



(Creative Common)

Esta obra está bajo una licencia Reconocimiento-No comercial-Sin obras derivadas 2.5 España de Creative Commons. Puede copiarlo, distribuirlo y transmitirlo públicamente siempre que cite al autor y la obra, no se haga un uso comercial y no se hagan copias derivadas. La licencia completa se puede consultar en <http://creativecommons.org/licenses/by-nc-nd/2.5/es/deed.es>.



Proyecto MOFM: Made of Felt Market



I. T. Informàtica de Gestió
TFC J2EE. Proyecto Mofm memoria.
Pedro Antonio Leal Martínez
Consultor: Oscar Escudero Sanchez

Índice de contenido

1. INTRODUCCIÓN.....	6
1.1 Justificación del proyecto.....	7
1.2. Objetivos del trabajo.....	7
1.2.1. Objetivos técnicos.....	7
1.2.2. Objetivos funcionales.....	8
1.3. Enfoque metodológico.....	11
1.4. Planificación del proyecto.....	13
1.4.1. Etapas.....	13
1.4.2. Diagrama de Gantt.....	13
1.5. Producto obtenido.....	15
2. ESPECIFICACIÓN Y REQUERIMIENTOS.....	15
2.1. Información inicial.....	15
2.2 Glosario.....	15
2.3 Modelo del dominio.....	17
2.4 Diagrama de casos de uso.....	18
2.5 Documentación textual casos de uso.....	19
2.5.1. Conexión con credenciales (Login).....	19
2.5.2. Registro de cliente.....	19
2.5.3. Ver productos.....	20
2.5.4. Detalle productos.....	20
2.5.5. Agregar producto a cesta.....	20
2.5.6. Ver cesta.....	21
2.5.7. Quitar producto de cesta.....	21
2.5.8. Finalizar compra.....	22
2.5.9. Desconectar (Logout).....	22
2.5.10. Listar usuarios.....	22
2.5.11. Agregar usuario.....	23
2.5.12. Editar usuario.....	23
2.5.13. Bloquear usuario (Eliminar usuario).....	23
2.5.14. Agregar familia.....	24
2.5.15. Ver familias.....	24
2.5.16. Agregar productos a familia.....	24
2.5.17. Editar familia.....	25
2.5.18. Borrar familia (bloquear familia).....	25
2.5.19. Agregar clasificación.....	26
2.5.20. Editar clasificación.....	26
2.5.21. Listar clasificaciones.....	26
2.5.22. Borrar clasificación (bloquear clasificación).....	27
2.6 Requisitos de la Interfaz de usuario.....	27
3. ANÁLISIS.....	28
3.1. Identificación de clases y atributos.....	28
3.1.1. Usuario.....	28
3.1.2. Cliente.....	29
3.1.3. Administrador.....	29
3.1.4. Gestor.....	29
3.1.5. Familia.....	30
3.1.6. Producto.....	30
3.1.7. Clasificación.....	31
3.1.8. Compra.....	31
3.1.9. Relaciones de herencia.....	32

3.2. Diagramas de secuencia.....	33
3.2.1. Login de usuario.....	33
3.2.2. Logout de usuario.....	33
3.2.2. Listar/EDITAR/BORRAR usuarios.....	34
3.2.4. Nuevo usuario.....	35
3.2.5. Ver/FILTRAR productos Y ver detalle.....	36
3.2.6. Comprar producto/Ver cesta/Eliminar producto de cesta/Finalizar compra.....	37
3.2.7. Listar/Editar/Borrar Familia.....	38
3.2.8. Agregar familia.....	39
3.2.9. Listar/editar/borrar clasificación.....	40
3.2.10. Agregar clasificación.....	41
3.2.11. Listar productos (Gestor).....	42
3.3. Modelo de datos. Diagrama entidad-relación.....	43
4. Diseño Técnico.....	43
4.1. Uso de Patrones.....	43
4.1.1. Patrón MVC Modelo Vista Controlador.....	43
4.1.2. Patrón Singleton.....	44
4.1.3. Patrón Persistent Customer SesSion.....	45
4.1.4. Patrón i18n.....	45
4.2. Diagrama de paquetes.....	45
4.3. Diagrama de CLASES GESTORAS (CONTROLADORAS).....	47
4.4. JERARQUÍA DE VENTANAS.....	48
4.5. Diseño interfaz.....	48
5. Implementación.....	56
5.1. Herramientas de desarrollo.....	56
5.1.1. NetBeans.....	56
5.2. MANUAL DE INSTALACIÓN.....	57
6. Conclusiones.....	59
7. Bibliografía.....	60

1. INTRODUCCIÓN

El proyecto pretende desarrollar un entorno web para la comercialización en Internet de diferentes productos elaborados a mano con fieltro. Dicho proyecto se denominará “MOFM: Made of Felt Market”.

El proyecto “MOFM” cubrirá las diferentes necesidades para la comercialización de estos productos a través de varias áreas: Gestión de los productos, gestión de usuarios y venta de productos.

De esta manera se dividirá el proyecto en tres grandes subsistemas: Subsistema gestor de productos, subsistema gestor de usuarios y subsistema de comercialización de productos.

En cuanto al área de gestión de productos o subsistema de comercialización de productos, y a través del entorno web, cualquier cliente potencial podrá ver todos los productos a la venta disponibles en la tienda. Además por cada producto podrá visualizar todos los datos de interés que pueda necesitar para su adquisición: descripción del mismo, precio, disponibilidad, ... Además de una fotografía.

Un cliente podrá registrarse en el sistema proporcionando los datos solicitados por el sistema y obtener de esta manera unas credenciales. Una vez registrado el usuario podrá acceder a través de estas credenciales y acceder de esta manera a nuevas funcionalidades dentro de la aplicación.

Una vez que el cliente esté correctamente registrado en el sistema además podrá agregar productos a su cesta, consultar los productos agregados hasta el momento y eliminar cualquiera de ellos.

Finalmente el cliente podrá finalizar su compra formalizando así la compra de los productos agregados a su cesta hasta ese momento.

Dentro del subsistema gestor de productos, un usuario con el rol de “gestor” podrá iniciar sesión para gestionar los productos disponibles en la tienda. Podrá añadir o eliminar cualquier producto. Clasificar el producto y editar cualquier información al respecto del mismo.

Una vez modificados los productos o cualquier información de ellos, el subsistema de comercialización de productos mostrará los nuevos productos o la nueva información disponibles para éstos.

Dentro del área de gestión de usuarios o subsistema gestor de usuarios, un usuario con el rol de “administrador” podrá dar de alta, modificar o eliminar usuarios del sistema.

Un usuario con el rol “administrador” o gestor sólo podrá ser dado de alta a través del subsistema gestor de usuarios y por consiguiente por un usuario con el rol de “administrador”.

Cualquiera de los subsistemas que forman el proyecto proporcionarán funcionalidades suficientes para mostrar toda la información utilizada por ellos mismos. Por ejemplo, dentro del subsistema gestor de usuarios se podrá visualizar todos los usuarios del sistema. Así mismo el subsistema gestor de productos deberá permitir la visualización de todos los productos y sus características.

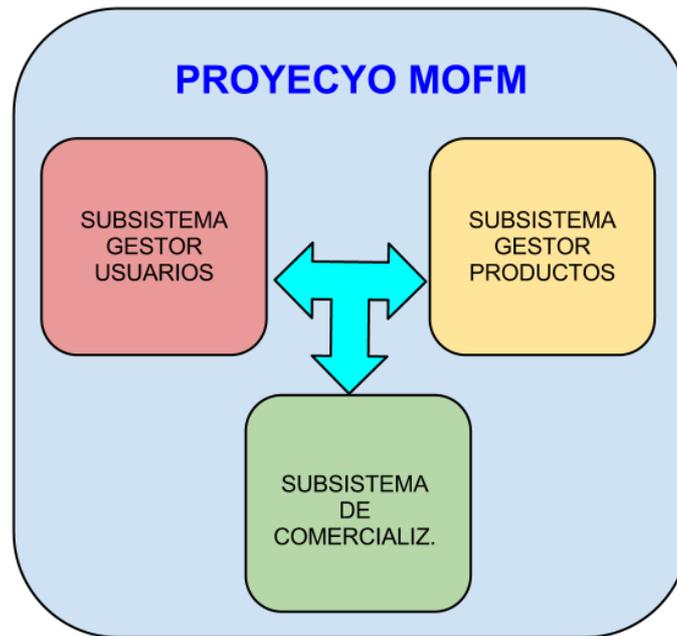


IMAGEN 1.

1.1 JUSTIFICACIÓN DEL PROYECTO

Se ha elegido un proyecto de este tipo por el auge que las aplicaciones web están teniendo en la sociedad del momento.

Desarrollar una aplicación web basada en el modelo vista controlador (MVC) permite al alumno profundizar e investigar sobre las nuevas tecnologías que permiten este tipo de desarrollo y de esta manera alcanzar un mayor conocimiento de las herramientas que existen actualmente en el mercado para desarrollar un producto de este tipo.

De esta manera, la profundización y el conocimiento adquiridos, prepara al alumno para poder afrontar el desarrollo de este tipo de proyectos en el futuro.

Específicamente se ha considerado que una tienda web (de objetos de fieltro en nuestro caso) cumple los objetivos pedidos y es un ejemplo muy significativo dentro de este campo del desarrollo informático.

1.2. OBJETIVOS DEL TRABAJO

1.2.1. OBJETIVOS TÉCNICOS

Se pretende que el proyecto sea multiplataforma por lo que se será accesible a través de un entorno web. De esta manera será accesible a través de cualquiera de los navegadores que hay disponibles en el mercado y sobre cualquier plataforma: Windows, Linux, Mac, ...

También se pretende disfrutar de las ventajas del uso de un entorno web como la facilidad de instalación (o ausencia de la misma) en la parte cliente, el consumo mínimo de recursos, portabilidad, ...

Para fomentar la escalabilidad, reusabilidad y propensión a los cambios del proyecto en la

parte servidor, se utilizará una filosofía de programación basado en el modelo MVC (Modelo - Vista – Controlador).

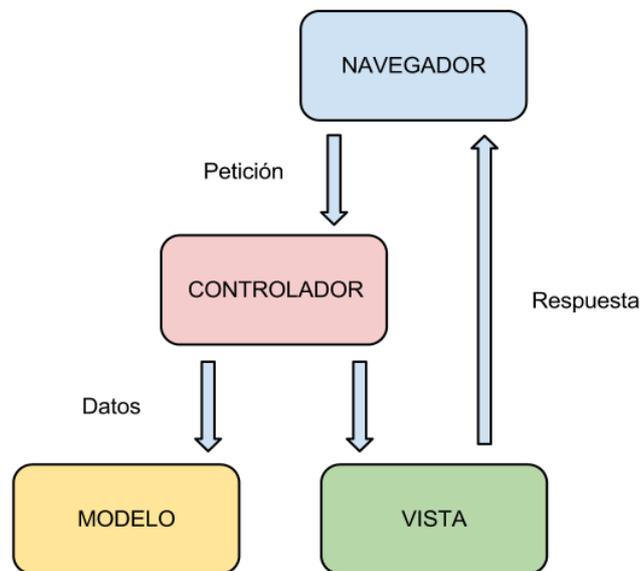


IMAGEN 2.

EL desarrollo se hará en Java J2EE (Java Enterprise Edition) bajo el IDE (Entorno de Desarrollo Integrado) NetBeans IDE 7.0.1.

Dentro de NetBeans se utilizará JDK 1.7 para la compilación de los diferentes paquetes del proyecto.

También se configurará el entorno IDE con Apache Tomcat para poder ejecutar el proyecto y probar su completa funcionalidad.

También se utilizará la librería de J2EE JSTL 1.1 (JavaServer Pages Standard Tag Library) que extiende la conocida librería JSP (Java Server Pages) proporcionando utilidades ampliamente utilizadas en el desarrollo de páginas web dinámicas.

Se utilizará la API de J2EE Servlets para manejar peticiones HTTP y generar respuestas en HTML de manera dinámica. Además, también se pretende.

Para implementar la base de datos se utilizará el SGBD (Sistema Gestor de Bases de Datos) MySQL. A éste lo utilizará la aplicación para mantener toda la información necesaria en el proyecto. Para realizar el acceso a la base de datos a través de Java se utilizará el framework JPA (Java Persistence API).

1.2.2. OBJETIVOS FUNCIONALES

Dentro del “*Subsistema de comercialización de productos*” se tendrán las siguientes funcionalidades:

- Visualización de todas las familias de productos a la venta.
- Visualización de fotografía de cada una de las familias de productos.
- Visualización de descripciones de cada una de las familias de productos.

- Acceso con credenciales. Login.
- Salida del sistema (con las credenciales proporcionadas). Logout.
- Comprar productos.
- Visualizar productos adquiridos.
- Eliminar productos adquiridos.
- Registro de un nuevo usuario cliente.

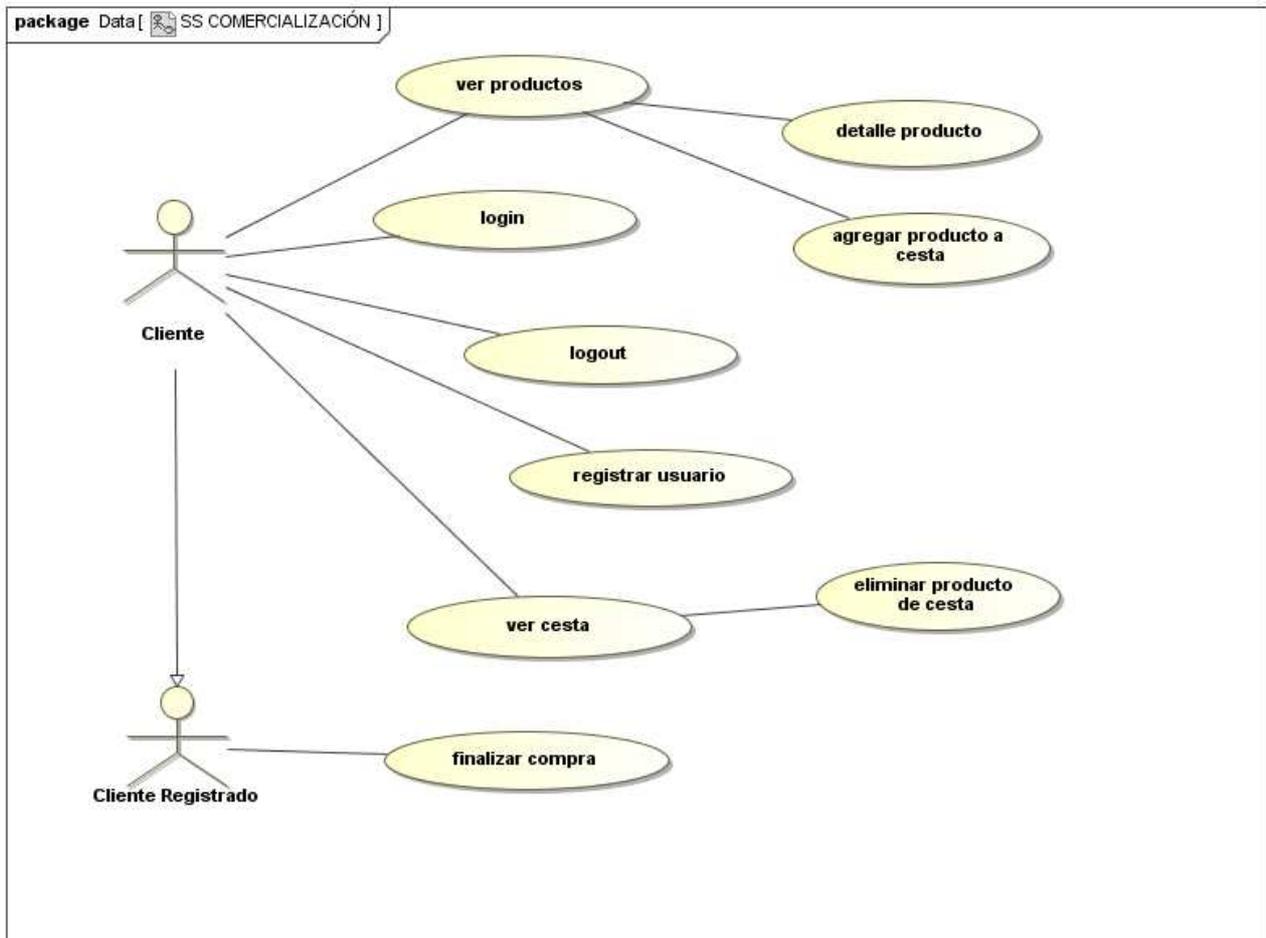


IMAGEN 3.

Dentro del “Subsistema gestor de usuarios” tendremos las siguientes funcionalidades:

- Nuevo usuario
- Ver usuarios del sistema
- Editar usuarios
- Eliminar usuarios

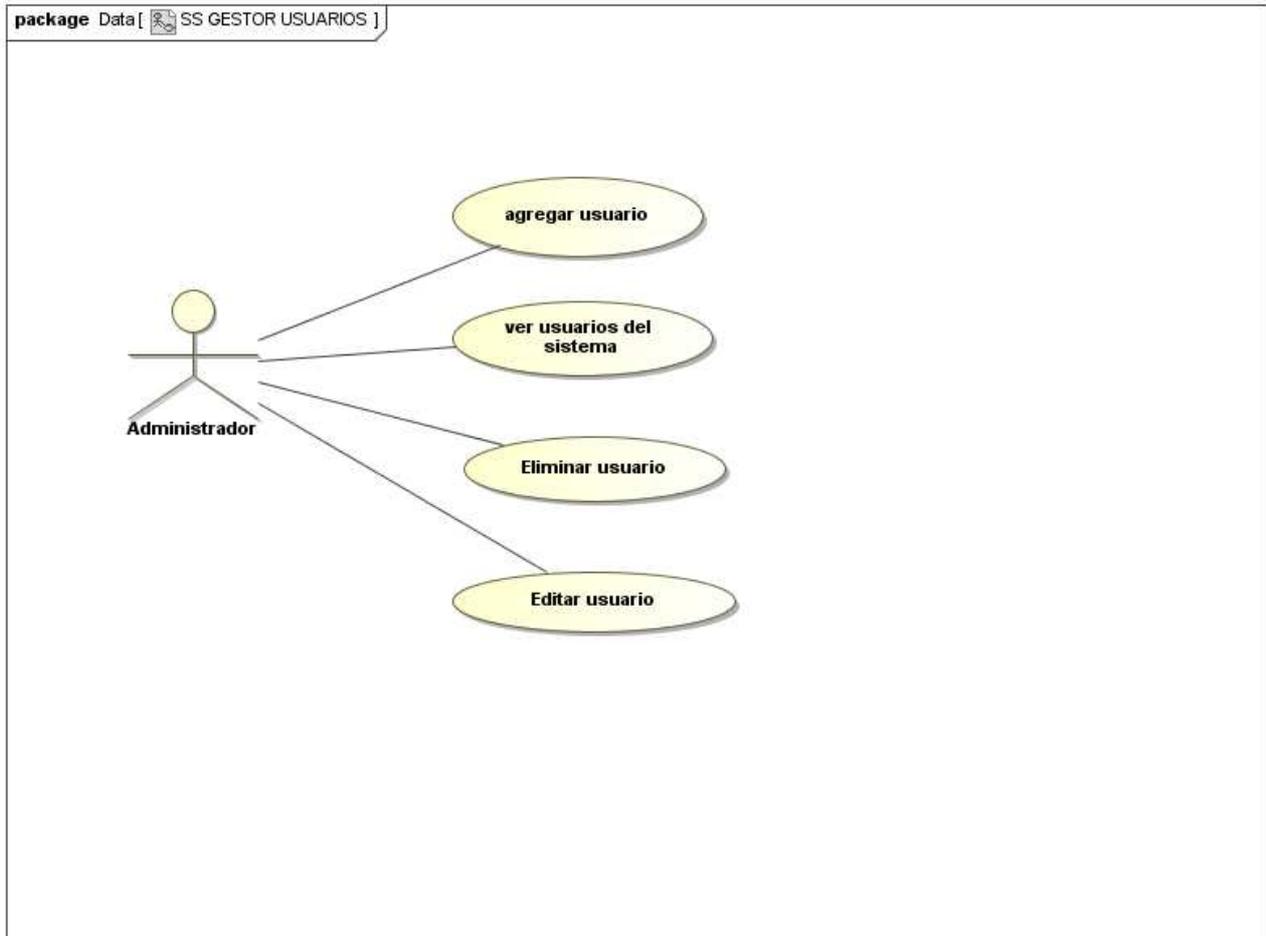


IMAGEN 4.

Dentro del “Subsistema *gestor de productos*” tenemos las siguientes funcionalidades:

- Agregar familia
- Ver familias
- Editar familias
- Borrar familias
- Agregar producto a familia
- Alta clasificación
- Editar clasificación
- Borrar clasificación
- Ver clasificaciones
- Ver productos
- Logout

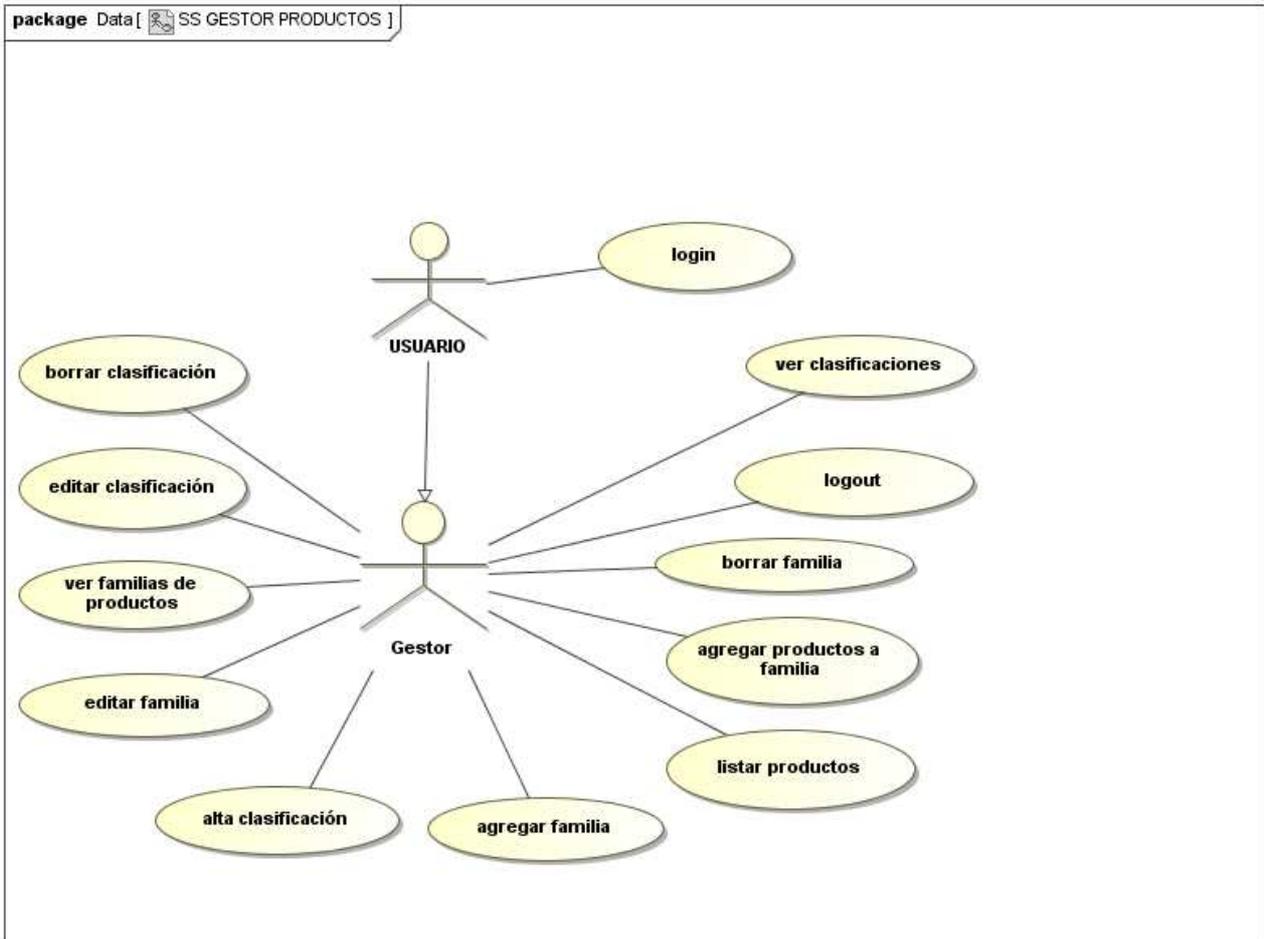


IMAGEN 5.

1.3. ENFOQUE METODOLÓGICO

El enfoque metodológico que se va a emplear en el desarrollo del proyecto está basado en el modelo tradicional en cascada (ver imagen 6).

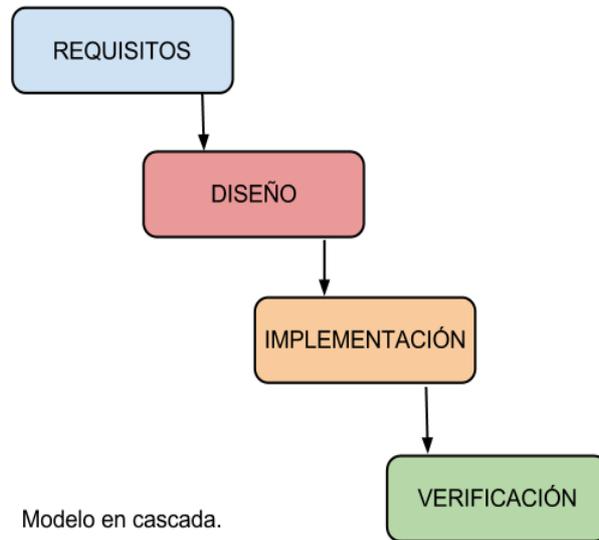


IMAGEN 6.

Sin embargo, debido a las debilidades que ofrece este modelo se permitirá la retroalimentación entre capas. De esta manera se pretende alcanzar un enfoque metodológico más real puesto que nos podremos adaptar de esta manera a las dificultades y cambios que surjan durante las fases posteriores y la manera que éstas pueden modificar el resultado obtenido en capas anteriores.

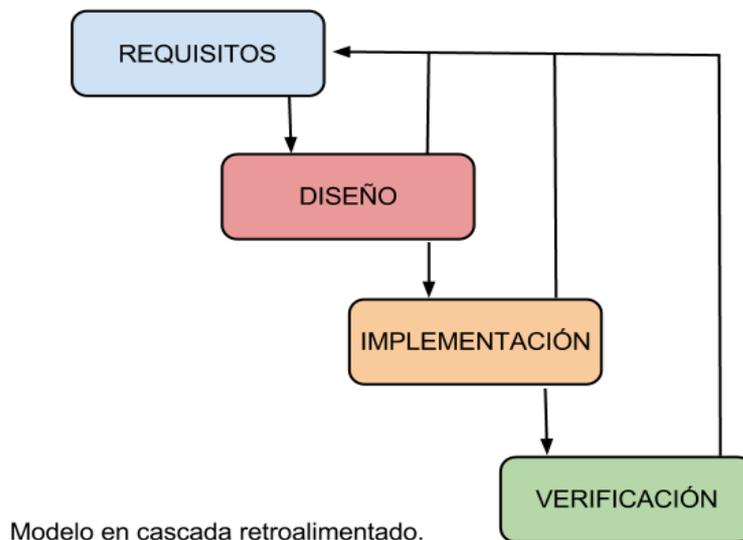


IMAGEN 7.

1.4. PLANIFICACIÓN DEL PROYECTO

1.4.1. ETAPAS

- ANÁLISIS.
 - Requerimientos del cliente.
 - Necesidades y estrategias.
 - Objetivos del proyecto.
 - Organización del trabajo.
 - Control y revisión.
- DISEÑO.
 - Diseño de clases y objetos.
 - Diseño de base de datos.
 - Diseño de interfaz.
 - Control y revisión.
- IMPLEMENTACIÓN.
 - Implementación base de datos.
 - Implementación modelo.
 - Implementación controlador.
 - Implementación vista.
 - Control y revisión.
- PRUEBAS.
 - Testing.
 - Control y revisión.
- DOCUMENTACIÓN.
 - Elaboración de la memoria.
 - Elaboración de la presentación.
 - Control y revisión.
- PRESENTACIÓN DEL PROYECTO.

1.4.2. DIAGRAMA DE GANTT

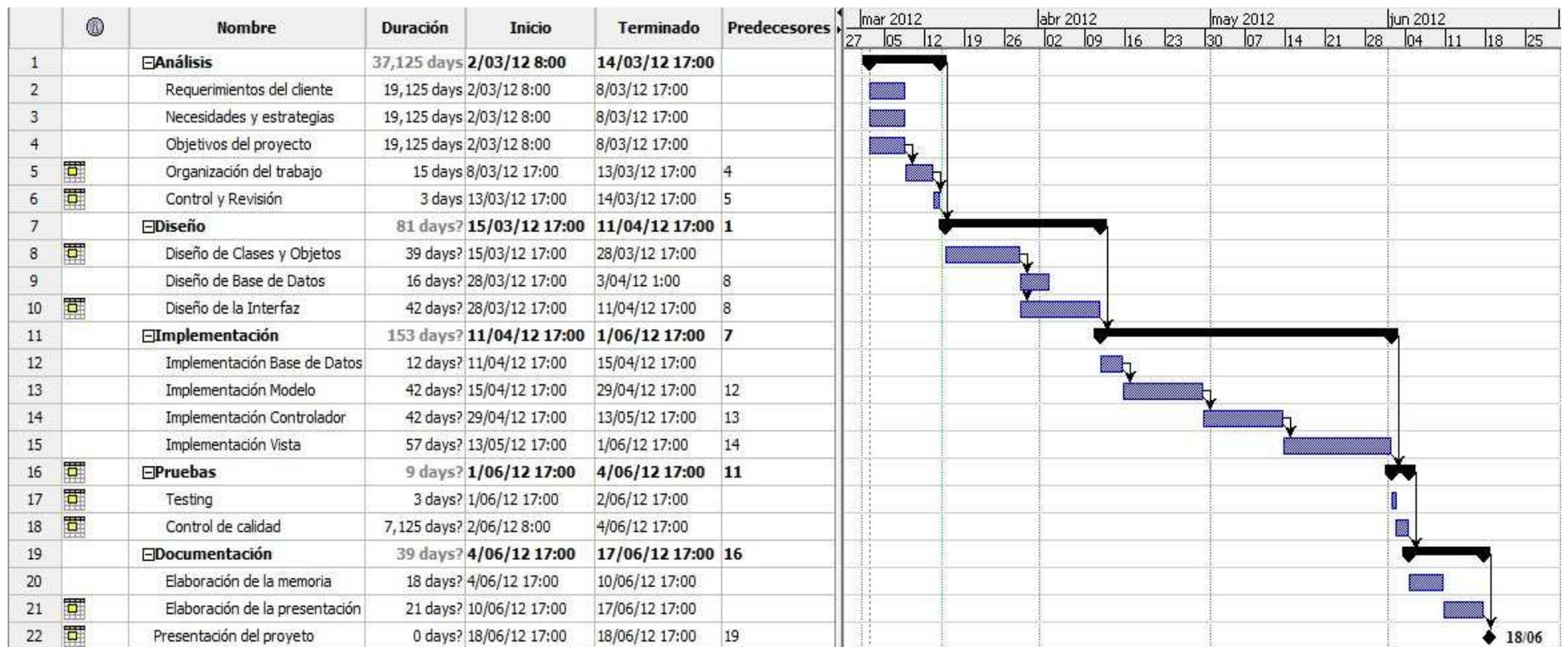


IMAGEN 8.

1.5. PRODUCTO OBTENIDO

Cuando el proceso de creación de la aplicación haya concluido se pretende obtener los siguientes productos:

- Aplicación Mofm: Paquetes java (con sus clases), páginas jsp, base de datos, datos.
- Documentación: Archivos java con la implementación de las clases del sistema, memoria completa del proyecto realizado.
- Presentación del proyecto: Documento para la presentación visual del proyecto que sintetice el trabajo realizado durante todas las fases del proyecto.

2. ESPECIFICACIÓN Y REQUERIMIENTOS

2.1. INFORMACIÓN INICIAL

Cuando un usuario se conecta a la web de la tienda de fieltro (Mofm) puede visualizar todos los productos que hay disponibles a la venta en la tienda. Podrá ir comprando productos y añadiéndolos a su cesta aunque no podrá finalizar su compra si antes no se ha registrado y ha facilitado sus datos personales: dni, dirección, nombre, apellidos, usuario y contraseña.

En cualquier momento también podrá cancelar cualquiera de los productos que tenga agregados a la cesta.

Un usuario puede solicitar ver información más detallada de un producto.

El usuario entonces podrá acceder al sistema y finalizar la compra de sus productos.

Un administrador podrá ver todos los usuarios del sistema así como agregar nuevos, editar cualquier información de alguno de ellos o eliminar un usuario.

Un gestor de productos podrá ver, modificar o eliminar una familia de productos para que se muestre o no dentro de los productos que se venden en la tienda.

Una familia de productos deberá tener productos asociados a él. Cuando un usuario compra un producto está comprando uno de los productos asociados a una familia de productos.

Cada producto comprado será único y tendrá que identificarse para poder hacer un seguimiento de quién lo compró y cuándo.

Cada familia de productos tendrá una foto, una descripción y un precio y estará clasificado de manera que una clasificación podrá contener muchas familias de productos.

Un gestor de productos también podrá ver, modificar, eliminar y agregar nuevas clasificaciones de productos.

2.2 GLOSARIO

- Administrador: Es un usuario que tiene permiso para realizar ciertas tareas de administración como la gestión íntegra de todos los usuarios dados de alta o la

eliminación de los mismos.

- Cesta: Una cesta es el elemento contenedor donde un usuario podrá ir agregando y almacenando los productos a los que tenga interés de compra. Una vez que están almacenados también podrán ser quitados de la misma.
- Clasificación: Una clasificación es la agrupación de diferentes familias de productos que tienen una o algunas características en común. Cada una de estas agrupaciones tendrá un nombre y los usuarios gestores de productos serán los encargados de asociarles productos, agregar y denominar nuevas agrupaciones o eliminar algunas de ellas.
- Cliente: Un cliente es un usuario que accede a la tienda virtual con la finalidad de adquirir alguno de los productos. Aunque no necesariamente necesita llevar a cabo una compra.
- Compra: Se produce cuando un usuario cliente tiene interés en adquirir un producto y procede a su adquisición. Cabe destacar dos acepciones del término: Cuando un usuario cliente agrega un producto de su interés a la cesta, y cuando un usuario cliente está de acuerdo con los productos que ha agregado a la cesta para adquirir e intenta formalizar la adquisición.
- Contraseña: Es una cadena de caracteres que sólo conoce un usuario cliente y que puede ser diferente para cada uno de ellos, y que junto al nombre de usuario identifica inequívocamente a cualquier usuario del sistema.
- Descripción: Es una definición extensa de cómo es y qué cualidades tiene una familia de productos.
- Familia de productos: Es una agrupación de todos los productos que son iguales y que se venden bajo una misma denominación. Estas agrupaciones tienen un nombre y unos atributos que las distinguen entre ellas.
- Foto de producto: Es una imagen tomada de uno de los productos que forman una familia de productos.
- Gestor de productos: Es un usuario que tiene permiso para acceder a ciertas funcionalidades que otros no pueden. Este usuario podrá gestionar las clasificaciones, gestionar las familias de productos y gestionar los productos asociados a cada familia.
- Precio de producto: Es la cantidad de dinero que cuesta una familia de productos y que tendrá que ser abonada por un cliente para la compra de uno de los productos asociados a esa familia.
- Producto: Es cada una de las unidades que se ponen a la venta en la tienda para que sean adquiridas por un usuario cliente y que juntas forman una familia de productos.
- Registro: Es la acción que realiza un usuario cliente para almacenar sus datos personales en el sistema y que les sea proporcionado un login y una contraseña que le permitirá acceder al sistema y poder finalizar una compra.
- Sistema: Se refiere a la tienda virtual y a todos los elementos que la forman y que la hacen funcionar correctamente.

- Usuario: Es cualquier persona que puede interactuar con los elementos del sistema.

2.3 MODELO DEL DOMINIO

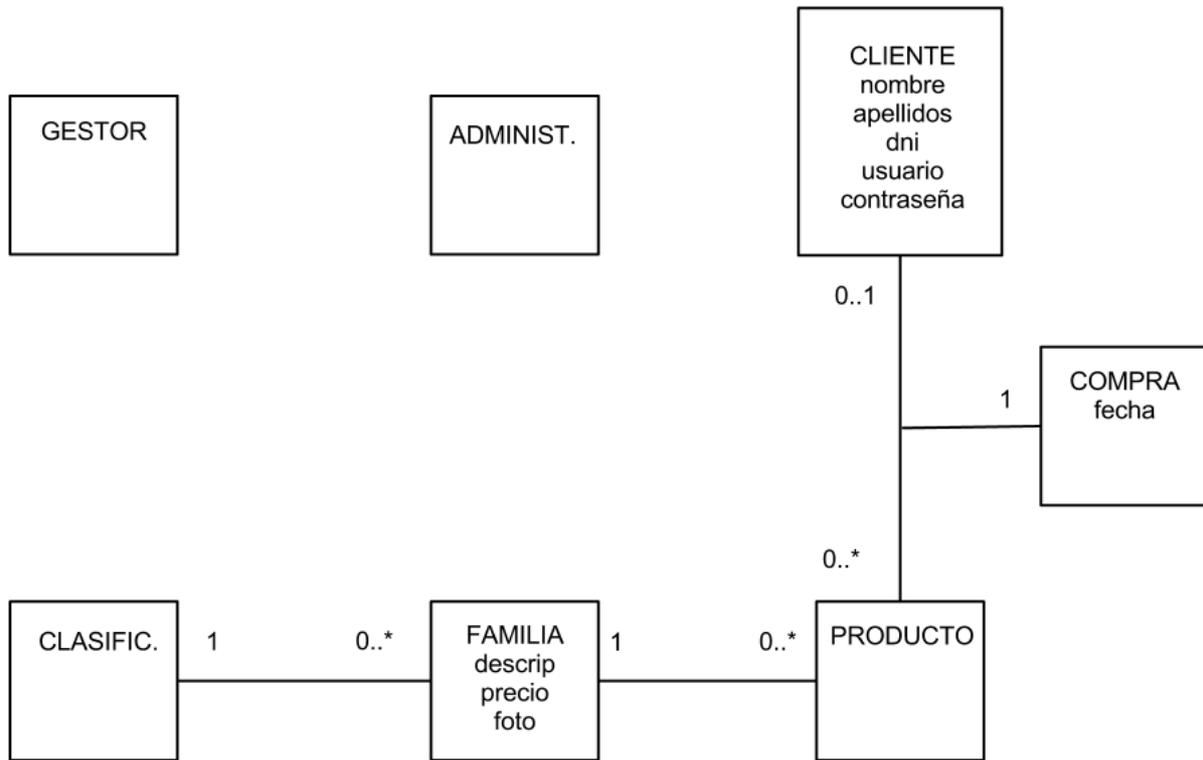


IMAGEN 9.

2.4 DIAGRAMA DE CASOS DE USO

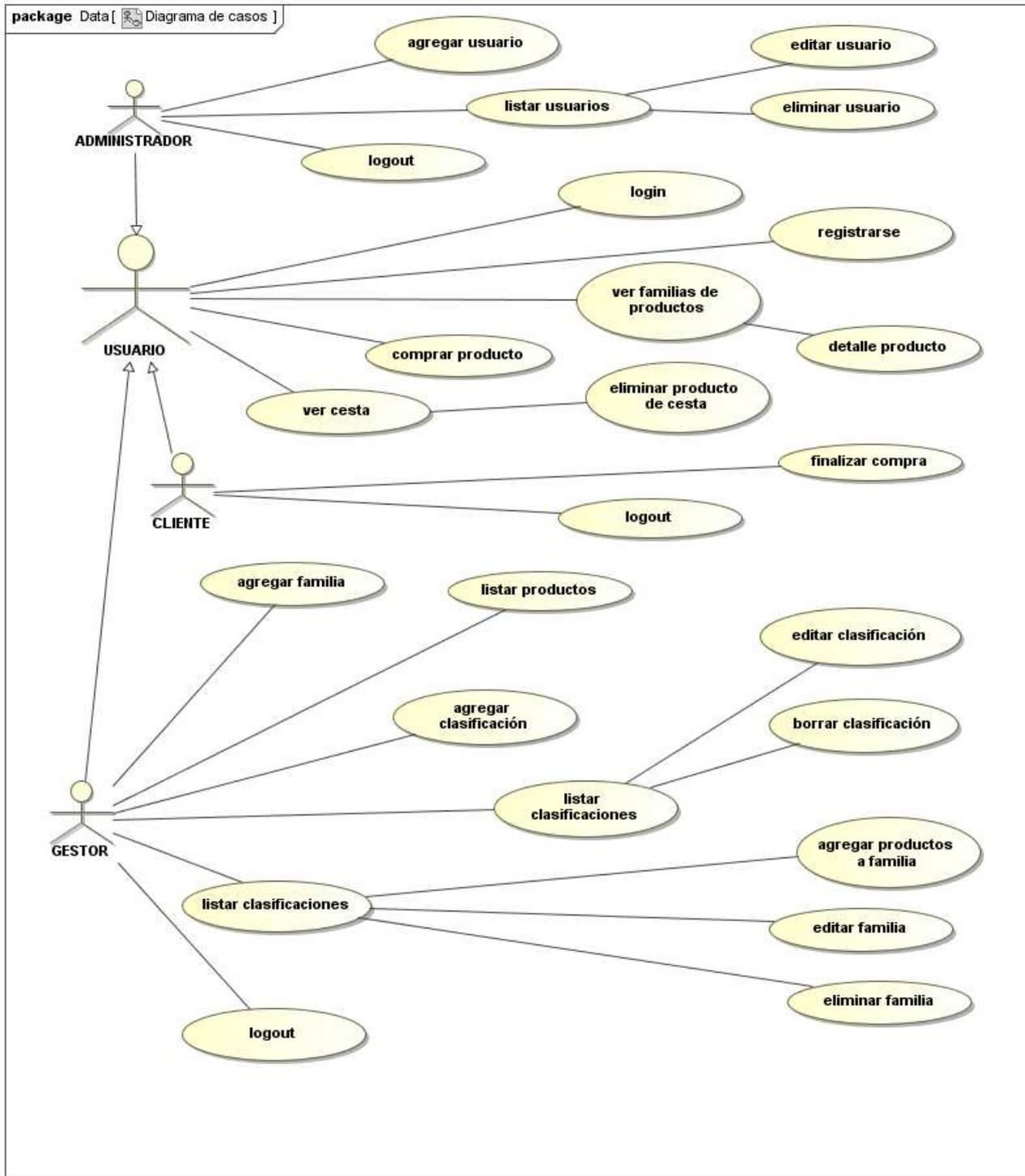


IMAGEN 10.

2.5 DOCUMENTACIÓN TEXTUAL CASOS DE USO

2.5.1. CONEXIÓN CON CREDENCIALES (LOGIN)

Cuando un usuario accede a la aplicación puede, a través de la opción de “Login”, acceder utilizando sus credenciales (si previamente se ha dado de alta o un administrador lo ha hecho por él).

De esta manera, podrá acceder a los servicios personalizados que le ofrece la aplicación que varían según el rol asignado al usuario. Y en el caso de que sea cliente, también podrá finalizar la compra adquiriendo los productos que haya agregado a su cesta hasta ese momento.

Para realizar la conexión con credenciales pulsará la opción “Login”, después la aplicación le mostrará una pantalla para que introduzca su usuario y contraseña y confirme el acceso.

Una vez que el usuario haya insertado sus credenciales, el sistema le mostrará un mensaje informándole de la realización correcta o incorrecta de la operación.

El sistema dispondrá al principio de un usuario registrado “admin” con contraseña “admin1234” que será facilitado al administrador del sistema correspondiente. Pudiendo éste, acceder por primera vez a la aplicación y realizar los cambios que considere oportunos.

Debido a que sin este usuario no se podrían crear nuevos usuarios con el rol de administradores o gestores de productos.

- *Resumen:* Acceder al sistema con credenciales.
- *Actores:* Cliente, administrador, gestor de productos.
- *Casos de uso relacionados:* Ninguno.
- *Precondición:* Exista el usuario en la base de datos.
- *Postcondición:* Puede acceder a nuevas funcionalidades.

2.5.2. REGISTRO DE CLIENTE

Un alta de cliente se produce cuando un usuario accede a la aplicación y desea acceder a las opciones que tienen los usuarios registrados: finalizar la compra de productos agregados a su cesta.

Para ello pulsará en la opción “Registrarse” y la aplicación le mostrará un formulario para que introduzca sus datos: Usuario, contraseña, nombre, dni, dirección, ...

Una vez confirmados los datos, el usuario pasa a ser un usuario registrado del sistema y puede acceder a través de la opción “Login”.

El sistema le mostrará un mensaje informando de la realización correcta de la operación.

- *Resumen:* Agregar un nuevo cliente a la base de datos.
- *Actores:* Usuario.
- *Casos de uso relacionados:* Ver usuarios, eliminar usuario, agregar usuario.

- *Precondición:* No existe el cliente en la base de datos.
- *Postcondición:* Se puede logear.

2.5.3. VER PRODUCTOS

Cualquier usuario que acceda a la aplicación, esté o no registrado, y haya o no accedido con sus credenciales, puede visualizar los productos que están en venta.

Para ello se le muestran siempre a través de la parte principal de la pantalla de bienvenida.

Además, siempre que esté en otra pantalla, podrá volver a ver los productos pulsando en cualquier elemento del menú que muestra las clasificaciones y las familias de productos a la venta. Y que está siempre visible en la derecha de la pantalla.

- *Resumen:* Ver listado de productos.
- *Actores:* Usuario.
- *Casos de uso relacionados:* Ninguno.
- *Precondición:* Ninguna.
- *Postcondición:* Ninguna.

2.5.4. DETALLE PRODUCTOS

En cualquier momento un usuario que acceda a la aplicación, esté o no registrado, y haya o no accedido con sus credenciales, puede ver el detalle de uno de los productos que están en venta.

Pulsando sobre la opción correspondiente cerca de la imagen del producto, la aplicación mostrará una imagen más grande del producto y una descripción más detallada del mismo.

- *Resumen:* Ver descripción detallada de un producto.
- *Actores:* Usuario.
- *Casos de uso relacionados:* Ninguno.
- *Precondición:* Ninguna.
- *Postcondición:* Ninguna.

2.5.5. AGREGAR PRODUCTO A CESTA

Un cliente esté o no registrado, mientras está visualizando los productos a la venta, podrá pulsar la opción de comprar producto.

Una vez que un cliente registrado compra un producto, éste se agrega a su cesta personal. De manera que cuando acceda a la opción de ver su cesta, podrá ver todos los productos comprados.

- *Resumen:* Agregar un nuevo producto a la cesta.

- *Actores:* Usuario.
- *Casos de uso relacionados:* Ver cesta y quitar producto de cesta.
- *Precondición:* Exista el producto en la base de datos.
- *Postcondición:* Se podrá ver el producto en la cesta.

2.5.6. VER CESTA

Un cliente esté o no registrado podrá visualizar su cesta accediendo a través de la opción correspondiente del menú. Una vez que acceda, la aplicación le mostrará todos los productos que ha adquirido hasta el momento.

Una vez que está dentro de la cesta, el cliente podrá también eliminar cualquiera de los productos que haya adquirido hasta ese momento. De manera que este producto desaparecerá de la cesta.

También podrá seguir comprando a través de las opciones correspondiente del menú de productos.

Una vez que considere que tiene los productos que necesita, el cliente registrado tendrá la opción de finalizar compra. De esta manera formaliza su compra y los productos pasan al estado “comprados”, y desaparecen de la cesta.

- *Resumen:* Ver el listado de productos agregados a la cesta.
- *Actores:* Usuario.
- *Casos de uso relacionados:* Ninguno.
- *Precondición:* Ninguna.
- *Postcondición:* Ninguna.

2.5.7. QUITAR PRODUCTO DE CESTA

Como ya se ha comentado, mientras que un cliente registrado visualiza su cesta tendrá la opción de eliminar el producto de la misma.

Al quitar el producto se elimina el producto de la lista de productos de la cesta no volviendo a aparecer cuando se visualiza ésta.

Se considera que no es necesario mantener constancia de estos productos puesto que pueden ser adquiridos por error del usuario.

- *Resumen:* Eliminar producto de la cesta.
- *Actores:* Usuario.
- *Casos de uso relacionados:* Ver cesta.
- *Precondición:* Exista el producto en la cesta.
- *Postcondición:* Ninguna.

2.5.8. FINALIZAR COMPRA

Como ya se ha comentado, cuando un cliente registrado visualiza su cesta y considera que son los productos que desea adquirir, puede finalizar su compra pulsando en la opción correspondiente que se le muestra en la pantalla.

De esta manera formalizará su compra.

Todos los productos adquiridos pasarán a tener el estado “comprado” por lo que no se volverán a mostrar en la cesta del usuario.

- *Resumen:* Finalizar la compra de productos.
- *Actores:* Cliente logeado.
- *Casos de uso relacionados:* Listar productos.
- *Precondición:* Existan productos en la cesta.
- *Postcondición:* No se pueden volver a adquirir los productos.

2.5.9. DESCONECTAR (LOGOUT)

Cualquier usuario registrado; administrador, cliente o gestor podrá, a través de la opción correspondiente del menú, cerrar su sesión.

Una vez que cierra la sesión se dejarán de mostrar las opciones personalizadas correspondientes a su usuario registrado. Aunque podrá seguir visualizando los productos a la venta.

- *Resumen:* Borrar el usuario logeado del sistema.
- *Actores:* Gestor de productos, Administrador y Cliente.
- *Casos de uso relacionados:* Ninguno.
- *Precondición:* Exista un usuario logeado.
- *Postcondición:* Se pierden los datos de la sesión.

2.5.10. LISTAR USUARIOS

Solamente un administrador de sistema tendrá acceso a la opción de listar usuarios. Para ello deberá haber accedido con sus credenciales correspondientes.

El sistema le mostrará un listado con todos los usuarios registrados del sistema.

Por cada usuario del listado, la aplicación, ofrecerá la opción de borrarlo o editarlo.

- *Resumen:* Ver un listado de los usuarios.
- *Actores:* Administrador.
- *Casos de uso relacionados:* Ninguno.
- *Precondición:* Ninguna.
- *Postcondición:* Ninguna.

2.5.11. AGREGAR USUARIO

Un administrador de sistema registrado también podrá agregar nuevos usuarios, sea cual sea el rol de éstos, al sistema.

Para ello, y accediendo a través de la opción correspondiente del menú, el sistema le mostrará un formulario con los datos que se deben rellenar. Además, a diferencia de la opción registrar que puede realizar cualquier usuario, el administrador podrá asignarle uno de los roles existentes: Administrador, gestor o cliente.

No se podrá registrar un usuario con un “dni” que esté ya dentro del sistema.

- *Resumen:* Agregar un nuevo usuario a la base de datos.
- *Actores:* Administrador.
- *Casos de uso relacionados:* Comprar producto, listar usuarios, eliminar usuario y editar usuario.
- *Precondición:* No exista el usuario en la base de datos.
- *Postcondición:* Se puede utilizar el usuario.

2.5.12. EDITAR USUARIO

Un administrador de sistema, como ya se ha comentado, una vez que haya accedido a la opción correspondiente y esté visualizando el listado de usuarios del sistema, podrá editar la información de cualquiera de ellos.

Una vez que lo haga, a través del botón correspondiente del usuario elegido, la aplicación le mostrará un formulario con la información del usuario.

El administrador podrá modificar para actualizar cualquier dato a excepción del “idUsuario”.

Tampoco podrá modificar un “dni” si está tratando de poner uno que coincide con el de un usuario registrado del sistema.

- *Resumen:* Modificar los datos asociados a un usuario.
- *Actores:* Administrador.
- *Casos de uso relacionados:* Comprar producto, listar usuarios y eliminar usuario.
- *Precondición:* Exista el usuario en la base de datos. No exista el dni modificado.
- *Postcondición:* Ninguna.

2.5.13. BLOQUEAR USUARIO (ELIMINAR USUARIO)

Un administrador de sistema, como ya se ha comentado, una vez que haya accedido a la opción correspondiente y esté visualizando el listado de usuarios del sistema, podrá eliminar a cualquiera de ellos.

La política de la aplicación es que no se elimine el usuario de la base de datos, tan sólo se le cambiará el estado a “bloqueado”.

A efectos prácticos, un usuario bloqueado no existe para el sistema. Aunque se mantenga

la información del usuario así como sus compras realizadas y final.

Se seguirá esta política para poder mantener en la base de datos los productos comprados por un usuario aunque este haya decidido darse de baja del sistema poniéndose en contacto con un administrador.

- *Resumen:* Cambiar el estado de un usuario a bloqueado.
- *Actores:* Administrador.
- *Casos de uso relacionados:* Comprar producto, ver usuarios y editar usuarios.
- *Precondición:* Exista el usuario en la base de datos.
- *Postcondición:* No se puede volver a utilizar la familia.

2.5.14. AGREGAR FAMILIA

Un gestor registrado podrá agregar una nueva familia al sistema. Para hacerlo deberá acceder a la opción correspondiente del menú.

Agregar una familia significa que existe un nuevo tipo de producto.

Una vez que se accede el sistema mostrará un formulario donde se pedirá la información necesaria para dar de alta la familia: clasificación, nombre, descripción, URL de las imágenes (pequeña y grande), precio, ...

- *Resumen:* Agregar una nueva familia.
- *Actores:* Gestor de productos.
- *Casos de uso relacionados:* Agregar producto.
- *Precondición:* No existe la familia en la base de datos.
- *Postcondición:* Se puede utilizar la familia agregada.

2.5.15. VER FAMILIAS

Un gestor registrado podrá ver todas las familias disponibles en el sistema.

La aplicación le mostrará un listado con todas las familias de productos disponibles en el sistema. Además al lado de cada familia ofrecerá las siguientes opciones: agregar productos, borrar familia y editar familia.

- *Resumen:* Ver un listado de familias.
- *Actores:* Gestor de productos.
- *Casos de uso relacionados:* Ninguno.
- *Precondición:* Ninguna.
- *Postcondición:* Ninguna.

2.5.16. AGREGAR PRODUCTOS A FAMILIA

Un gestor, como ya se ha comentado, una vez que haya accedido a la opción

correspondiente y esté visualizando el listado de familias del sistema, podrá agregar productos a esa familia.

Para ello debe pulsar en la opción correspondiente dentro de la pantalla ver familias. Una vez hecho esto el sistema creará un nuevo id para el producto que se le ha indicado y lo agregará a la base de datos. De esta manera quedará disponible para que pueda ser adquirido por cualquier cliente.

Un producto sólo podrá dejar de estar disponible para su compra si es adquirido por un cliente.

Con este identificador de producto se podrá identificar el producto de manera que se podrá hacer un seguimiento en caso de que hubiera algún problema tras su venta.

El identificador de producto irá también de manera física en el producto.

- *Resumen:* Agregar nueva familia a la base de datos.
- *Actores:* Gestor de productos.
- *Casos de uso relacionados:* Agregar producto.
- *Precondición:* No existe la familia en la base de datos.
- *Postcondición:* Se puede utilizar la familia agregada.

2.5.17. EDITAR FAMILIA

Un gestor, como ya se ha comentado, una vez que haya accedido a la opción correspondiente y esté visualizando el listado de familias del sistema, podrá editar la información de una familia.

Para ello se le mostrará un formulario con la información de la familia de manera que el gestor pueda modificar algún dato en el caso de que fuera necesario.

- *Resumen:* Editar los atributos de una familia.
- *Actores:* Gestor de productos.
- *Casos de uso relacionados:* Agregar producto.
- *Precondición:* Exista la familia en la base de datos.
- *Postcondición:* Ninguna.

2.5.18. BORRAR FAMILIA (BLOQUEAR FAMILIA)

Un gestor, como ya se ha comentado, una vez que haya accedido a la opción correspondiente y esté visualizando el listado de familias del sistema, podrá borrar cualquiera de ellas.

Por política de la aplicación la familia no se borra de la base de datos. Solamente se modifica su estado a “bloqueado” de manera que ya no se muestra en el listado de familias de productos disponibles para la venta. Aunque se siguen manteniendo las compras de productos de estas familias que ya se hayan realizado.

- *Resumen:* Cambiar el estado de una familia a bloqueado.

- *Actores:* Gestor de productos.
- *Casos de uso relacionados:* Agregar producto.
- *Precondición:* Exista la familia en la base de datos.
- *Postcondición:* No se puede volver a utilizar la familia.

2.5.19. AGREGAR CLASIFICACIÓN

Un usuario gestor de productos que haya accedido al sistema con sus credenciales podrá agregar una nueva clasificación al sistema. Una vez agregada, ésta podrá ser usada para clasificar a una o varias familias de productos.

- *Resumen:* Agregar una nueva clasificaciones.
- *Actores:* Gestor de productos.
- *Casos de uso relacionados:* Listar familias, agregar familia y editar familia.
- *Precondición:* Exista la clasificación en la base de datos.
- *Postcondición:* Se puede utilizar la nueva clasificación.

2.5.20. EDITAR CLASIFICACIÓN

Un usuario gestor de productos que haya accedido al sistema con sus credenciales podrá editar cualquier clasificación que esté en el sistema modificando su nombre. De esta manera todos las familias de productos asociadas a ella también cambiarán el nombre de su clasificación.

- *Resumen:* Modificar la información de una clasificación.
- *Actores:* Gestor de productos.
- *Casos de uso relacionados:* Añadir familia, listar familias y editar familia.
- *Precondición:* Existe la clasificación en la base de datos.
- *Postcondición:* Ninguna.

2.5.21. LISTAR CLASIFICACIONES

Un usuario gestor de productos que haya accedido al sistema con sus credenciales podrá listar todas las clasificaciones existentes en el sistema.

A través del listado de clasificaciones se ofrecerán las opciones de editar y borrar cualquiera de ellas.

- *Resumen:* Ver listado de clasificaciones.
- *Actores:* Gestor de productos.
- *Casos de uso relacionados:* Ninguno.
- *Precondición:* Ninguna.
- *Postcondición:* Ninguna.

2.5.22. BORRAR CLASIFICACIÓN (BLOQUEAR CLASIFICACIÓN)

Un usuario gestor de productos que haya accedido al sistema con sus credenciales podrá en cualquier momento a través de la opción correspondiente de la pantalla listar clasificaciones borrar cualquier clasificación existente.

Las clasificaciones no se eliminan, sólo se cambiará su estado a bloqueado de manera que puedan mantenerse la clasificación de los productos adquiridos por un cliente.

- *Resumen:* Cambiar estado clasificación a bloqueado.
- *Actores:* Gestor de productos.
- *Casos de uso relacionados:* Listar clasificaciones y listar familias..
- *Precondición:* La clasificación existe en la base de datos.
- *Postcondición:* No se puede volver a utilizar la clasificación.

2.6 REQUISITOS DE LA INTERFAZ DE USUARIO

	FRECUEN	USUARIO	T PREVIA	T POST	ENTRADA	SALIDA	DESTINO	PROBLEM
Login	alta	Todos	Registro	-	Usuario y contraseña	-	-	Usuario o contraseña incorrectos
Registro	media	Cliente	-	-	Datos cliente	-	-	Dni o usuarios duplicados
Compra	alta	Cliente	Alta producto	-	Producto	-	-	Producto agotado
Fin. compra	alta	Cliente	Login	-		-	-	Sin productos en cesta
Ver detalle	alta	Cliente	Alta producto	-	Producto	Datos producto	Cliente	-
Ver productos	alta	Cliente	-	-		Productos	Cliente	-
Ver cesta	alta	Cliente	-	-		Productos agregados cesta	Cliente	-
Quitar producto cesta	media	Cliente	Agregar producto a cesta	-	Producto	-	-	Sin productos en cesta
Agregar producto cesta	alta	Cliente	Alta producto	-	Datos producto	-	-	-
Agregar usuario	baja	Admin.	-	-	Datos usuario	-	-	Dni o usuario duplicados
Editar usuario	baja	Admin.	Alta usuario	-	Usuario	-	-	Dni o usuarios

								duplicados
Borrar usuario	baja	Admin.	Alta usuario	-	Usuario	-	-	-
Listar usuarios	media	Admin.	-	-	-	Usuarios	Admin.	-
Agregar familia	media	Gestor	-	-	Datos familia	-	-	-
Editar familia	baja	Gestor	Alta familia	-	Familia	-	-	-
Listar familias	media	Gestor	-	-	-	Familias	Gestor	-
Borrar familia	baja	Gestor	Alta familia	-	Familia	-	-	-
Agregar clasificac.	media	Gestor	-	-	Nombre clasificac.	-	-	-
Borrar clasificac.	baja	Gestor	Alta clasificac.	-	Clasificac.	-	-	-
Editar clasificac.	media	Gestor	Alta clasificac.	-	Clasificac.	-	-	-
Listar clasificac.	media	Gestor	-	-	-	Clasificac.	Gestor	-
Agregar producto a familia	alta	Gestor	Alta familia	-	Familia	-	-	Familia no existe
Listar productos	media	Gestor	-	-	-	Prodcutos	Gestor	-
Logout	alta	Todos	login	-	-	-	-	No se ha hecho login

3. ANÁLISIS

3.1. IDENTIFICACIÓN DE CLASES Y ATRIBUTOS

3.1.1. USUARIO

Un usuario es cualquier persona que utilice la aplicación MOFM (Made of Felt Market).



IMAGEN 11.

3.1.2. CLIENTE

Se considerará un cliente cualquier persona que acceda a la aplicación web y que se dé de alta creando así un “login” en el sistema. De manera que, además de acceder a ver los productos, puede realizar compras.

3.1.3. ADMINISTRADOR

Un Administrador es un usuario que accede a la aplicación a través de sus credenciales (usuario y contraseña) y al que se le permite realizar labores de administración de usuarios. Como crear o modificar datos de los usuarios del sistema.

3.1.4. GESTOR

Un gestor es un usuario que accede a la aplicación a través de sus credenciales (usuario y contraseña) y al que se le permite realizar labores de gestión de productos y familias. Será el encargado de mantener los productos disponibles en la tienda y que éstos se reflejen perfectamente en la aplicación MOFM.

3.1.5. FAMILIA

Una familia es la generalización de todos los productos iguales. Es decir, todos los productos de una misma forma, tamaño y color (que se podrían considerar un mismo producto) forman una familia. Una determinada familia contendrá una cantidad X de productos que se pueden vender. En la aplicación se mostrarán todas las familias, y un cliente podrá comprar un producto de una familia.

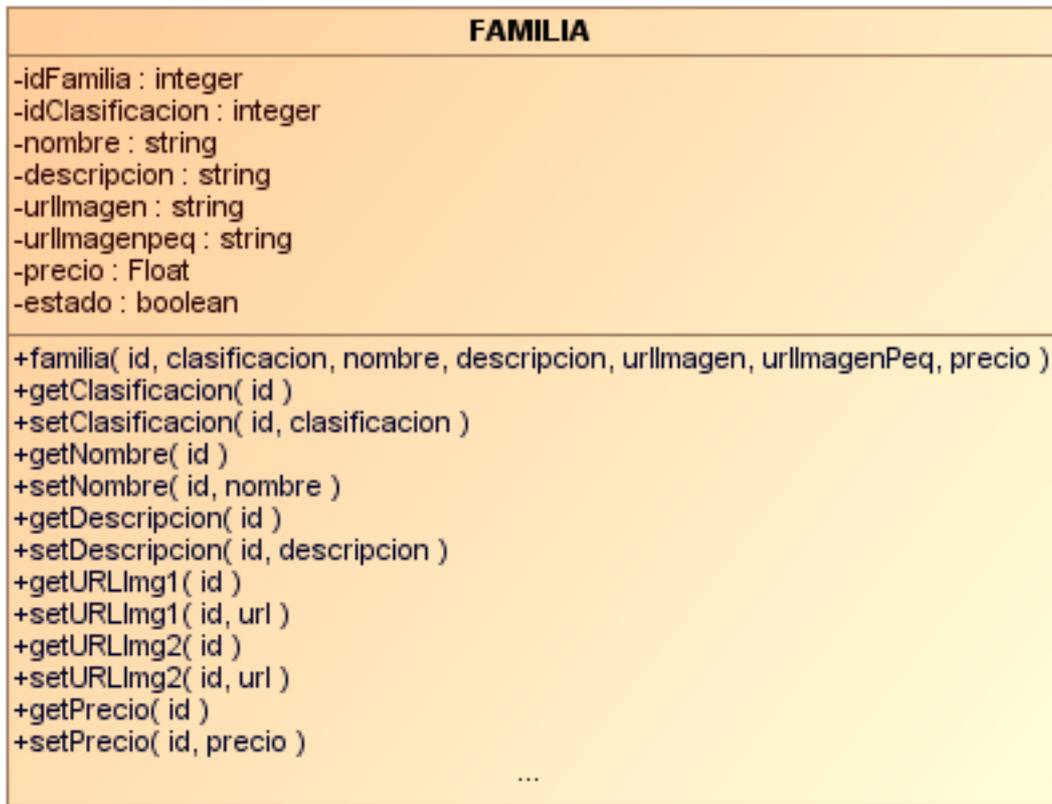


IMAGEN 12.

3.1.6. PRODUCTO

Un producto es una unidad que está a la venta en la aplicación. Todos los productos iguales, forman una familia. Es decir, para una determinada familia de productos existirán X productos o unidades a la venta. Cada producto dispondrá de un identificador único que ayudará a realizar un seguimiento de cada uno de ellos si fuera necesario.



IMAGEN 13.

3.1.7. CLASIFICACIÓN

Una clasificación es una generalización de las familias. Se utilizará para poder mantener las diferentes familias de productos clasificadas de la manera que los gestores de la tienda deseen.



IMAGEN 14.

3.1.8. COMPRA

Una compra es una instancia del tipo Producto-Cliente que representa la compra de ese producto por ese cliente en una fecha determinado. El estado nos dice si se ha finalizado la compra o no. Si no se ha finalizado aparecerá en la cesta del cliente.



IMAGEN 15.

3.1.9. RELACIONES DE HERENCIA

Se considerarán las siguientes relaciones de herencia de atributos:

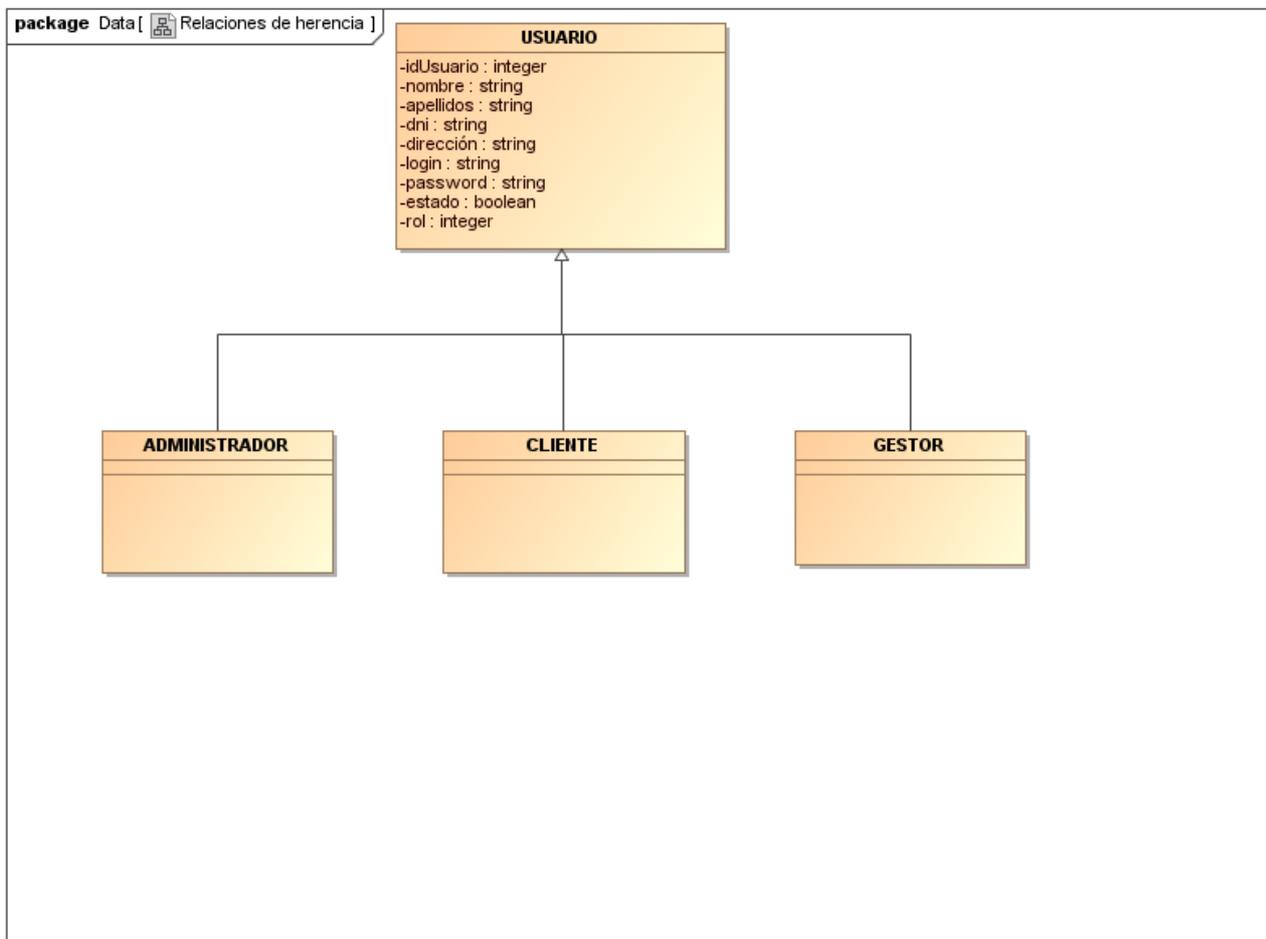


IMAGEN 16.

3.2. DIAGRAMAS DE SECUENCIA

3.2.1. LOGIN DE USUARIO

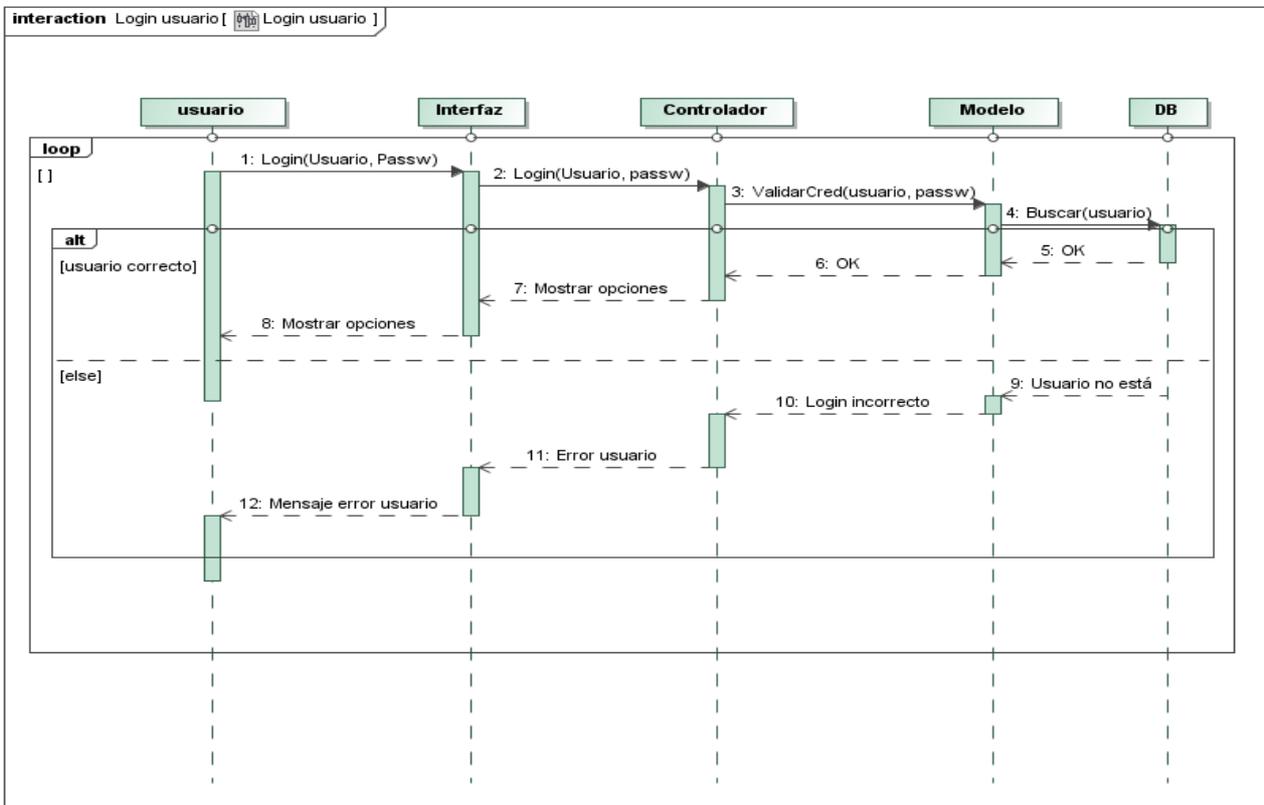


IMAGEN 17.

3.2.2. LOGOUT DE USUARIO

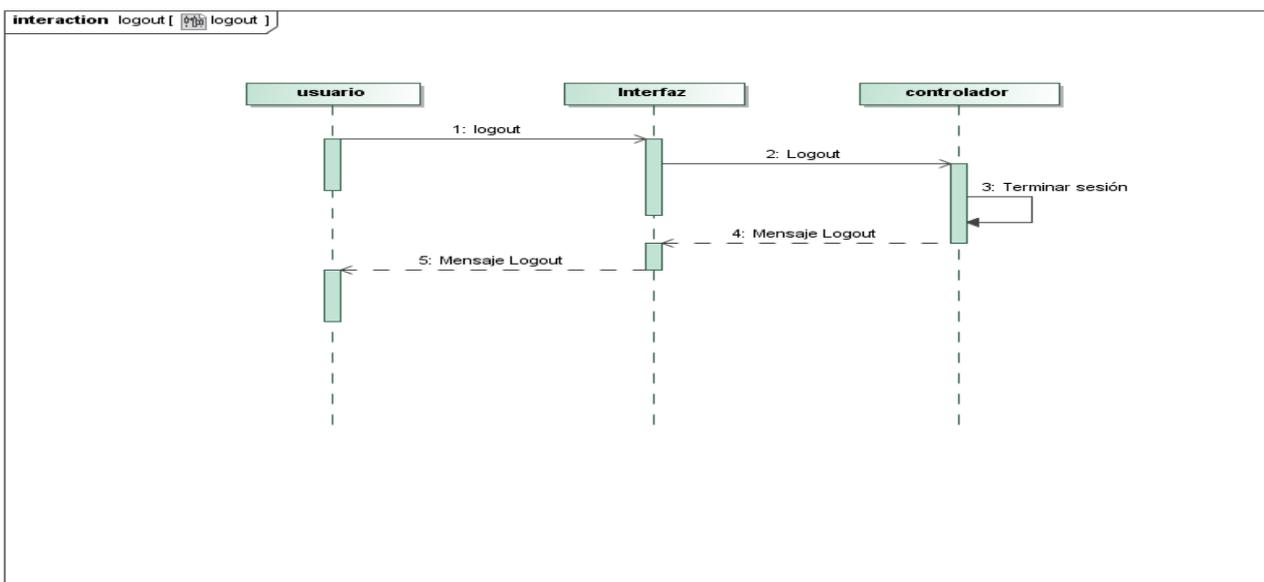


IMAGEN 18.

3.2.2. LISTAR/EDITAR/BORRAR USUARIOS

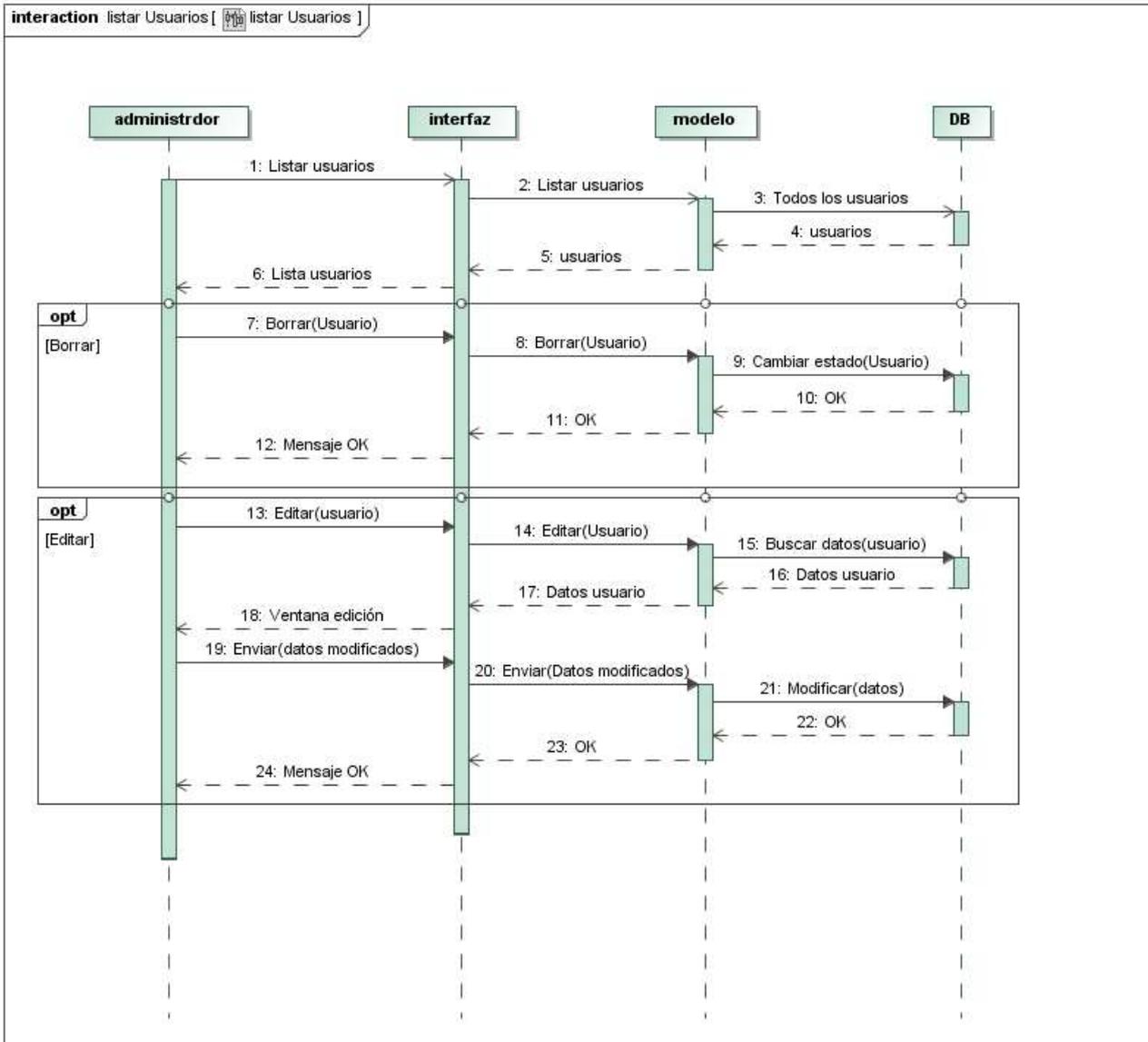


IMAGEN 19.

3.2.4. NUEVO USUARIO

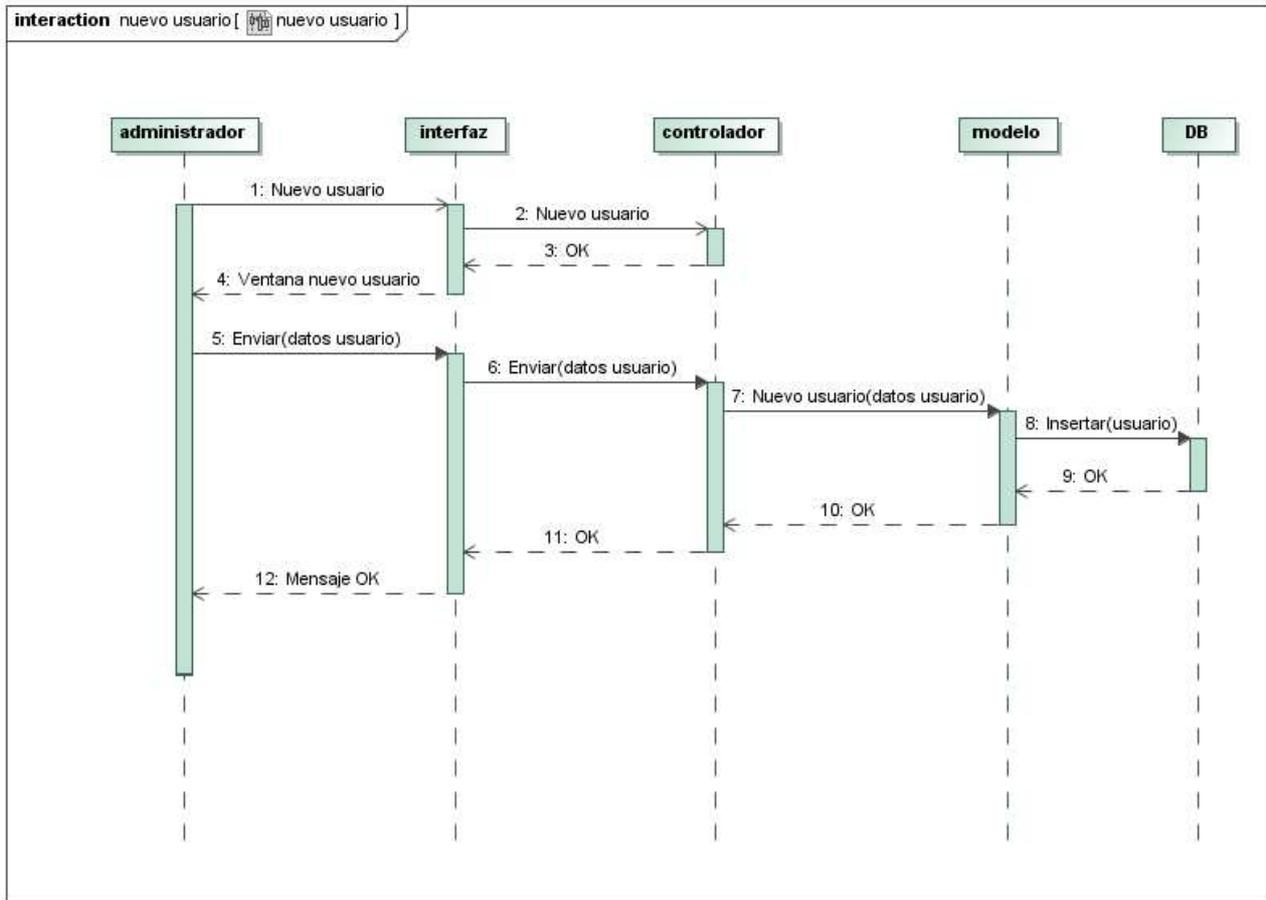


IMAGEN 20.

3.2.5. VER/FILTRAR PRODUCTOS Y VER DETALLE

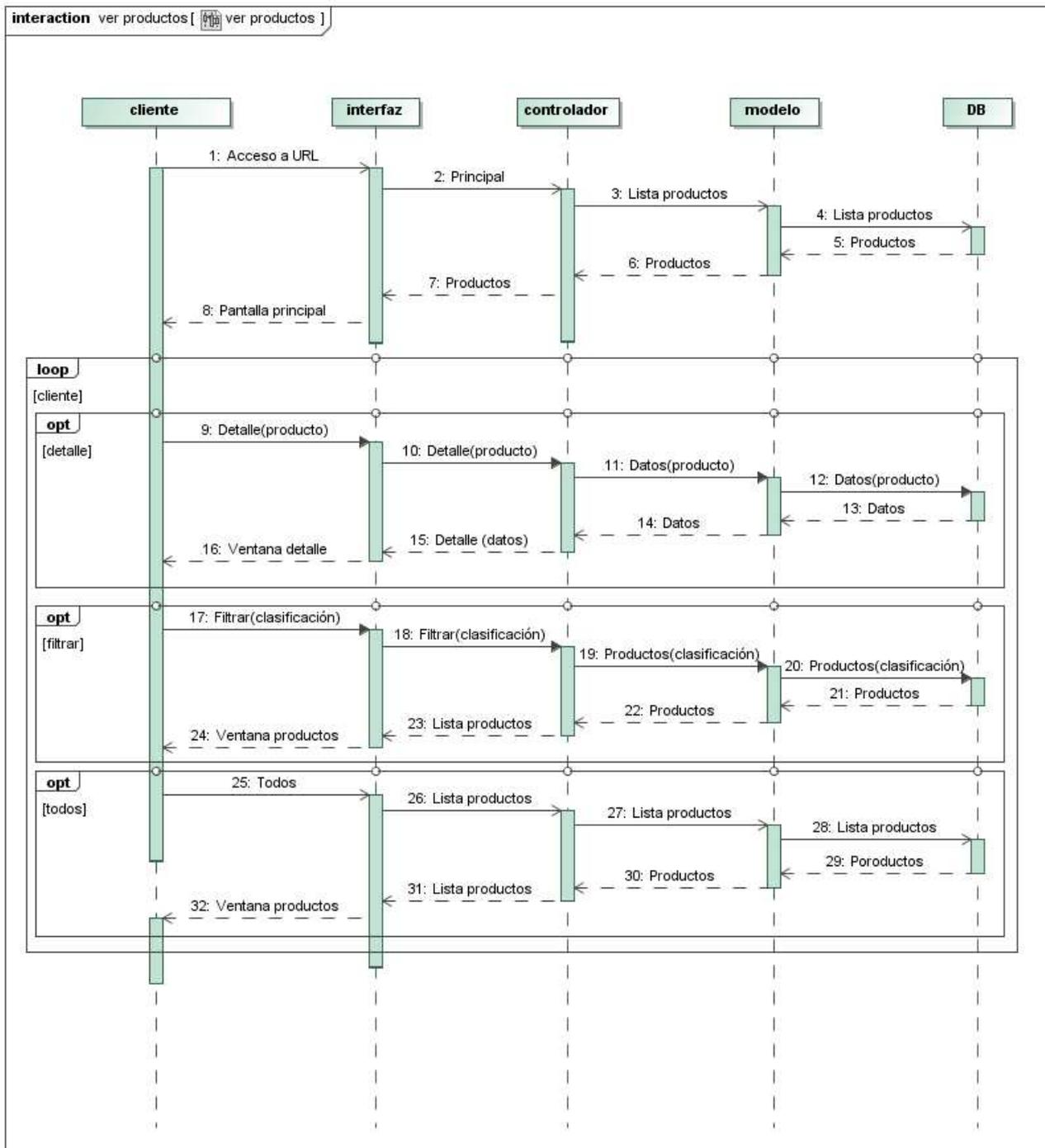


IMAGEN 21.

3.2.6. COMPRAR PRODUCTO/VER CESTA/ELIMINAR PRODUCTO DE CESTA/FINALIZAR COMPRA

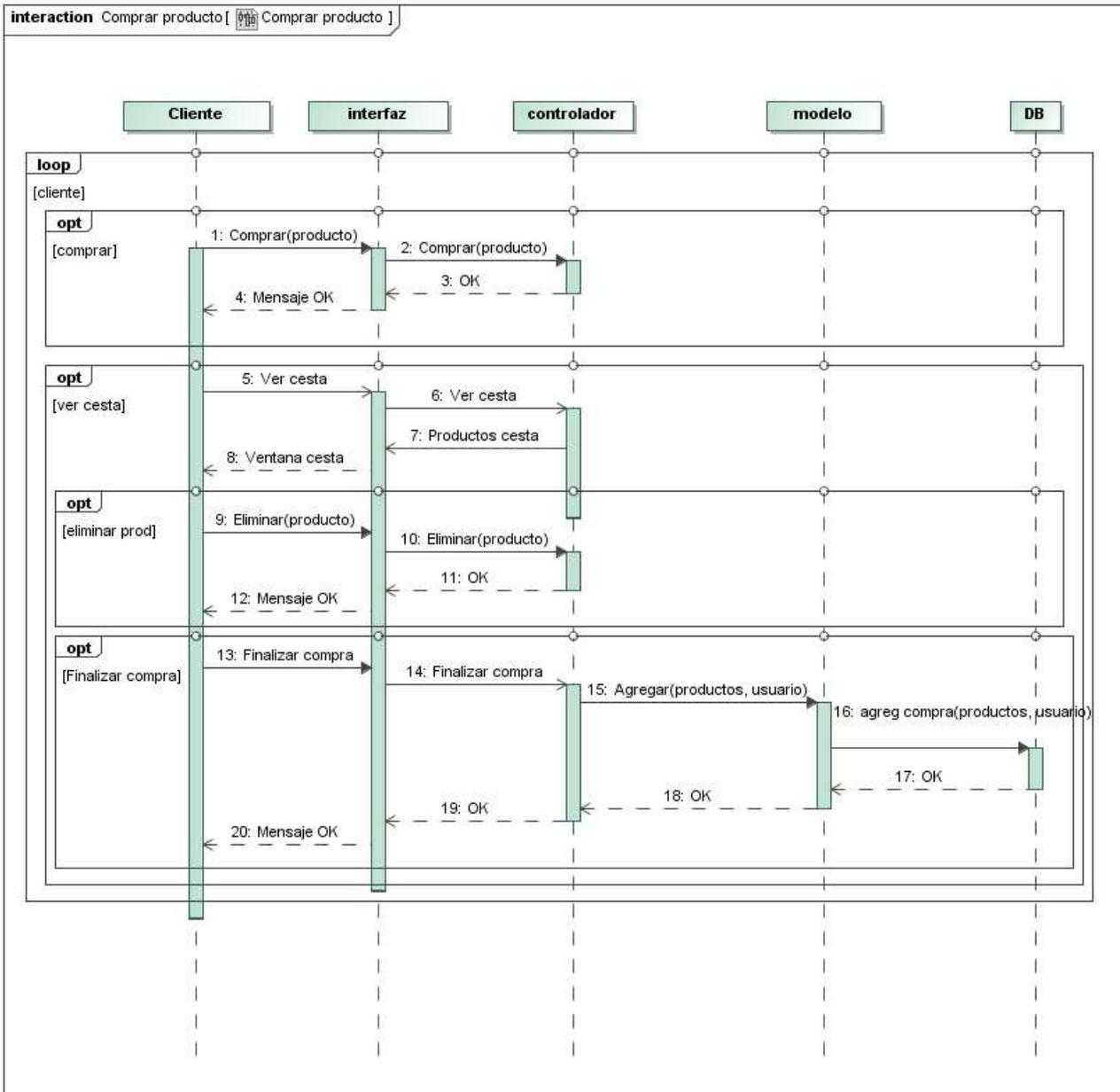


IMAGEN 22.

3.2.7. LISTAR/EDITAR/BORRAR FAMILIA

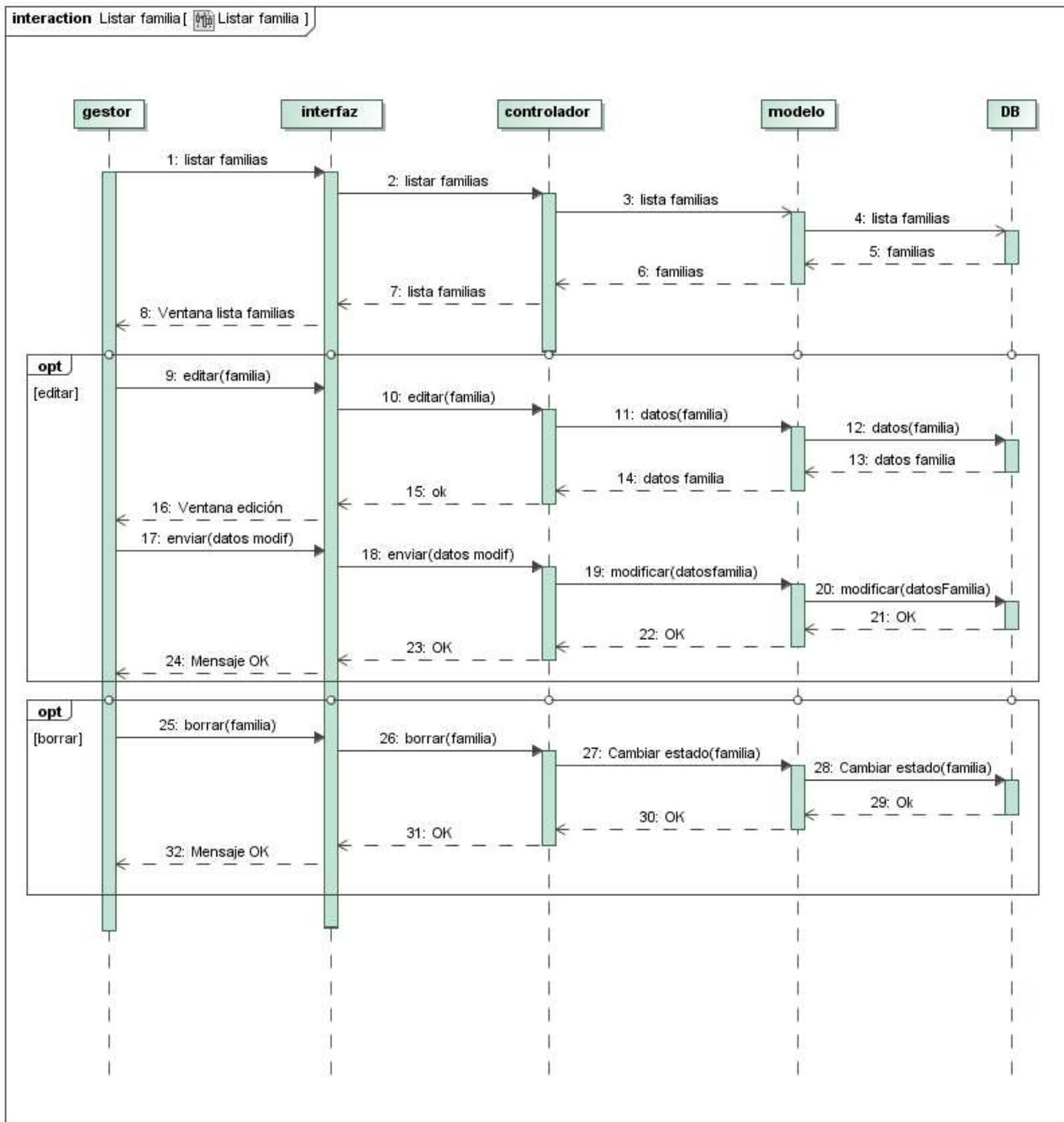


IMAGEN 23.

3.2.8. AGREGAR FAMILIA

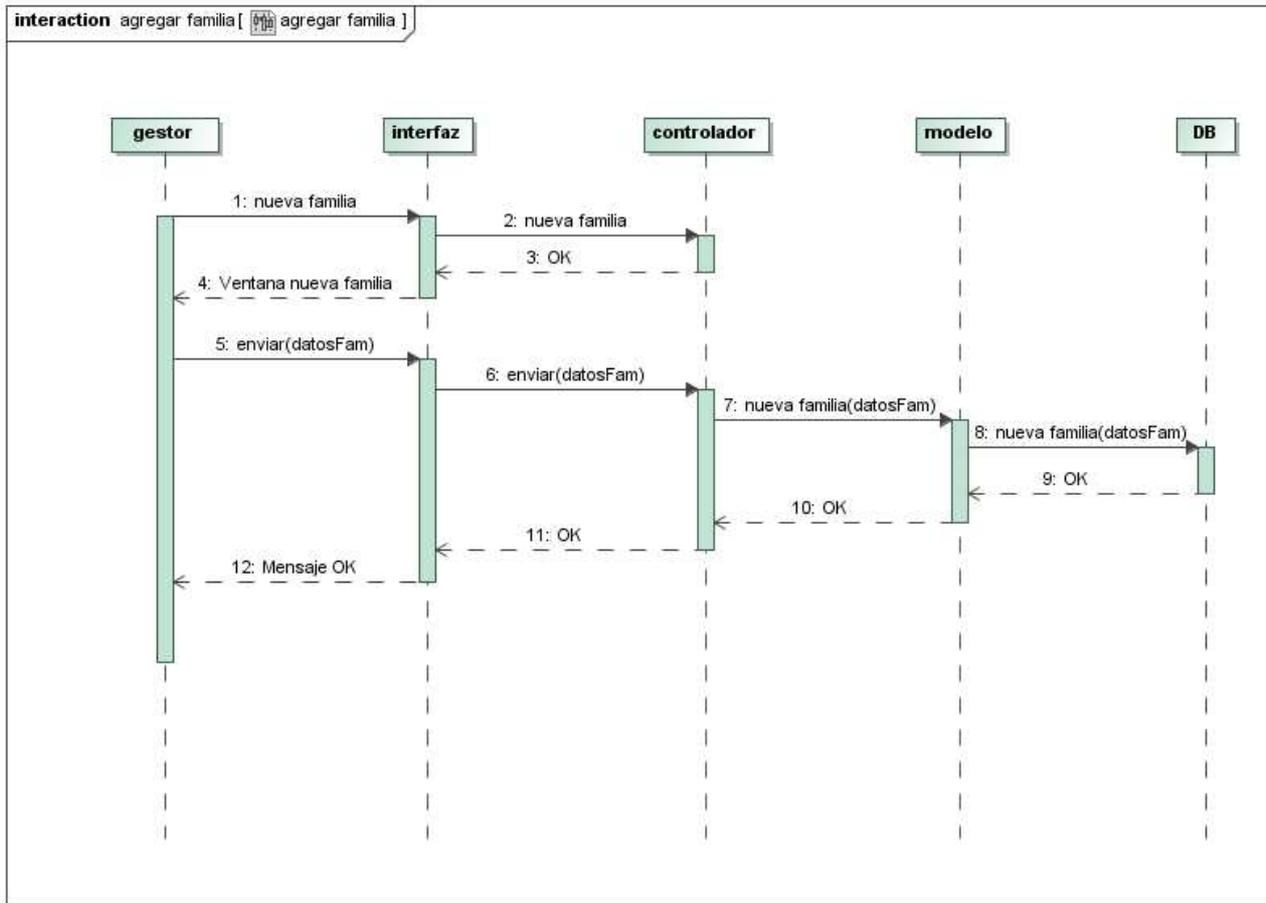


IMAGEN 24.

3.2.9. LISTAR/EDITAR/BORRAR CLASIFICACIÓN

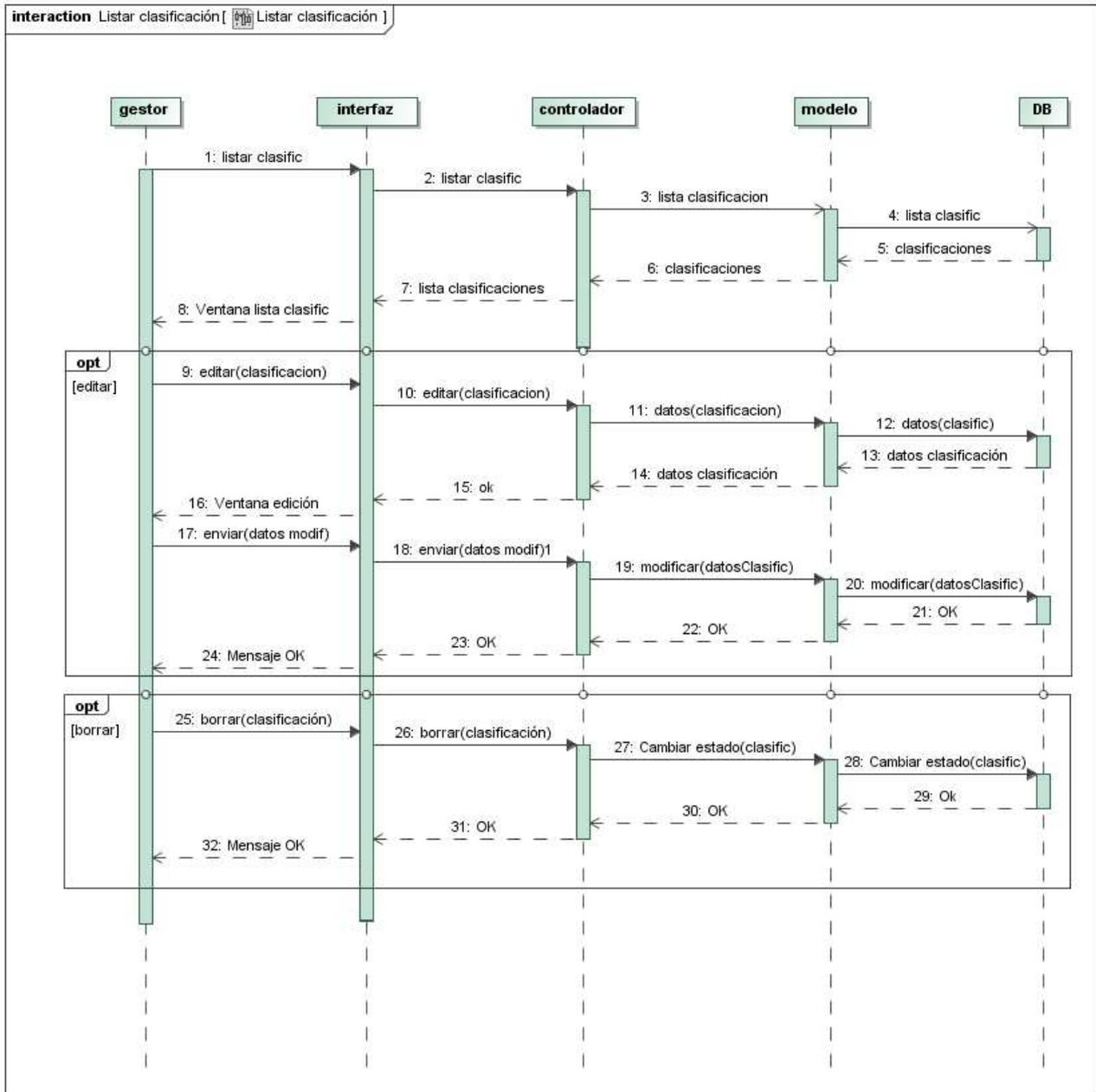


IMAGEN 25.

3.2.10. AGREGAR CLASIFICACIÓN

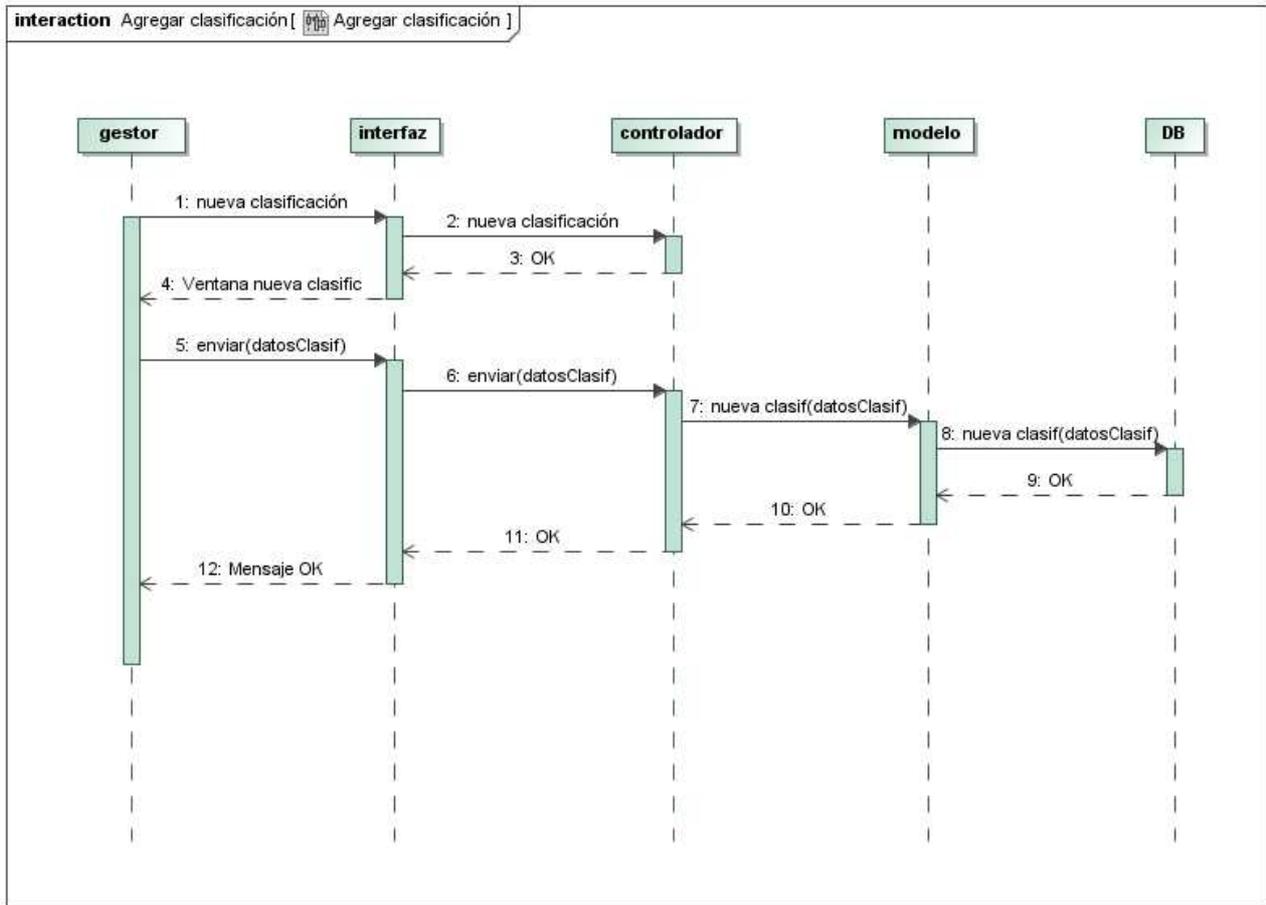


IMAGEN 26.

3.2.11. LISTAR PRODUCTOS (GESTOR)

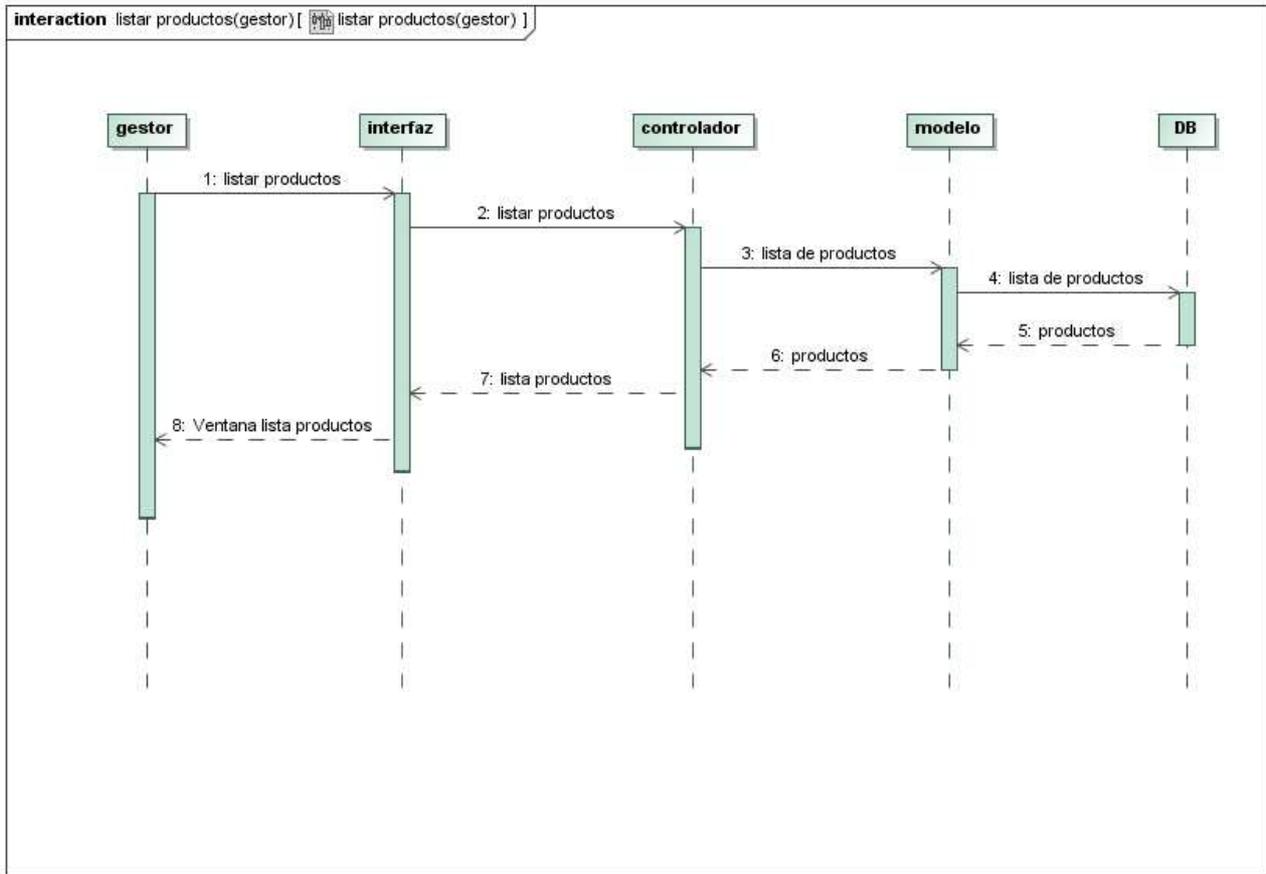


IMAGEN 27.

3.3. MODELO DE DATOS. DIAGRAMA ENTIDAD-RELACIÓN

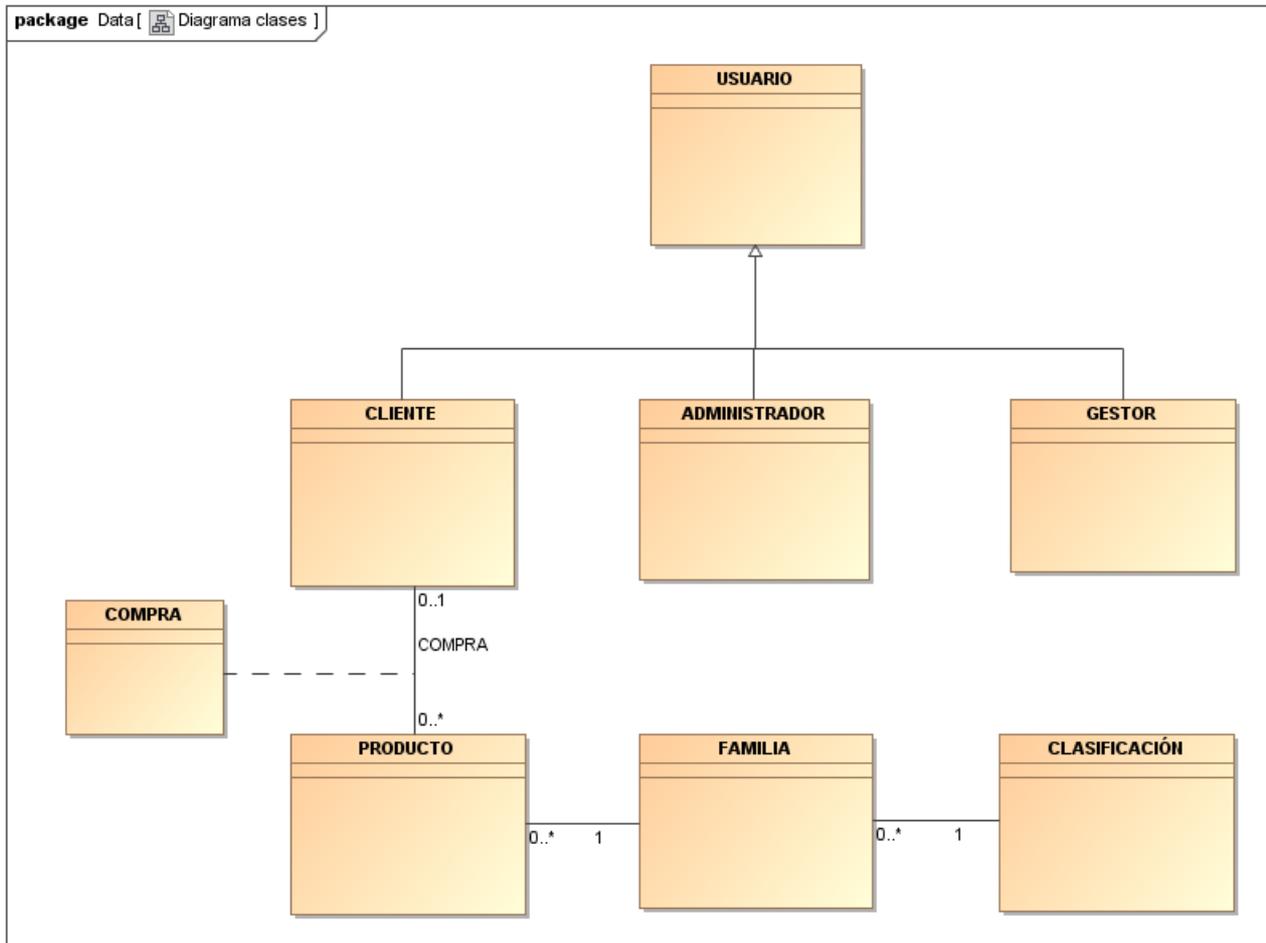


IMAGEN 28.

4. DISEÑO TÉCNICO

4.1. USO DE PATRONES

4.1.1. PATRÓN MVC MODELO VISTA CONTROLADOR

Para la realización del proyecto Mofm se ha utilizado uno de los patrones de diseño de arquitectura más utilizados en proyectos J2EE, el patrón MVC (Modelo Vista Controlador).

Este modelo separa los datos de la aplicación (modelo), la interfaz (vista) y el modelo de negocio (controlador).

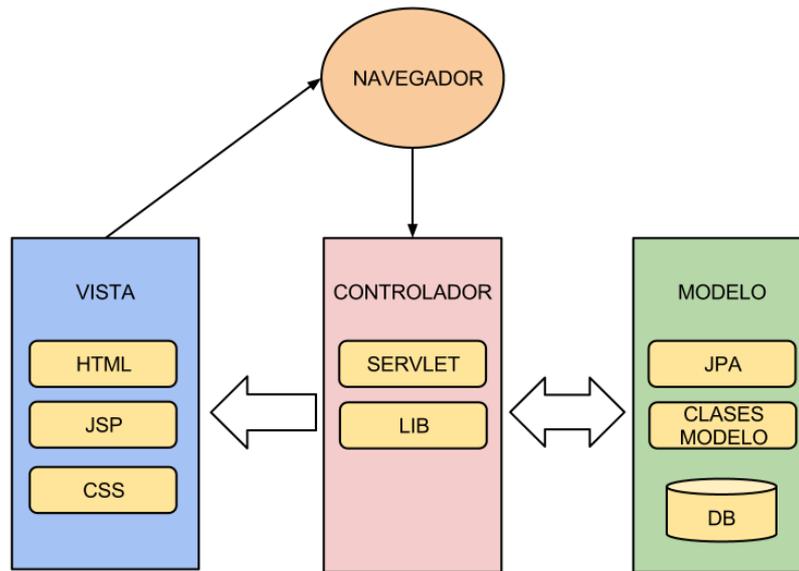


IMAGEN 29.

Las principales ventajas de este modelo son:

- Tener separado el acceso a los datos (modelo) nos permite tener una aplicación que de muy fácil manera se puede adaptar a otro SGBD sistema gestor de base de datos, puesto que sólo modificando una parte de la aplicación se puede lograr.
- Por otra parte, tener separada la interfaz (vista) también nos permite poder utilizar la aplicación en diferentes dispositivos modificando sólo esta parte. Por ejemplo nuestra aplicación está diseñada para utilizarse con navegadores web, pero se podría crear una nueva vista para utilizarse en navegadores para dispositivos móviles.

En resumen, escalabilidad, reutilización de componentes, simplicidad de mantenimiento, ...

4.1.2. PATRÓN SINGLETON

El patrón de diseño singleton (instancia única) está diseñada para restringir la creación de objetos pertenecientes a una clase o el valor de un tipo a un único objeto.

Aprovechamos el uso del patrón singleton para la creación del objeto Entity Manager Factory puesto que tiene un coste muy grande hacerlo una y otra vez en cada servlet. De esta manera nos aseguramos de que hay una instancia única.

```
public class Mofm {

    private static EntityManagerFactory emf;
    private Mofm(){

    }

    public static EntityManagerFactory getEMF(){
```

```
        if (emf==null){
            emf = Persistence.createEntityManagerFactory("mofmWebPU");
        }
        return emf;
    }
}
```

Por lo que nos garantizamos que la clase sólo tenga una instancia y proporcionamos un punto de acceso global a ella.

4.1.3. PATRÓN PERSISTENT CUSTOMER SESSION

También se ha optado por el uso de sesiones para almacenar parte de la información de un usuario que se conecta.

En este sentido se guarda dentro de la sesión el usuario conectado (si éste decide logearse) y la lista de productos adquiridos y que se van agregando a su cesta.

Ésto provoca que para visualizar la cesta no se necesita acceder a la base de datos en ningún momento. Simplemente accediendo a los atributos de la sesión se obtienen los productos comprados.

Se ha optado por el uso de este tipo de objetos a pesar de sus inconvenientes a la hora de pérdidas de sesiones durante la conexión, o por cualquier error del navegador.

Además, se ha utilizado el patrón singleton para que la sesión sea única durante toda la visita de un usuario a la web. Por lo que se crea un objeto `httpsession` sólo si no está éste creado.

4.1.4. PATRÓN I18N.

Para fomentar la escalabilidad en el uso de mensajes dentro del controlador, se ha optado por el uso de la Internationalization (I18N).

De manera que todos los mensajes de error están separados en un fichero "mofmvista.mensajes.properties". Por lo que el cambio de idioma o la modificación de cualquier texto de error es más sencillo de realizar.

De esta manera también se ha pretendido ser más fiel al Modelo MVC. Enmarcando los mensajes dentro de la vista.

4.2. DIAGRAMA DE PAQUETES

Se ha realizado una división de paquetes concorde a los criterios del modelo MVC:

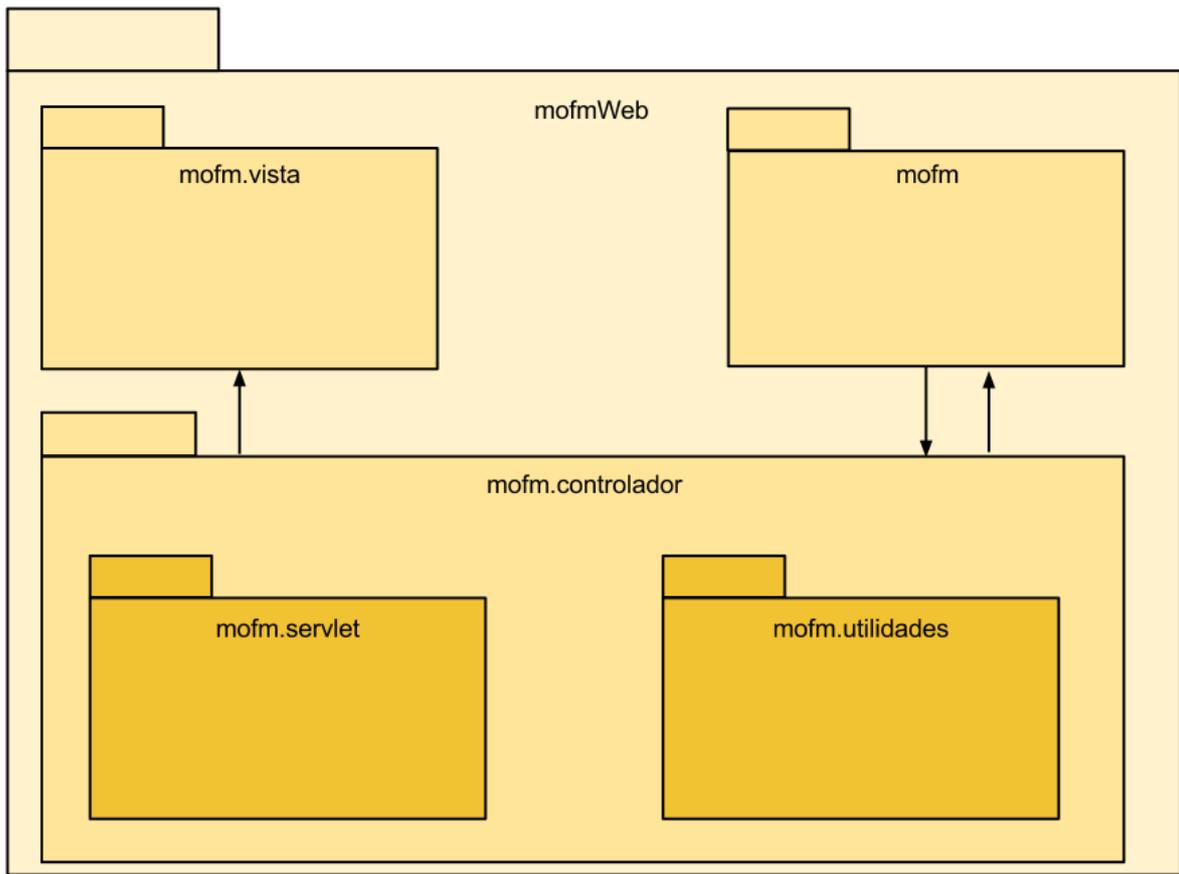


IMAGEN 30.

4.3. DIAGRAMA DE CLASES GESTORAS (CONTROLADORAS)

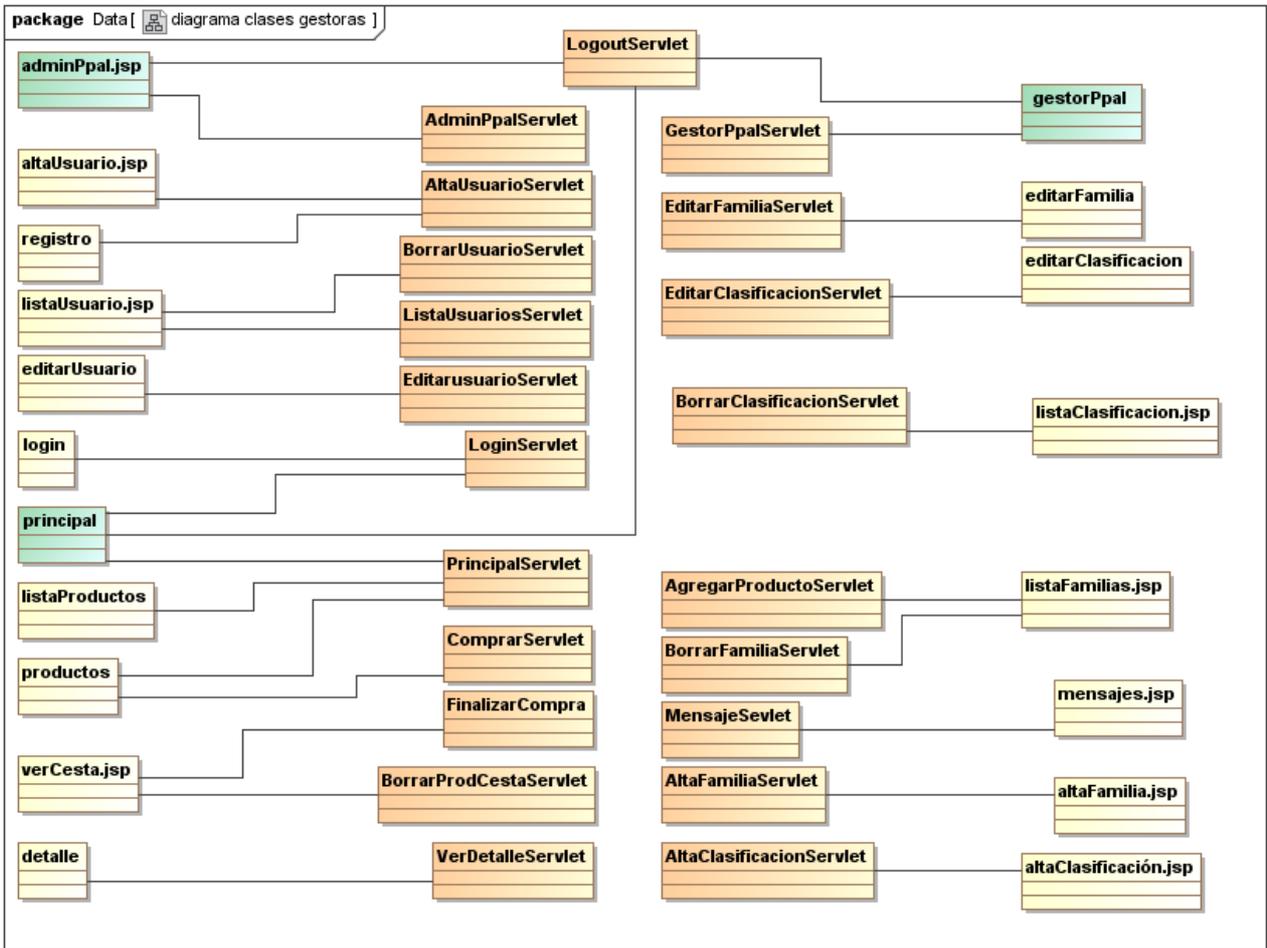


IMAGEN 31.

4.4. JERARQUÍA DE VENTANAS

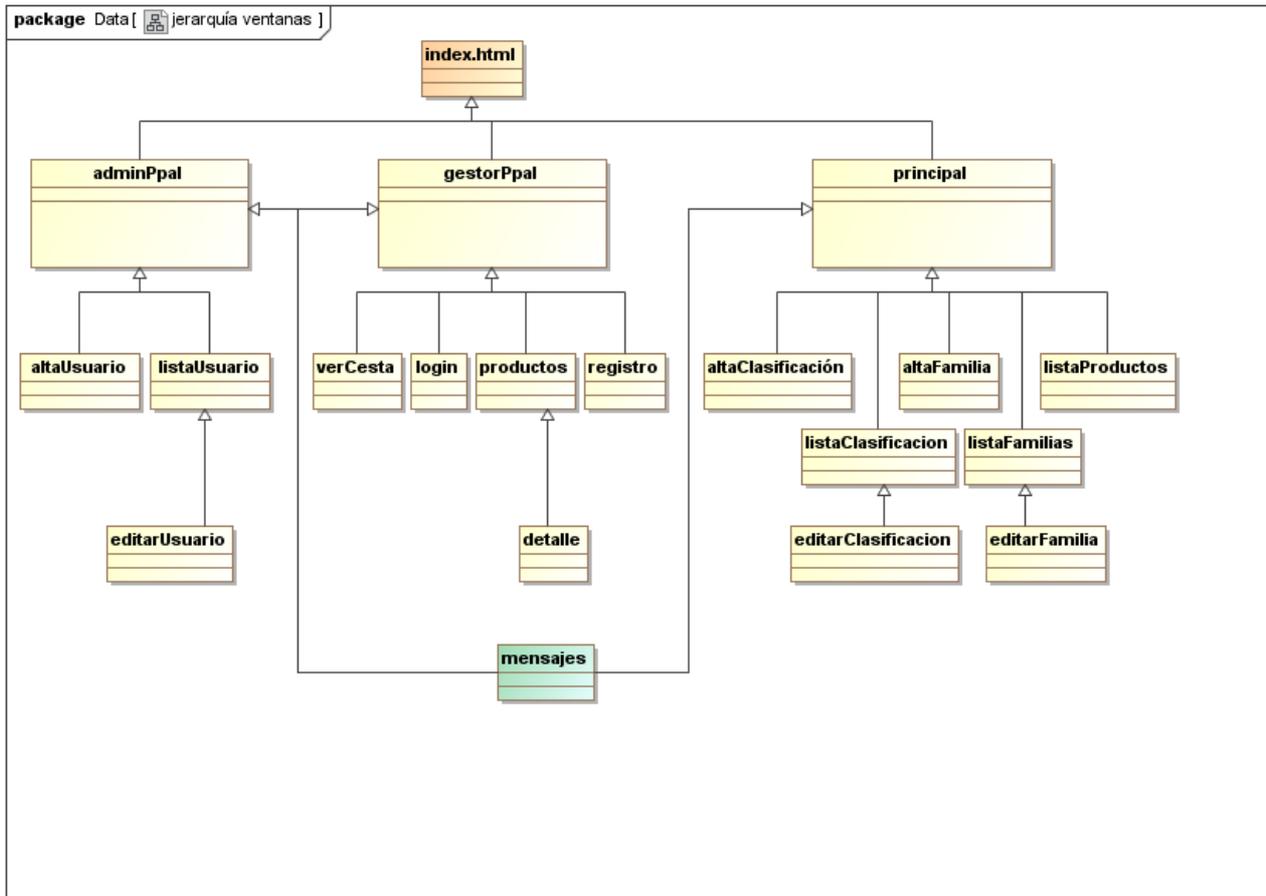


IMAGEN 32.

4.5. DISEÑO INTERFAZ

Cuando un usuario se conecte a través de su navegador a la página donde estará la aplicación podrá directamente consultar todos los productos a la venta a través del panel principal. Además en el menú de productos tendrá las diferentes familias de productos ordenadas por clasificaciones de manera que sea más fácil localizar el producto que se necesita.

En el menú “OPCIONES” en la parte izquierda de la pantalla tendrá todas las opciones disponibles para cada usuario.

En este caso al ser un usuario no registrado, sólo podrá acceder (login) al entorno con su clave o darse de alta. Aparte de seguir visualizando productos.



IMAGEN 33.

Pulsando sobre la opción Detalle de cada uno de los productos, un usuario sin registrar podrá ver una descripción detallada del producto así como una fotografía más grande del mismo.

Cuando un usuario accede por primera vez para realizar una compra deberá registrarse.

Para ello accederá a la opción registrarse que se proporciona dentro del menú opciones a los usuarios no registrados.

Una vez que haya accedido el sistema le mostrará la siguiente pantalla:



IMAGEN 34.

Una vez que el usuario está dado de alta, podrá acceder con sus credenciales a través de la siguiente ventana:



IMAGEN 35.

Una vez que el sistema compruebe sus credenciales accederá a la aplicación como usuario registrado de manera que ya podrá comprar productos a través de la opción correspondiente que existe debajo de la imagen de cada producto.

Cada vez que compra un producto, éste se añade a su cesta personal.

Un usuario registrado podrá ver su cesta personal a través de la opción correspondiente de su menú. Una vez que pulsa ver cesta la aplicación le mostrará la siguiente ventana:



IMAGEN 36.

Dentro de su cesta el usuario podrá finalizar la compra o quitar uno o varios de los productos que estén añadidos.

Un usuario registrado también podrá salir de su cuenta a través de la opción salir (logout).

Cuando el usuario que acceda al sistema tenga el rol "Administrador" sus opciones serán diferentes y podrá:

- Dar de alta un usuario
- Listar usuarios
- Cerrar sesión (logout)

Dispondrá de todas estas opciones en el menú opciones dentro de su sesión.

La ventana que se mostrará será como esta:



IMAGEN 37.

Cuando el usuario administrador pulse sobre “alta usuario” se abrirá la siguiente ventana donde tendrá que rellenar los datos necesarios para realizar el alta satisfactoriamente:



IMAGEN 38.

Además un usuario administrador puede listar todos los usuarios registrados en el sistema

de manera que se le mostrará la siguiente ventana:



IMAGEN 39.

Para cada usuario el Administrador podrá editar su información (pulsando sobre el icono del lápiz) o eliminarlo (pulsando sobre el icono del aspa).

Cuando se edita la información de un usuario se muestra la siguiente ventana:



IMAGEN 40.

Por otra parte, un usuario registrado también puede tener el rol “Gestor”. Cuando esto ocurra tendrá las siguientes opciones:

- Agregar familia de productos nueva.
- Listar familia de productos.
- Salir de la sesión (logout).

Cuando pulse agregar familia le aparecerá la siguiente ventana:



The image shows a web application interface for 'MADE OF FELT MARKET'. On the left is a purple sidebar with a circular logo containing 'Mo FM' and a menu with the following items: SALIR(LOGOUT), LISTAR FAMILIAS, ALTA FAMILIA, LISTAR CLASIF., ALTA CLASIF., and LISTAR PRODUCTOS. The main content area has a purple header with the text 'MADE OF FELT MARKET'. Below the header is a form titled 'ALTA FAMILIA' with the following fields: 'Nombre:' (text input), 'Descripción:' (text input), 'URL Imagen Pequeña:' (text input), 'URL Imagen:' (text input), 'Precio:' (text input), and 'Clasificación:' (dropdown menu with 'símbolos' selected). An 'Enviar' button is located below the 'Clasificación' field.

IMAGEN 41.

Una vez rellene los datos se dará el alta de la familia de productos.

Cuando utilice la opción listar familias parecerá la siguiente ventana:



IMAGEN 42.

Dentro del listado de familias el Gestor podrá:

- Editar la información de una familia (Icono lápiz).
- Borrar una familia (Icono de aspa).
- Añadir productos a una familia (pulsando sobre el icono correspondiente).

Cuando se edita la información se muestra la siguiente ventana:

MADE OF FELT MARKET

EDICIÓN DE FAMILIA

Nombre:

Descripción:

URL Imagen Pequeña:

URL Imagen:

Precio:

Clasificación:

MENÚ

- SALIR(LOGOUT)
- LISTAR FAMILIAS
- ALTA FAMILIA
- LISTAR CLASIF.
- ALTA CLASIF.
- LISTAR PRODUCTOS

IMAGEN 43.

Desde la cual se podrá modificar cualquier dato de la familia seleccionada.

Análogamente un Gestor podrá listar, editar, agregar y borrar nuevas clasificaciones. Y obtener un listado de todos los productos. A estas funcionalidades se accede a través de la opción correspondiente del menú a la izquierda de la pantalla.

5. IMPLEMENTACIÓN

5.1. HERRAMIENTAS DE DESARROLLO

5.1.1. NETBEANS

Para la realización de este proyecto se ha utilizado el IDE Entorno de desarrollo integrado NetBeans 7.0.1.

A través de esta herramienta se han implementado todas las clases, servlets, jsp's y hojas de estilos css. Para ello se han necesitado las siguientes librerías:

- Persistence: Como ya se ha comentado para realizar el proyecto se ha decidido utilizar JPA Java Persistence API como método de acceso a la base de datos. De manera que se facilite el mapeo de atributos entre la Base de datos y el modelo de objetos diseñado en la aplicación.
- Hibernate: Como implementación de JPA se ha decidido utilizar el componente Hibernate.

- MySQL JDBC Driver: Para la comunicación con el Sistema Gestor de Bases de Datos se necesitará el driver para JDBC de MySQL, ya que éste es el que se ha utilizado en la realización del proyecto.
- JSTL 1.1.: Por último, también se ha decidido utilizar una librería de tags para jsp's para facilitar el acceso a la información desde los mismos. En este caso se ha optado por JSTL.

5.2. MANUAL DE INSTALACIÓN

El contenido del producto (plealm_Producto.zip) es el siguiente:

- Build: Con todas las clases compiladas del proyecto.
- Dist: Con el archivo .war para la instalación en el servidor web.
- Nbproject: Con archivos del IDE utilizado Netbeans.
- Src: Con los archivos .java de todas las clases del proyecto.
- Web: Con los archivos jsp, imágenes, y hoja de estilos de la vista del proyecto.
- CreaciónBD DDL.sql: Archivo de creación de la base de datos.
- CreaciónBD DML.sql: Archivo de inserción de un juego de datos de prueba para testear el proyecto.
- Librerías.zip: Archivo con las librerías necesarias para el funcionamiento del proyecto (Persistence, Hibernate, driver jdbc de MySQL y jstl 1.1)

Para la instalación de la aplicación se seguirá el siguiente procedimiento:

NOTA: Las siguientes instrucciones corresponden a una instalación bajo entorno Windows aunque se podrá realizar en cualquier plataforma de una forma análoga.

a. Instalación del SGBD MySQL

Para ellos nos descargamos el instalador desde la página de MySQL seleccionando la plataforma y el sistema operativo concreto que se va a utilizar.

<http://www.mysql.com/downloads/mysql/>

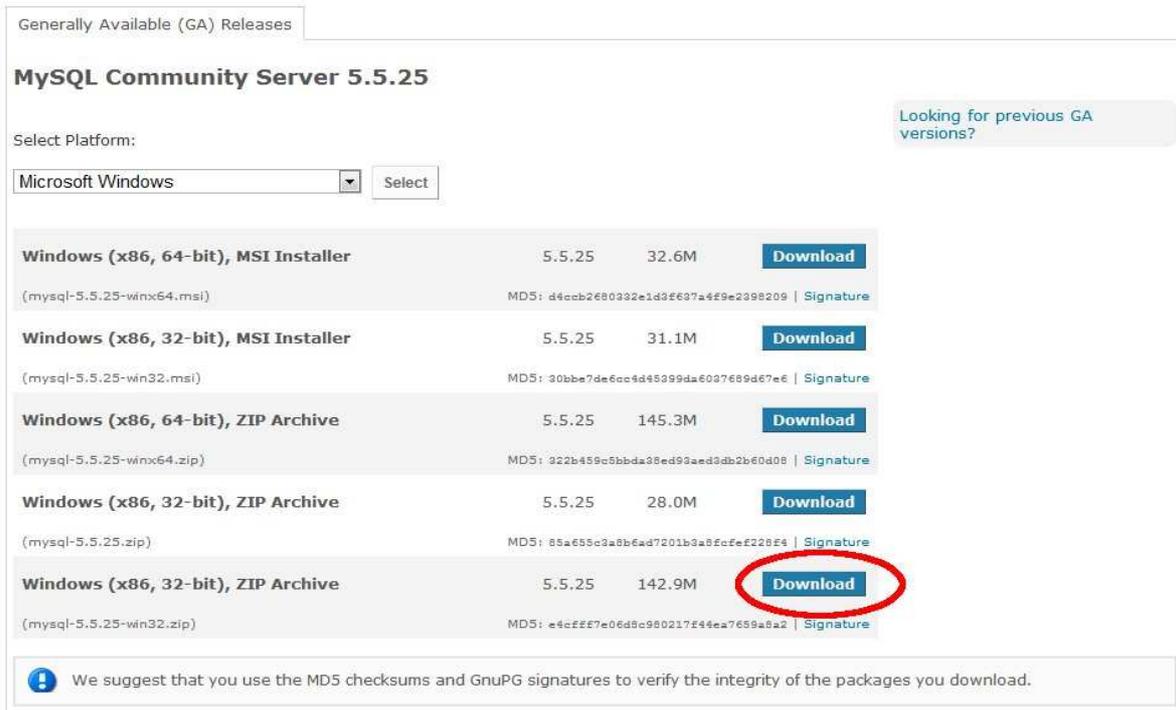


IMAGEN 44.

b. Creación de la base de datos con MySQLWorkbench

Una vez que se ha instalado se ejecuta MySQL Workbench se abre una conexión a través de “Open connection to Start Querying” / “SQL Editor (Local Instance)”.

Después se abre una nueva Query y se ejecuta las siguientes instrucciones que se pueden encontrar en el archivo “**CeaciónBD DDL.sql**”:



IMAGEN 45.

De esta manera ya tendremos creada la base de datos.

También se adunta en el archivo “CreaciónBD DML.sql” con datos de prueba para poder probar la aplicación.

c. Instalación de JDK 1.7

Podemos descargar JDK 1.7 desde la página de Java: <http://oracle.com>

Una vez que lo hayamos hecho sólo tendremos que ejecutar el archivo descargado siguiendo las instrucciones para su instalación.

d. Instalación de Apache Tomcat

- Se descarga Apache Tomcat desde la página oficial: <http://tomcat.apache.org/>
- Se descomprime en la ubicación elegida.
- Se realiza la configuración deseada según las necesidades de cada uno a través

de los archivos /conf. Existen numerosos manuales al respecto en la red.

- Se inicia el servicio a través del archivo Startup.bat que se encuentra situado en la carpeta /bin

e. Instalar archivo Mofmweb.war en Apache Tomcat

Para instalar el archivo MofmWeb.war sólo es necesario copiarlo en el directorio donde se instaló Apache Tomcat en la carpeta “/webapps”.

f. Obtención e instalación de las librerías

Se añadirán las siguiente librerías a la carpeta “\webapps\mofmWeb\WEB-INF\lib” de Apache Tomcat:

- Persistence
- Hibernate
- driver JDBC MySQL
- JSTL

Se puede descargar el paquete completo de librerías de la aplicación en:

<https://docs.google.com/open?id=0B3Clxb3QehHGNzdoTE4yU3lxdU0>

Aunque se adjunta un archivo zip con las mismas en el producto: “librerías.zip”

g. Uso del navegador para el acceso a la aplicación

Accediendo desde cualquier navegador a la URL donde está el servidor web que hemos instalado accederemos a la aplicación.

En nuestro caso accederemos localmente desde <http://localhost:8080/mofm/>

6. CONCLUSIONES

En la elección de la realización de este proyecto en J2EE siempre ha estado un objetivo preferente; el aprendizaje y/o el perfeccionamiento de una gran gama de herramientas, librerías y opciones que permiten implementar una cantidad de funcionalidades muy alta y muy plena siempre enmarcadas en la universalidad y estandarización que el entorno web proporciona.

El intenso uso que ya se hace de los entornos web para la realización de aplicaciones unidos a la cada vez mayor demanda de acceso remoto para aplicaciones que auguran un futuro muy prometedor para esta tecnología han sido claves en la elección del tipo de proyecto y de la tecnología de desarrollo.

En este sentido, creo que se ha podido satisfacer este objetivo primario de aprendizaje durante las diferentes fases: Especificación, Análisis, diseño, implementación, ... que han formado el ciclo de vida del proyecto MOFM.

Además, el producto obtenido: tanto aplicación, memoria y presentación, ha alcanzado e incluso superado el nivel esperado en los inicios del proyecto. Tanto por funcionalidades, como por diseño, integración de las partes, modularidad, reutilización, claridad, ...

Por lo que sólo puedo decir que la realización de el proyecto MOFM ha sido una grata

experiencia personal que ha conseguido aportarme una mayoría de elementos positivos.

7. BIBLIOGRAFÍA

- Enciclopedia de contenido libre. “**Wikipedia**”.
<http://es.wikipedia.org/wiki/Wikipedia:Portada>
- Benet Campderrich Falgueras, Recerca Informàtica, S. L. (2004) ”**Ingeniería del Software**”. Fundació per a la Universitat Oberta de Catalunya.
- API specification for version 6 of the Java™ Platform, Standard Edition. “**Java 6 API Specification**”.
<http://docs.oracle.com/javase/1.2.1/api/>
- Joaquín Gracia (2005). Diseño de software orientado a objetos. “**Patrones de diseño**”.
<http://www.ingenierosoftware.com/analisisydiseno/patrones-diseno.php>
- Raúl Gómez Martil (2012). Trabajo final de carrera UOC. “**Inter Golf**”.
<http://openaccess.uoc.edu/webapps/o2/handle/10609/36>
- Ivan Folgueira Bande (2012). Trabajo final de carrera UOC. “**UOC Book. Xarxa social fent ús de J2EE** ”.
<http://openaccess.uoc.edu/webapps/o2/handle/10609/36>
- David Geary (2003). Java World. “**Simply Singleton**”.
<http://www.javaworld.com/javaworld/jw-04-2003/jw-0425-designpatterns.html>
- David Marco (2010). Tutoriales página personal David Marco. “**Introducción al JPA**”:
<http://www.davidmarco.es/blog/entrada.php?id=144>
- Ubuntu life (2008). Ubuntu life #WorldRevolution. “**Introducción al JPA. Java Persistence API**”.
<http://ubuntulife.wordpress.com/2008/10/20/introduccion-a-jpa-java-persistence-api/>
- Desarrollo web (2012). Desarrollo web Manuales. “**Manual de CSS. Hojas de estilos**”:
<http://www.desarrolloweb.com/manuales/manual-css-hojas-de-estilo.html>
- Springhibernate.net (2009). Hibernatutorial. “**Hibernate and Spring Tutorial**”.
<http://www.hibernate-tutorial.com/index.html>
- Universidade da Coruña. “**Minitutorial de MySQL Workbench**”.
<http://coba.dc.fi.udc.es/~bd/bd2/MySQLLWB/tutorialWB.html>