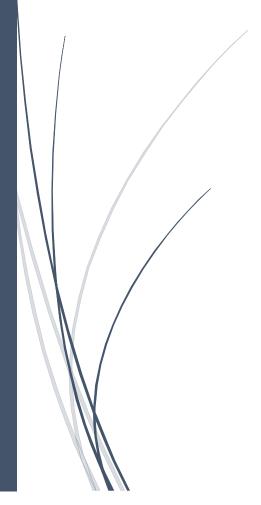
30-5-2022

Manual de Código

Abel Fernández Caballero



Abel Fernández Caballero RETROASSISTANT

Organización de Ficheros

Los ficheros han sido organizados mediante el sistema de módulos y componentes de Angular, en el cual se separan unidades de componentes que realizan acciones independientes entre si y pueden ser reutilizados en el mismo proyecto con el fin de no repetir grandes cantidades de código, como es lógico en Angular el código ha sido desarrollado en TypeScript, HTML 5, CSS 3 y con las directivas que proporciona el mismo angular.

Esta es la estructura de archivos del Font-End en angular (Lo he obtenido mediante un tree de la aplicación para una mayor claridad de la estructura).

Cada modulo que podemos observar se compone de 4 archivos, <Nombre del Modulo>.html <Nombre del Modulo>.ts <Nombre del Modulo>.css <Nombre del Modulo>.html <Nombre del Modulo>.spcec.ts



El back-end del sistema ha sido realizado en php, he estructurado los archivos por la función que desempeñan y en que subsistema se encuadran.

```
C:\USERS\ABEL\DOCUMENTS\PROYECTOFINDECURSO\RETROASSISTANTSERVIDOR
RetroAssistant.sql
Carrito
    annadirProducto.php
    eliminarProducto.php
    eliminarProductosUsuario.php
    obtenerCarrito.php
Compras
    actualizacionDatosProducto.php
    annadirObjetoProducto.php
    annadirProducto.php
    comprarProductos.php
    eliminarObjetoDeProducto.php
    eliminarProducto.php
    obtenerPedidos.php
    obtenerProductos.php
-Rodadas
    obtenerDatosVelocidadConsumo.php
    obtenerMedias.php
    subirRodada.php
-Usuarios
    cambiarContrasenna.php
    cambiarDatosUsuario.php
    crearUsuario.php
    eliminarUsuario.php
    iniciarSesion.php
    obtenerProductosCompradosPorUsuario.php
    obtenerUsuarios.php
```

Se pueden apreciar 3 carpetas separadas por la función que desempeñan sus archivos,

La primera (Carrito) se encarga de gestionar las entradas salidas de los productos del carrito de cada usuario,

La segunda carpeta (Compras) se encarga de gestionar los productos que el usuario ha comprado y de devolver al Front-End los datos que necesita para saber si el usuario tiene permisos para ciertas funciones, también se encarga de modificar los datos de los productos.

La tercera (Rodadas) gestiona la subida de datos del vehículo al servidor al igual que también los devuelve al Front-End con cierto formato para que se pueda interpretar y mostrar las graficas y las medias correspondientes.

La cuarta (Usuarios) se encarga de gestionar todos datos, los inicios de sesión y la creación de nuevos usuarios, al igual que eliminarlos y mandar al Front-End datos para saber que tipo de usuario es el actual

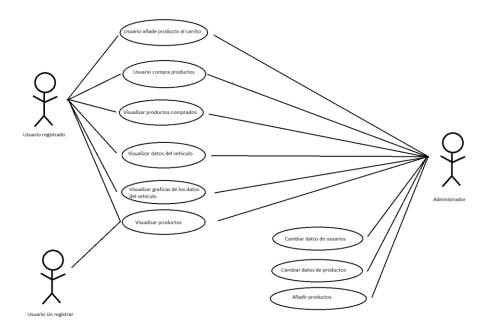
Diccionario de Datos

Al realizar el proyecto con lenguajes no orientados realmente a objetos, aunque si que se puedan utilizar de esta manera, he preferido no utilizar este recurso, pero a pesar de ello si que existen unas pseudo clases que si que se pueden notar en el proyecto, estas coinciden con las tablas de la base de datos pero las podemos desarrollar en este punto.

Usuario	IdUsuario
	Usuario
	Contrasenn
	TipoUsuario
	Email
Carrito	IdUsuario
	Producto
	Cantidad
Pedido	IdPedido
	FechaYHora
	IdUsuario
	ArrayDesglose
Desglose	Producto
	Cantidad
	Precio
Producto	IdProducto
	Nombre
	Descripcion
	Precio
	Imagen
	ArrayDeObjeto
Objeto	Nombre
Rodadas	IdUsuario
	FechaYHora
	VelocidadMedia
	ConsumoMedio
	Duración
	DistanciaRecorrida

Diagramas

Diagrama de casos de uso



Diseño de la base de datos

