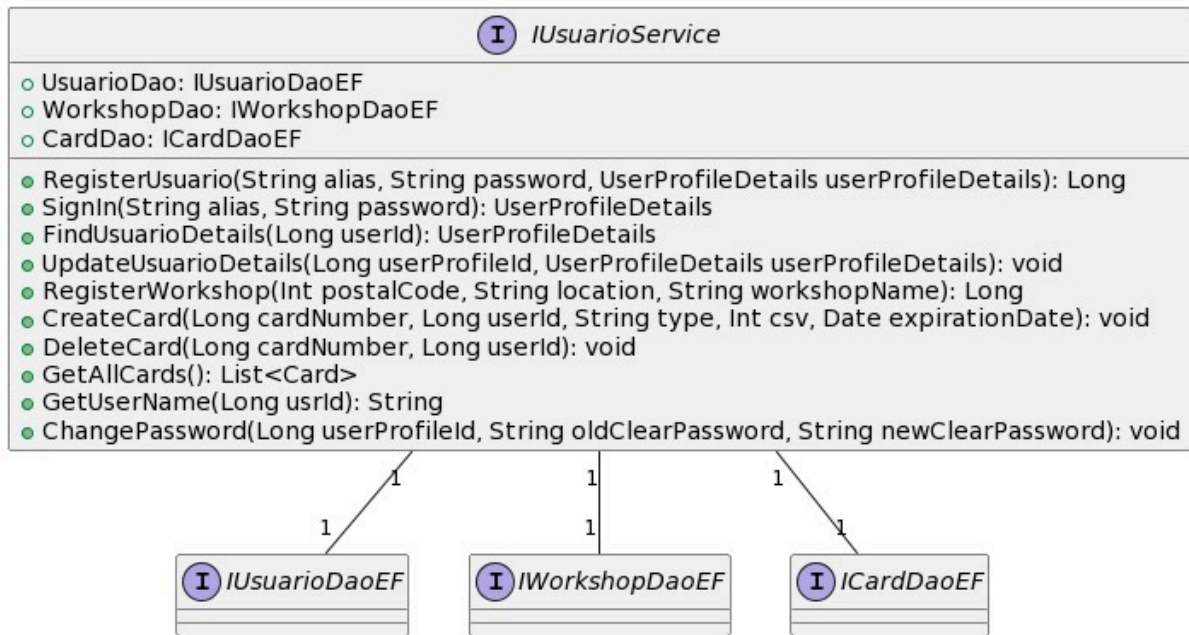


## Interfaces de servicios

Agrupamos los casos de uso en 3 servicios diferenciados:

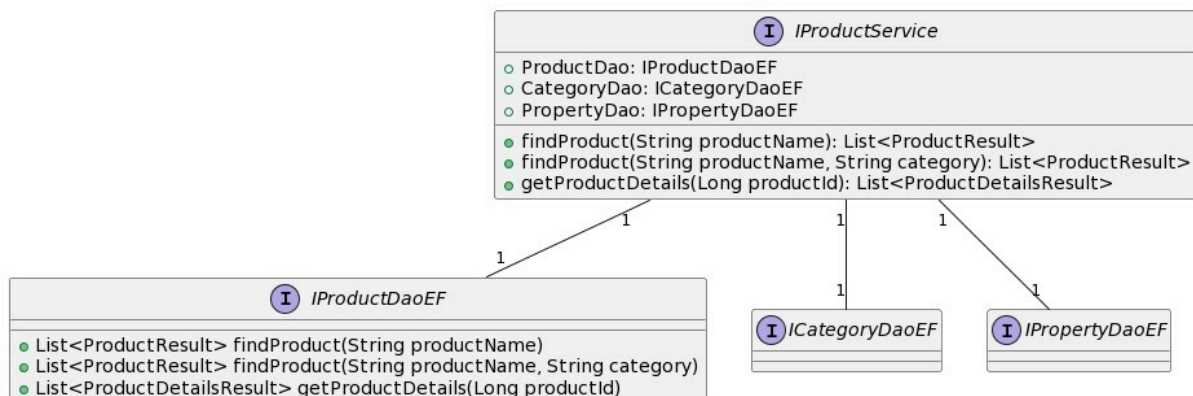
UsuarioService:

Agrupar todas las operaciones relacionadas al usuario o información del usuario, crear y borrar tarjetas, autenticación y registro en la página así como registro de un taller.



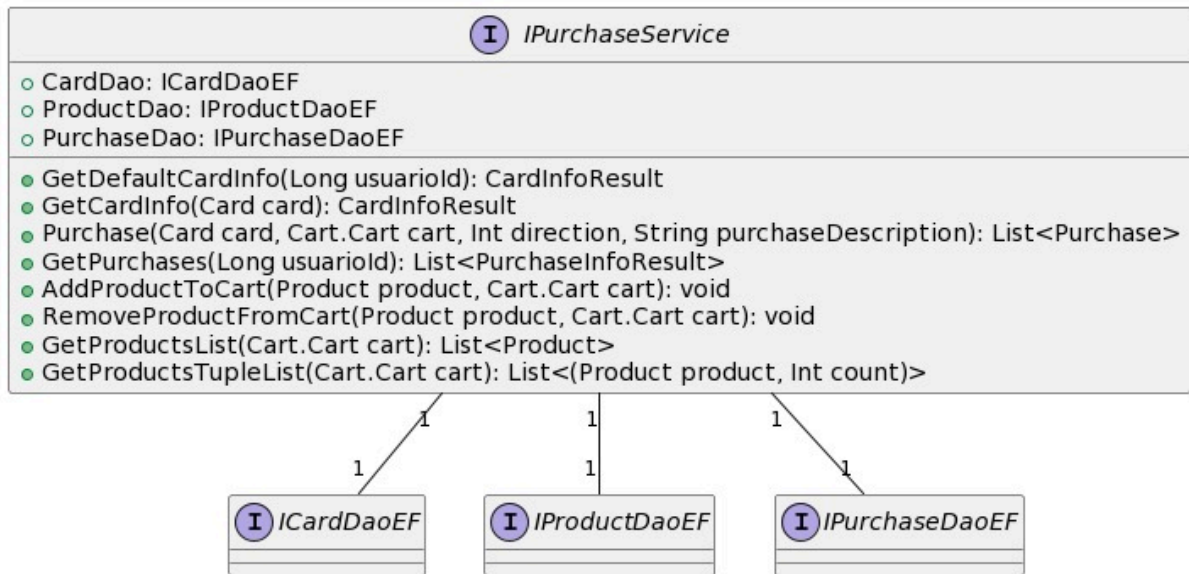
ProductService:

Agrupar las operaciones relacionadas a los productos: encontrar productos, devolver detalles de producto...

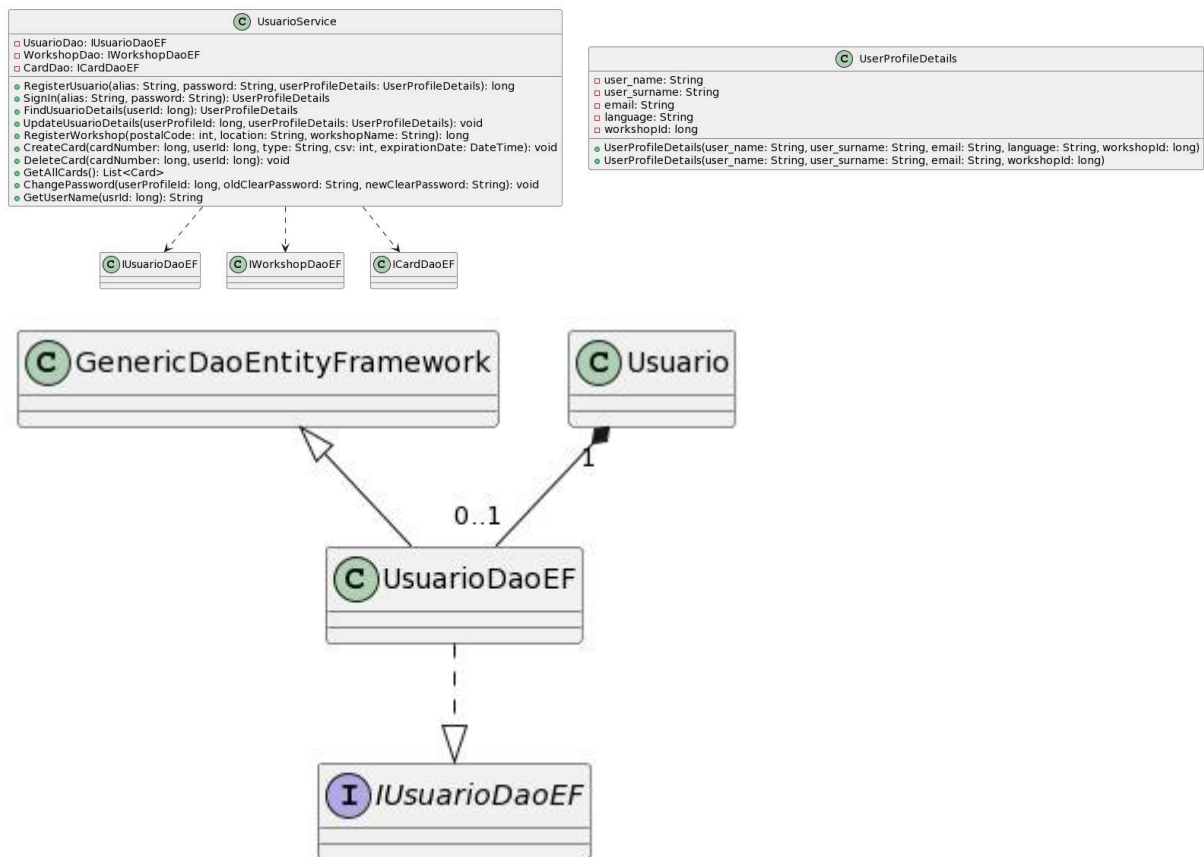


PurchaseService:

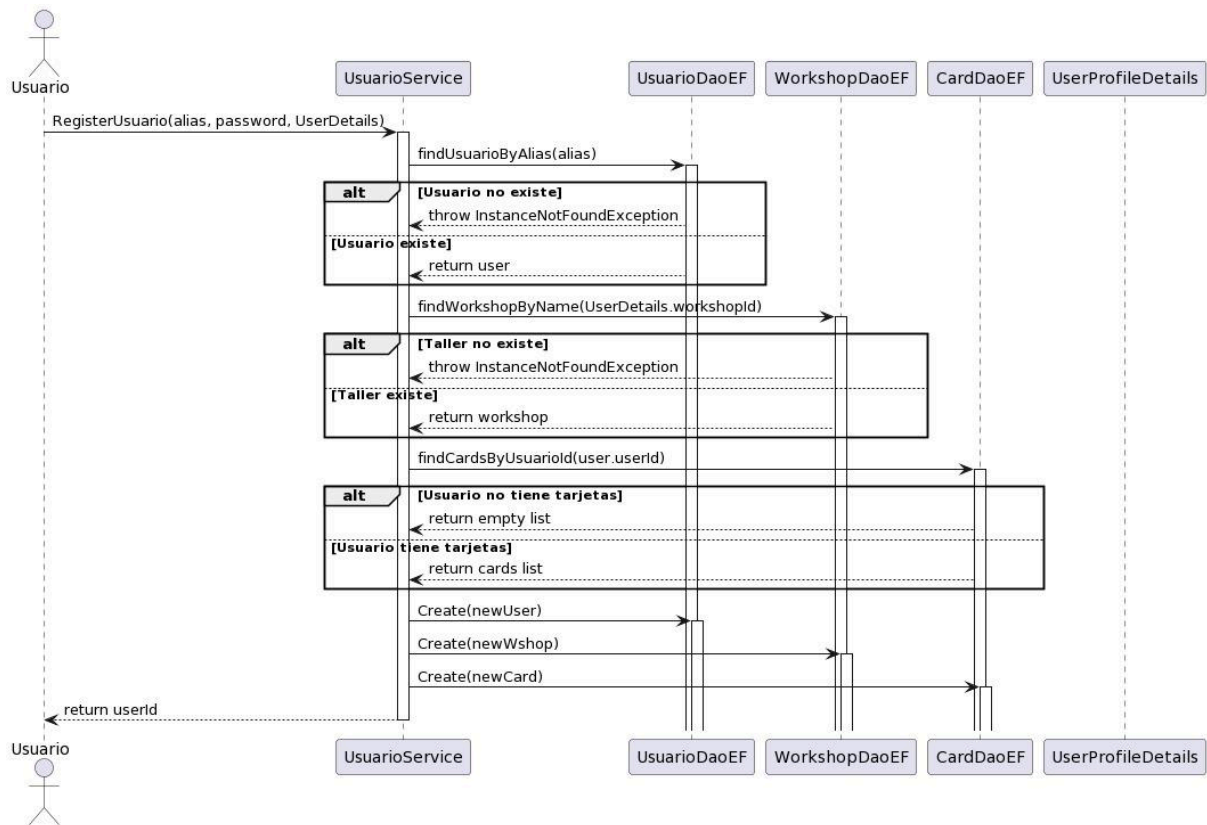
Agrupar todas las operaciones que conlleva la compra de productos, comprar, consultar los pedidos realizados...



## Diseño de un DAO

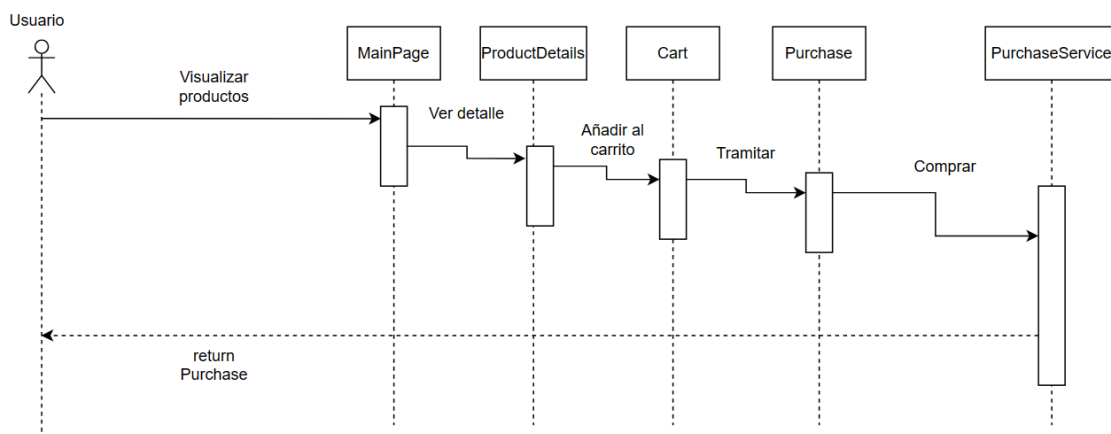


## Diseño de un servicio



## Otros aspectos

## Interfaz Gráfica





## **Compilación e instalación**

Para compilar y ejecutar la práctica se necesita:

- Cambiar data source="" en : Web.Config, Model.App.Config y Test.App.Config