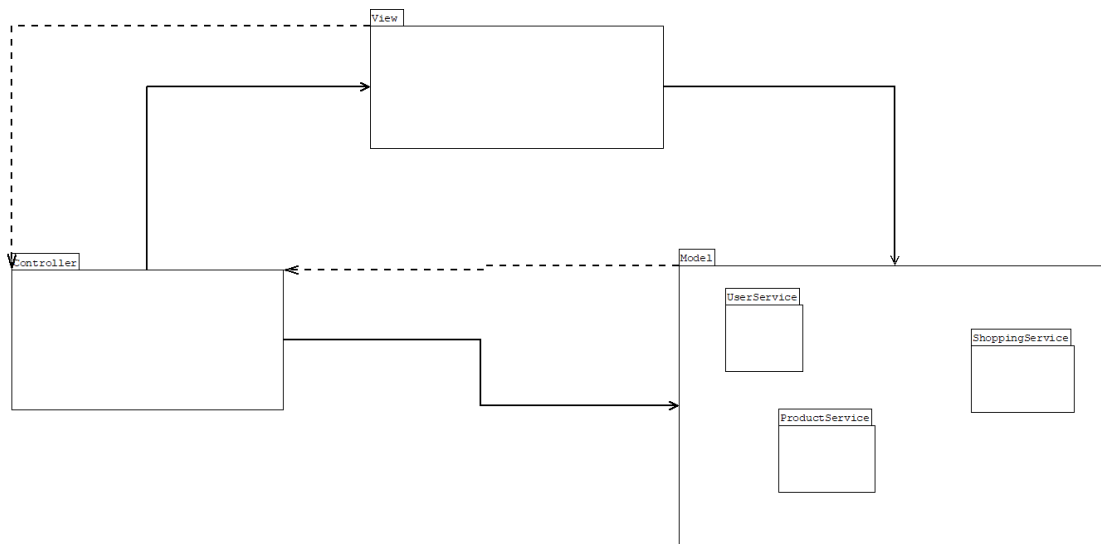


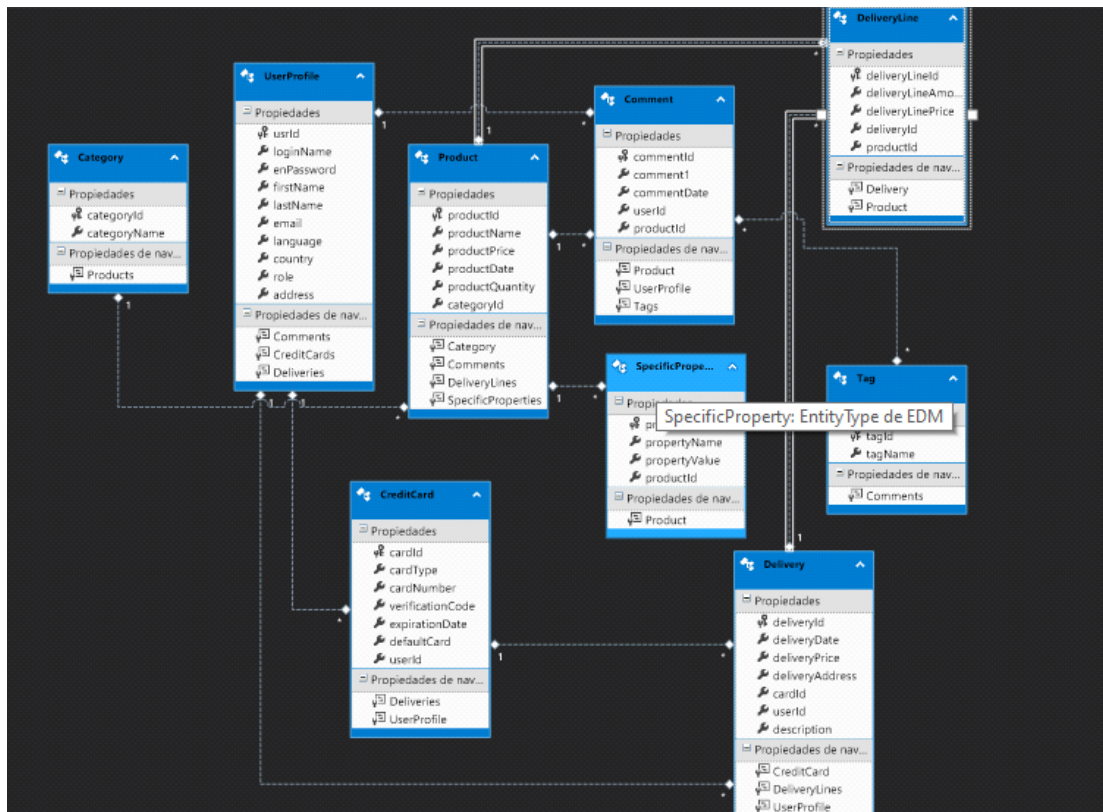
1. Arquitectura global



- View: Paquete encargado de gestionar la interfaz de interacción con el usuario, contiene las ASPX.
- Controller: Paquete encargado de gestionar las iteraciones del view y model, ASPX.cs
- Model: Todo tipo de servicios que implementan los casos de uso e interactúa con la base de datos:
 - UserService: Contiene todos los casos de uso correspondientes al usuario.
 - ShoppingService: Contiene todos los casos de uso relacionados con compras.
 - ProductService: Contiene todos los casos de uso que requieren la manipulación de productos.

2. Modelo

2.1 Clases persistentes



2.2 Interfaces de los servicios ofrecidos por el modelo

IUserService
<pre>+ChangePassword(userProfileId:long,oldClearPassword:string, newClearPassword:string) : void +FindUserProfileDetails(userProfileId:long) : UserProfileDetails +Login(passwordIsEncrypted:Boolean,password:string, loginName:string) : LoginResult +RegisterUser(loginName:string,clearPassword:string, userProfileDetails:UserProfileDetails) : Long +UpdateUserProfileDetails(userProfileId:long, userProfileDetails:UserProfileDetails) : void +UserExists(loginName:string) : bool +UpdateCreditCard(creditCardId:long,creditCardDetails:CreditCardDetails) : void +AddCreditCard(creditCardDetails:CreditCardDetails) : long +AssignDefaultCard(creditCardId:long,userId:long) : void +FindAllCreditCardsDetails(userID:long) : List<CreditCard> +FindCreditCardsDetails(cardID:long) : CreditCard</pre>

IShoppingService

```
+CreateDelivery (deliveryPrice:decimal, cardNumber:long,
                userId:long, description:String,
                shoppingCart:List<ShoppingCart>,
                deliveryAddress:String=null) : Delivery
+GetallDeliveries (userId:long, startIndex:int=0,
                  count:int=20) : DeliveryBlock
+GetDeliveryDetails (deliveryId:long, startIndex:int=0,
                    count:int=20) : DeliveryLineBlock
+UpdateShoppingCartDetails (shoppingCart:List<ShoppingCart>,
                           productId:long,
                           amount:int, gift:bool) : List<ShoppingCart>
+DeleteShoppingCartDetails (shoppingCart:List<ShoppingCart>,
                           productId:long) : List<ShoppingCart>
+ModifyAmountOfItems (shoppingCart:List<ShoppingCart>,
                     productId:long, amount:int) : List<ShoppingCart>
+ModifyGift (shoppingCart:List<ShoppingCart>,
            productId:long, gift:bool) : List<shoppingCart>
```

IProductService

```
+DecreaseProductStock (productId:long, quantity:int) : void
+UpdateProductDetails (productId:long, productName:String,
                      productId:long,
                      productPrice:decimal,
                      productQuantity:int) : void
+FindAllProducts (startIndex:int=0, count:int=20) : ProductBlock
+FindAllProductsByKeyword (keyword:string,
                          categoryId:long=-1,
                          startIndex:int=0,
                          count:int=20) : ProductBlock
+FindProduct (productId:long) : Product
+AddComment (productId:long, userId:long, commentBody:string) : long
+AddComment (productId:long, userId:long, commentBody:string,
            tags:List<long>) : long
+DeleteComment (commentId:long) : void
+UpdateComment (commentId:long, commentBody:String) : void
+UpdateComment (commentId:long, commentBody:string,
               newTags:List<long>) : void
+UpdateComment (commentId:long, commentBody:string,
               newTags:List<long>, removeTags:List<long>) : void
+FindAllProductComments (productId:long, startIndex:int=0,
                        count:int=20) : CommentBlock
+AddTag (tagName:string) : long
+FindAllTags (startIndex:int=0, count:int=20) : TagBlock
+FindAllCategories () : List<Category>
```

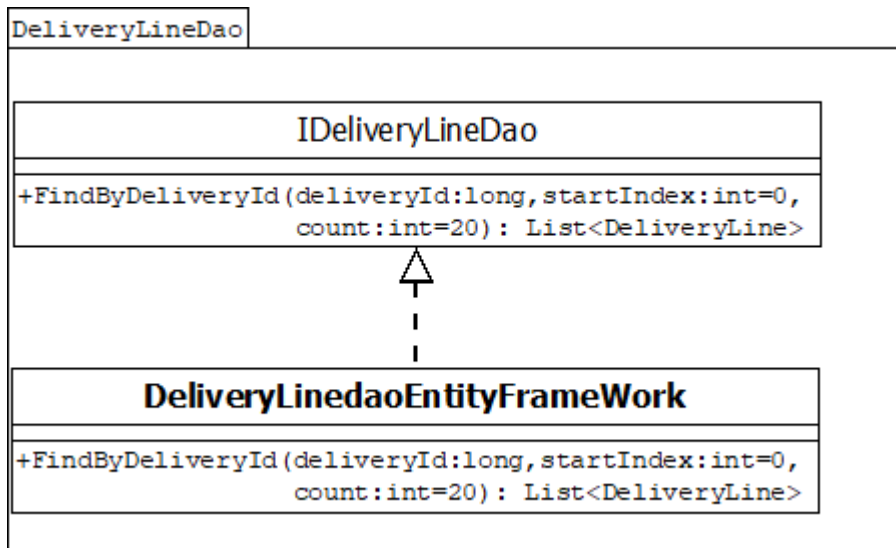
En la distribución de los casos de uso según el servicio, hemos realizado la diferenciación de tres grandes temáticas;

-Usuario, relacionado con todos los casos de uso que puedan afectar a un usuario ya sea por la modificación de sus datos personales, verificaciones sobre ellos, necesidad de obtener sus datos....

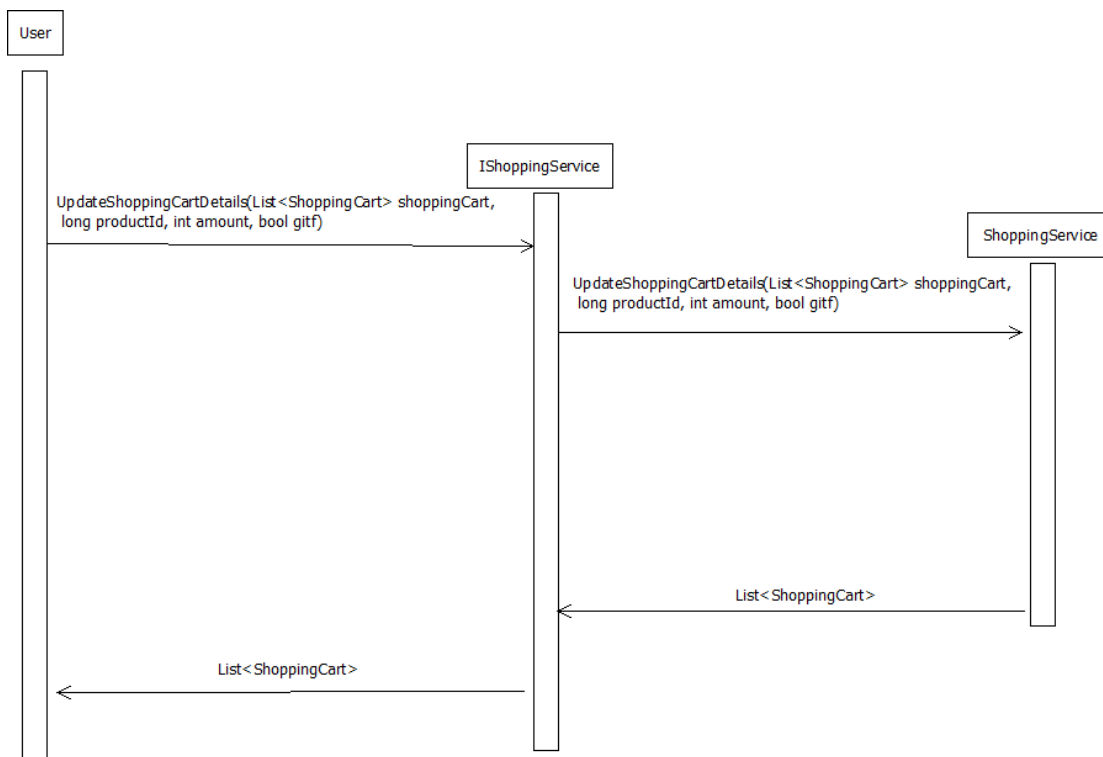
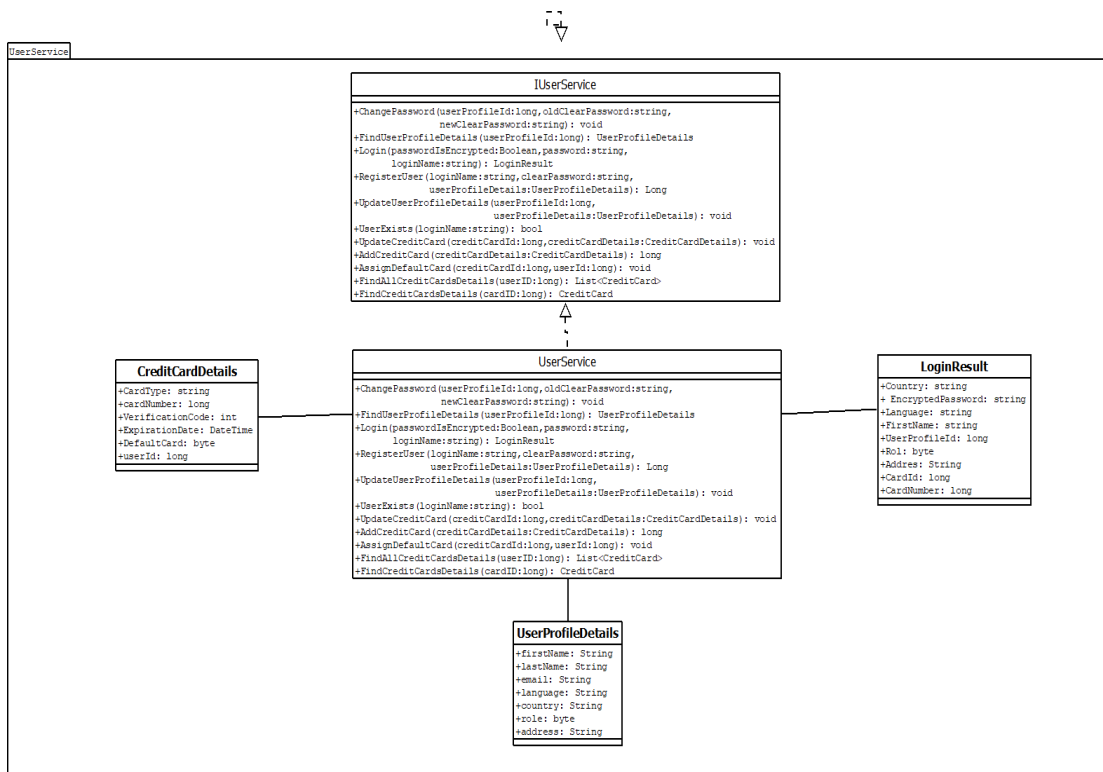
-Producto, relacionado con toda la interacción genérica con los productos, ya sea obtener información de ellos, hacer operaciones sobre ellos o complementarlos con mas información.

-Shopping, una vez establecidos los productos y los usuarios, toda relación de adquisición de productos y todo tipo de operación que intervenga en la realización de una compra por parte del usuario estará regulada por este servicio.

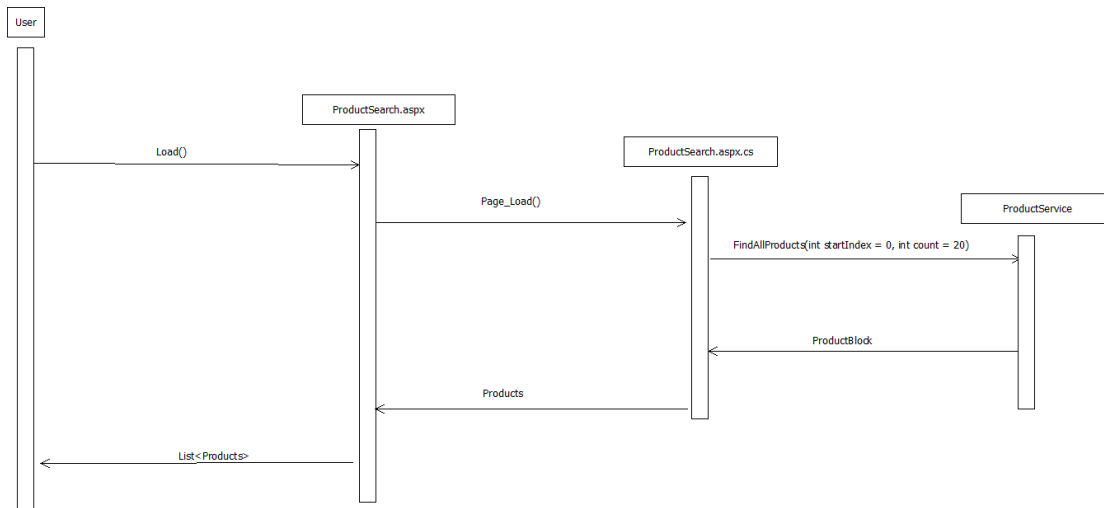
2.3 Diseño de un DAO



2.4 Diseño de un servicio del modelo



3. Interfaz Gráfica



4. Un apartado por cada parte opcional

Se explicará claramente cómo se ha abordado el diseño de esta parte opcional y

lo que se considere necesario explicar para la ejecución de la misma.

--Etiquetado de Comentarios:

En la realización de este apartado opcional hemos añadido a la base de datos las etiquetas, los tags, los relacionamos many to many con los comentarios. Después incluimos funciones en los daos para gestionar las etiquetas, y modificamos las funciones del dao de comentarios para contemplar la posibilidad de añadir o borrar sus etiquetas asociadas.

En la capa de servicios hicimos uso de todas esas funcionalidades para ofrecer a la capa web el soporte para llevar a cabo los casos de uso y finalmente en la vista Web seguimos las directrices de diseño del enunciado para implementar totalmente cada caso de uso.

Ejecución: Para probarlo podríamos añadir un comentario desde la vista de añadir comentario incluyendo en el campo de introducción de etiquetas las etiquetas que se desean relacionar con el comentario separadas por espacios. Otra forma sería modificar un comentario ya creado y añadiendo o eliminando etiquetas del campo de etiquetas.

--Comentario de productos:

En la realización de este apartado opcional hemos añadido a la base de datos una entidad comment, implementamos el dao de comment con funciones particulares, concretamos los casos de uso en la capa de servicios y terminamos de cumplir los requisitos del enunciado, como puede ser añadir eliminar y modificar comentarios en la capa Web.

Ejecución: Podría comprobarse su correcto funcionamiento entrando en los detalles de un producto y haciendo click en el enlace a los comentarios, esto mostraría una lista con todos los comentarios relacionados con el producto en el orden temporal del más reciente al más antiguo. Otra prueba de esta funcionalidad sería añadir un producto y un comentario también desde la vista de detalles del producto.

5. Compilación e instalación de la aplicación

-Click derecho en Solución > Propiedades > Dependencias del proyecto. En el desplegable elegimos Web y nos aseguramos de que model este marcado, le damos a aceptar a la ventana, vamos al explorador de soluciones le damos click derecho a web y vamos a publicar Web.

Saldra una ventana en la que tendremos que darle a todas las opciones por defecto a publicar, hará el proceso de publicar e iremos a la carpeta donde indique la terminal, dentro de la carpeta tendrá todo lo que tenemos que meter en un servidor IIS para poder instalarlo.

6. Problemas conocidos

Lista los errores que se conoce que tiene el código