



**Trabajo final de carrera**

**INGENIERIA TÉCNICA EN  
INFORMÁTICA DE SISTEMAS**

**Facultad de Matemáticas**

**Universidad de Barcelona**

---

**PROYECTO ECOMMERCE CON  
SINCRONIZACIÓN CON ERP**

---

**Francesc Guardiola Gonzalo**

Director: Pablo Almajano

Realizado a: Departament de Matemàtica  
Aplicada i Anàlisi. UB

Barcelona, 4 de octubre de 2013

## RESUMEN

El mundo de internet está marcando diferencias en un mercado global y mundial. Toda tienda física necesita estar en internet para poder estar a la altura de la competencia. Dicho paso es caro y prolongado, ya que la creación de un eCommerce (tienda en internet) y disponer del personal suficiente para desarrollar el proyecto, es demasiado costoso.

No obstante, existen herramientas de código abierto (*Open Source*) que ayudan a la creación de estas. Estos sistemas ya disponen de toda la estructura mínima necesaria para crear una tienda en internet, además son estructuras básicas que se pueden configurar y modular para poder adaptar las diferentes funcionalidades del eCommerce a tu negocio o necesidad.

Por otra parte, un sistema ERP (*Enterprise Resource Planning*) se encarga de la gestión de productos y stocks de los almacenes que proveen al eCommerce.

Proponemos el proyecto Mountlegend (<http://www.mountlegend.com>), actualmente online y en funcionamiento. Mountlegend gestiona el catalogo y stocks de la tienda eCommerce y lo sincroniza con un ERP. A la vez el ERP se encarga de gestionar los mismos productos en tiendas físicas de Andorra (Yeti Sports).

El proyecto consiste en la implementación de un eCommerce usando tecnologías existentes y con una comunicación de sincronización con el ERP, todo ello adaptado a las necesidades del cliente.

# Índice de contenido

<b>1. Introducción</b>	<b>4</b>
1.1. Ámbito del proyecto	5
1.2. Motivación	5
1.3. Objetivos generales	6
1.4. Objetivos específicos	6
1.5. Organización de la memoria	8
<b>2. Antecedentes</b>	<b>9</b>
2.1. Definición de ERP	9
2.2. Definición de eCommerce	12
<b>3. Análisis</b>	<b>16</b>
3.1. Diagrama de casos de uso	16
3.2. Casos de uso	16
3.3. Definición del modelo de dominio	21
3.4. Requerimientos generales: software y hardware	22
<b>4. Diseño</b>	<b>24</b>
4.1. Diagrama de secuencia	24
4.2. Diagrama de clases	26
<b>5. Implementación</b>	<b>27</b>
5.1. Instalación eCommerce	28
5.2. Módulo de importación	31
5.2.1. Preparación de datos	32
5.2.2. Sincronización eCommerce	33
5.3. Comprar y actualización de stocks	36
5.4. Diagrama de componentes	39
<b>6. Mejoras propuestas</b>	<b>40</b>
6.1. Diseño	41
6.1.1. Diagrama de secuencia	41
6.1.2. Diagrama de clases	42
<b>7. Conclusión</b>	<b>43</b>
<b>8. Referencias bibliográficas</b>	<b>44</b>

# 1. Introducción

Hoy en día tenemos un mercado nuevo y emergente en Internet. Gracias al canal de internet las empresas pueden abrir nuevos horizontes y pueden vender su producto por todo el mundo. Gran parte de tiendas que tiene un punto de venta físico puede optar a un punto de venta online. Con este paso el vendedor crea una nueva tienda en Internet donde muestra todo su producto. Este producto lo ofrece a un público global (internet), lo cual supone optar a distribuir su producto en su país (nivel local). Además, también opta a la posibilidad de vender su producto a los países que más le convenga (nivel internacional), ya que en internet, con la misma plataforma tiene total libertad para elegir dónde quiere vender su producto. Un eCommerce[1] es un sistema que nos permite crear una tienda para vender producto en Internet mediante una página web.

Para crear este portal online disponemos de diferentes plataformas prefabricadas, que nos permiten crear una web totalmente equipada y configurable para crear un eCommerce a medida de nuestras necesidades. Las plataformas actuales más utilizadas y más sofisticadas del mercado, son Magento y Prestashop.

Por otro lado, algunas tiendas suelen trabajar con un programa denominado ERP (*Enterprise Resource Planning*)[2], que es el encargado de gestionar todo el producto de la tienda. Con este gestionamos las ventas, precios, stocks, características y todos los detalles que el ERP incorpore en el sistema.

Básicamente vemos que se generan sinergias entre ambos sistemas. El eCommerce se encarga de gestionar las ventas de producto mediante una página web, y un ERP es capaz de gestionar las ventas de producto mediante tienda física. En este punto vemos que se puede crear una comunicación directa para unir ambas gestiones en un solo sistema.

Este proyecto lleva a cabo el montaje de un eCommerce (Magento) que se sincroniza con un ERP. Como resultado, optimiza la introducción de datos e información de los productos y la gestión de la venta de este en un solo canal.

## **1.1. Ámbitos del proyecto**

Para poder llevar a cabo el proyecto, ha sido necesario un cumulo de conocimientos, la implementación del eCommerce mediante un sistema prefabricado y configurable, su modificación y la sincronización con el ERP. En cada apartado he hecho uso del conocimiento adquirido durante la carrera.

Inicialmente para la instalación de la plataforma eCommerce usé los conocimientos adquiridos en Administración de Sistemas, Redes y comunicaciones y Bases de datos. Esta última me sirvió para montar los archivos en el servidor, crear la base de datos e realizar la instalación en el servidor remoto (computadora que se accede de forma remota y está conectada a internet). En segundo lugar, para poder modificar el sistema, use conocimientos de Sistemas operativos, programación y programación para la red, ya que la estructuración de la programación, el funcionamiento y la forma de tratar y entender los datos del eCommerce, se pueden extrapolar a prácticas y conocimientos realizados en dichas asignaturas. Finalmente para la sincronización entre ambos sistemas, usé conocimientos de las asignaturas de lógica, bases de datos y programación para la red, donde estructuré de forma lógica y eficiente, los datos a transferir entre ambos sistemas.

En la asignatura de programación para la red, es donde aprendí a realizar diseños en estándar UML, desarrollar proyectos de software y realizar una memoria de cada uno.

## **1.2. Motivación**

En este proyecto soluciono varias finalidades. Primeramente uno de los motivos para realizar el proyecto es construir un eCommerce para poder vender producto por Internet.

Por otro lado el motivo principal es la sincronización entre el eCommerce y el ERP, que nos permite optimizar el proceso de comunicación entre ambos sistemas, y de esta forma optimiza el funcionamiento de todo el sistema. El administrador del eCommerce y del ERP, optimiza e economiza su trabajo ya que solo debe introducir datos en el ERP y estos se sincronizan de forma automática con el eCommerce.

### **1.3. Objetivos generales**

El proyecto consiste en la construcción de un eCommerce, cuyo nombre es Mountlegend[3] (actualmente online y en funcionamiento), que debe ser capaz de sincronizarse con un ERP llamado PowerStock, que actualmente se sincroniza con tres almacenes externos.

He de conseguir que todo el producto que existe dentro del ERP, se actualice y exista dentro del eCommerce (Mountlegend) de forma automática.

Inicialmente tenemos varias premisas, el ERP nos pasará los datos mediante un archivo, el proceso debe tener diferentes funcionalidades, debe ser rápido e eficiente y se debe informar al administrador de todos los pasos y sucesos.

### **1.4. Objetivos específicos**

Debo construir un eCommerce que reúna un seguido de características. Primeramente, debe ser un proyecto vistoso y que atraiga al público solo con la imagen. En segundo lugar, debo configurar un catálogo que estará compuesto de categorías y atributos que ayudaran a realizar filtros para la búsqueda de producto. A parte, ha de mostrar un filtro especial de marcas y un campo de búsqueda para que el cliente pueda realizar búsquedas de producto directas. Todo esto afecta a la configuración de catálogo, en el ámbito de características de funcionamiento, debe tener dos idiomas, español e inglés. Debe poder mostrar precios en cuatro monedas diferentes (euro, dólar, libras y franco suizo). Debe tener dos métodos de pago, tarjeta de crédito y Paypal[8]. Finalmente para realizar envíos debe tener posibilidad de realizar envío a península e internacional.

Por otro lado, la integración del eCommerce con el ERP permite que solo exista un punto de entrada de datos, ya que los datos que existan en el ERP se deberán sincronizaran en el eCommerce. Este proceso se debe realizar para evitar el uso del administrador del eCommerce. Este acostumbra a ser lento y dificultoso, ya que se deben configurar muchos parámetros para mostrar la información correcta en el catálogo que visita el cliente. Además los

datos que el administrador ha de introducir, son los que ya existen en el ERP y nos ahorra volverlos a introducir en el administrador del eCommerce. Finalmente una razón muy importante para estos objetivos es la característica de entrar datos en el ERP, se realiza de forma local, ya que el ERP está instalado en la computadora en la que estamos trabajando, por el contrario, si el administrador introduce datos mediante el administrador del eCommerce, debe realizarlo en un navegador con Internet, hecho que obliga a tener conexión a Internet.

## 1.5. Organización de la memoria

La memoria se organiza con los siguientes apartados:

- **Antecedentes:** En esta sección se explica y se pone en contexto las dos partes más importantes del proyecto, el ERP y el eCommerce. Una vez explicados los dos pilares, se encuentran las tecnologías usadas para desarrollar el proyecto.
- **Análisis:** Se realiza el análisis de la estructura para desarrollar los problemas y necesidades del proyecto. Se usa la notación estándar para el modelado UML. Se detallan los casos de uso, el modelo de dominio y los requerimientos de software y hardware de la aplicación.
- **Diseño:** Apartado donde se desarrolla de forma conceptual los requisitos que requiere el proyecto. Aquí se detallan los diagramas de clases y diagramas de interacción en estándar UML.
- **Implementación:** Resultados y proceso de todo el desarrollo realizado para llevar a cabo el desarrollo conceptual planteado en el apartado de diseño. Se da el resultado de la implementación de una forma descriptiva y gráfica.
- **Mejoras propuestas:** Diseño y estructura de propuestas de mejoras, donde se amplía el proyecto realizado con nuevas estructuras y funciones. Amplia la compatibilidad con nueva plataforma eCommerce.
- **Conclusión:** Resumen final de todo el desarrollo, con los pros y contras que se encuentran a lo largo del proyecto. También se exponen las posibles mejoras que se podrían llevar a cabo para mejorar el sistema planteado.
- **Referencias bibliográficas:** Listado de referencias que aparecen durante la memoria del proyecto. Y que se deben consultar para profundizar en los conceptos que se usan en el proyecto.

## **2. Antecedentes**

### **2.1. Definición de ERP**

Un ERP (*Enterprise Resource Planning*) es un sistemas de planificación de recursos para empresas. Ofrece servicios de gestión, organización, estructuración y comunicación para el funcionamiento de una empresa o negocio.

El ERP es el encargado de la gestión del almacén, gestión de compras, contabilidad y organización de todos estos. Para poder realizar todo este manejo de datos, los ERP se caracterizan por tener una base de datos centralizada, lo cual facilita el uso de muchos datos en un mismo sitio. Al ser un sistema tan relevante para una empresa, normalmente los pequeños negocios o empresas que nunca han trabajado con un ERP deben modificar sus procesos para poder alinearlos con los del sistema ERP.

En este proyecto se ha hecho uso del ERP PowerStock. Este ha sido desarrollado a medida para realizar las funciones que necesitaban los clientes del propietario y desarrollador del sistema (Olivier Caux). El hecho de haberse desarrollado a medida y mejorado con el paso de los años, hace que sea un ERP estable, potente y con mejoras y funciones que facilitan el trabajo del usuario al usar el ERP.

PowerStock dispone de gestión de almacén. Se organiza con artículos que se componen de atributos como la referencia, código interno y EAN<sup>1</sup>. También tiene la posibilidad de trabajar con tallas y colores. Esto significa que con la creación de un artículo se generan los subartículos por talla y color. Soporta la introducción de datos descriptivos en 4 idiomas diferentes. Tiene soporte para la generación de etiquetas de códigos de barras en cualquier dispositivo, desde impresora Windows hasta impresora industrial, desde Artículos o cualquier documento de movimientos del sistema. Para poder realizar la gestión de cada artículo se realiza mediante una única pantalla de visualización de stocks y si tiene tallas y/o colores se puede ver también el stock de cada uno de ellos en esta misma pantalla. Finalmente se obtiene información histórica de movimientos de cualquier artículo.

Por otro lado, PowerStock dispone de gestión de compras, con lo cual se puede realizar la gestión de recepción y envío de pedidos de proveedores, realiza los descuentos y sumas de stock que comportan los pedidos a proveedores y la venta a clientes. Permite realizar altas de artículos desde la misma recepción de material. Por otro lado, también realiza la gestión de ventas, donde incluye la gestión de clientes, almacena sus datos, genera pedidos, facturas y tickets de venta, relacionados con el cliente que realiza la compra.

Finalmente un ERP completo integraría la parte de contabilidad, pero en este caso el desarrollador ha preferido crear un enlace de sincronización con el ERP SAGE[5] ContaPlus. Este es un ERP de gran envergadura que está muy insertado en el mercado actual. Este permite llevar la contabilidad de grandes, medianas y pequeñas empresas con cierta facilidad, por este motivo es un programa muy usado en el mercado.

Con todas estas características este ERP se adapta a la perfección en casos como el que tratamos en este proyecto, en el que gestiona tiendas físicas y van perfectamente relacionadas con la tienda eCommerce.

Pero en el sector de ERP, hay varias posibilidades que también integran la opción de integrarse con una plataforma eCommerce. Como he comentado antes uno de los sistemas ERP muy usado en el mercado es SAGE, el cual dispone de varios módulos de programa con el que te permite controlar cada departamento de tu negocio. Esta opción es muy empleada en muchos negocios, el problema que presenta es que al ser tan grande, no se adapta a los negocios, sino que los negocios se deben adaptar al sistema. Por otro lado, al tener tantas funcionalidades se convierte en un sistema caro, aun así el precio depende de las funcionalidades que se quieran usar.

Otro sistema muy empleado y que se proclama hoy en día como el sistema ERP más potente del mercado es SAP[6]. Sistema increíblemente potente y grande, que permite la gestión de corporaciones internacionales. Este sistema es muy caro y su uso es recomendado y habitual en grandes empresas con varias sedes. Finalmente un sistema muy usado en la conexión de plataforma eCommerce y ERP es el OpenERP[7], se trata de un ERP Open

Source, con lo que es gratuito su adquisición, y permite la gestión de un negocio. El inconveniente de este es su instalación y uso, que es complejo y necesita un conocimiento mínimo para configurarlo y adaptarlo a un negocio. Este último proceso hace que sea necesaria la contratación de un técnico o profesional que haya tenido experiencia en OpenERP y normalmente acostumbran a ser servicios de programación poco económicos.

El mercado no es pequeño en estos sistemas, pero para el proyecto de Mountlegend que necesita un ERP que se adapte un poco a sus necesidades, es preferible y más económico un ERP del estilo de PowerStock, que un ERP grande y complejo como SAGE, OpenERP o SAP. Este es uno de los motivos por los cuales se ha realizado este proyecto.

## 2.2. Definición de eCommerce

El comercio electrónico, eCommerce (*Electronic Commerce*) es el proceso de comercio mediante internet. Para realizar dicho proceso existen plataformas para construir sistemas que permiten crear una tienda donde montar tu propio catálogo de artículos. Cada artículo puede formar parte de una o más categorías, con las que podemos organizar todo el catálogo de nuestra tienda. Por otro lado, otra forma de organizar una tienda online es mediante los atributos. Estos permiten dotar a los productos de más información y características y a la vez permiten crear filtros para que los compradores puedan encontrar lo que buscan con la máxima facilidad posible, este proceso es de los más importante en un eCommerce, ya que en una tienda en internet es básico que el tiempo desde que un comprador entra en la página hasta que encuentra lo que busca sea el menor tiempo posible.

Un eCommerce no solo contiene un catálogo sino que este, también permite integrar métodos de pago, donde los más comunes son el pago por tarjeta de crédito, contra reembolso, PayPal (banco por internet) y transferencia bancaria. Todos estos pagos se realizan externamente al eCommerce. En el caso de tarjeta de crédito, el banco provee unas claves y códigos de encriptación, con las que mediante un módulo de conexión con bancos, se realiza la comunicación en ambos lados entre eCommerce y el banco. El funcionamiento es el siguiente: el eCommerce se identifica en el banco, le envía todos los productos, datos del cliente y pagos que el cliente debe realizar. El cliente en el momento de realizar el pago, salta a la pasarela de pago del banco, introduce los números e identificadores de su tarjeta, y realiza el pago. Una vez realizado, el banco le envía de vuelta al eCommerce, enviándole a este la comprobación del pago del cliente. En el caso de PayPal se realiza el mismo proceso, pero el cliente tiene opción a pagar con su propia cuenta de PayPal en caso de tenerla. Por otro lado, tenemos los métodos de pago manuales: contra reembolso, el cliente realiza el pago en el momento que recibe su compra, y la transferencia bancaria, el cliente debe realizar una transferencia a la cuenta que se informa en la web en el momento que se selecciona dicha forma de pago. Esta información se introduce en el administrador del eCommerce para informar al cliente. Una vez realizada, de

forma manual, el administrador consulta su banco para verificar la transferencia y procesa el pedido. Podemos ver que todos los pagos se realizan de forma que la responsabilidad y seguridad del pago queda presente en las empresas encargadas del cobro, no en el eCommerce. Si se diera el caso que el cliente tuviera que introducir el número de tarjeta en el eCommerce, la responsabilidad pasa a ser del administrador de este y es necesario hacer uso de sistemas de seguridad adicionales.

Un eCommerce también contiene herramientas para configurar métodos de envío. Se puede vender todo tipo de producto, como producto digital (no contiene gastos adicionales de envío para el cliente), producto descargable y producto físico. En este último el administrador del eCommerce debe realizar la gestión del envío del producto a la destinación que pida el comprador. Este proceso acostumbra a tener un coste adicional, y este coste va en función de la distancia, peso o cantidad de productos que solicite el comprador. Los eCommerce permiten integrar diferentes métodos de envío, como un coste fijo denominado tarifa plana, un coste por distancia, peso o cantidad, o configuraciones diversas que se imponen desde las compañías de transporte que realizaran el envío y entrega del producto.

Además de la gestión de compra y venta, también se deben configurar los diferentes impuestos y configuración legal para poder realizar comercio, pedidos, facturas, tiquetes compra...

Un eCommerce contiene todo lo necesario para poder realizar venta/comercio de productos mediante otro canal no presencial, como es internet. En un eCommerce, se crean inconvenientes o dificultades de tareas, nos brinda la oportunidad de vender el contenido de una tienda física en internet, pero en una tienda física podemos ver y tocar, en un eCommerce debemos exponer y enseñar este producto para que el cliente se haga una idea prácticamente real de este. La compra online no permite a los clientes poder tocar y probar el producto antes de comprarlo, ya que en internet es prácticamente imposible. Por esta razón, la explicación e imagen del producto es importantísimo para mostrar al cliente, que ese, es el producto que busca o desea. La introducción de toda esta información genera un problema grave a la hora de construir un eCommerce. Esta tarea es lenta y pesada, y en muchos

casos, según la magnitud o almacén de cada negocio, hay todo un departamento encargado de entrar y revisar todos los datos del producto, ya que es necesario introducir todos los datos por teclado, manualmente.

En este proyecto se usa el sistema open source Magento[9]. Este es un sistema para el desarrollo de eCommerce. Magento es robusto y estable, ya que gracias a sus principales características té permite configurar un sistema de envergadura y adaptado a todas las necesidades de un eCommerce.

Su primera característica principal es la velocidad. Su algoritmo está optimizado para una rápida ejecución. No obstante, es muy importante el servidor de internet donde se instala, para poder sacar el máximo rendimiento de este. Su distribución de código mediante un núcleo principal y su estructura modular, té permite una gran optimización y distribución con el que se ejecuta todo de forma ordenada y eficiente. El hecho de estructurarse de esta forma nos lleva a otra de sus características principales, es un sistema actualizable. Se puede ir actualizando nuestra tienda o tiendas virtuales según necesidades. Lo más importante de ser actualizable, es la seguridad del sistema, que día a día es necesaria con las vulnerabilidades que se generan con los nuevos virus e incursiones que se realizan por parte de los hackers.

Otra de las características principales es la escalabilidad. Magento permite crear y gestionar múltiples sitios Web y tiendas desde un administrador. Pudiendo crear tantas tiendas diferentes como pueda hacer falta (juguetes, ropa, zapatos) en una misma instalación. Cada una de estas puede ser multi-lenguaje. Toda la infraestructura de una tienda Magento está orientada a poder crear tantos idiomas como sea necesario. Y finalmente Magento dispone de una amplia configurabilidad. Se puede configurar y crear una tienda Magento con los mismos productos pero orientados a segmentos de mercados diferentes.

Magento permite disponer desde una pequeña, mediana y gran empresa con una estructura y robustez ejemplar. Al ser una plataforma construida sobre el lenguaje PHP[10], más específicamente sobre Zend Framework[11] (entorno de desarrollo creado por la misma gente de PHP) nos ofrece una de las plataformas más seguras del mundo eCommerce.

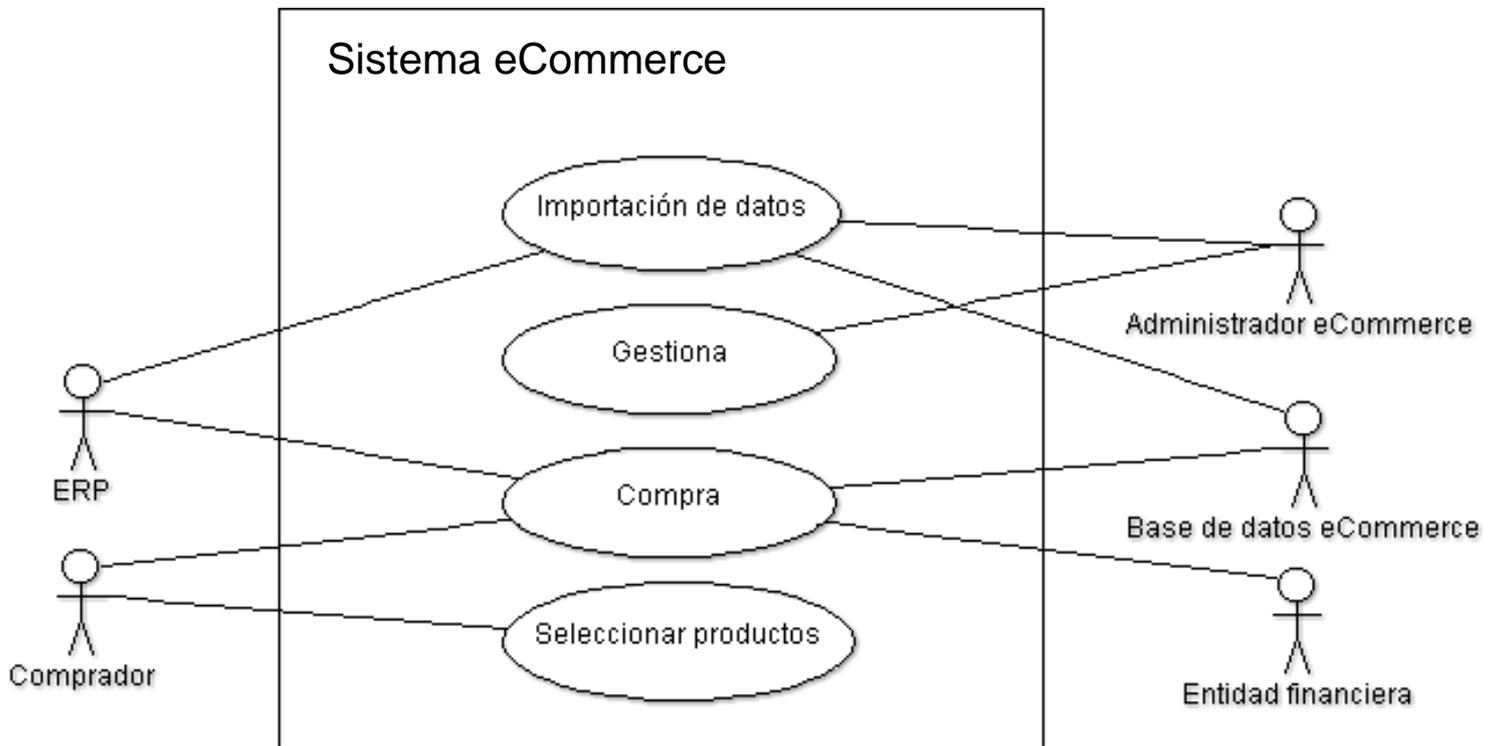
Otros sistemas eCommerce muy utilizados en la actualidad son osCommerce[12] y Prestashop[13]. Oscommerce fue de los primeros sistemas *Open Source* utilizados para construir tiendas online. Este sistema es un sistema programado en lenguaje PHP, construido de forma estructurada y eficiente. Este sistema da buenos resultados de cara a velocidad y sencillez, pero al ser un sistema no orientado a objetos, es difícil y laborioso realizar ampliaciones y actualizaciones. Oscommerce es el antecesor de Magento, ya que en el pasado parte del equipo de osCommerce se dio cuenta que el sistema era antiguo y carecía de potencia para poder adaptarse y seguir evolucionando para las nuevas características y comunicaciones del momento. Esta parte del equipo se embarcó en la construcción de un sistema orientado a objetos y con una base de datos adaptada a estos. Además crearon un núcleo de sistema con el que hoy en día se puede actualizar y ampliar Magneto sin afectar al resto de ampliaciones. Por este motivo, con el tiempo Magento ha evolucionado y sigue estando al día de todas las tecnologías actuales. Por otro lado, osCommerce ha quedado como un sistema obsoleto del cual se puede encontrar muchísima información y ayuda de foros por internet, pero carece de soporte ya que es un sistema que ha dejado de evolucionar.

Por otro lado existe el sistema Prestashop, que día a día evoluciona por ser la competencia más directa de Magento. Prestashop es un proyecto fin de carrera, nacido de Francia el cual ofrece más sencillez y velocidad que Magento. Carece de características y robustez en comparación con Magento, pero los comerciantes de hoy en día, le están dando mayor aprecio y más salida.

Prestashop es una buena solución para pequeños negocios o empresas con cierta envergadura que no va a pasar de cierto volumen, su fácil funcionamiento y configuración, lo hacen un sistema potente y a la vez económico. Pero si se busca montar un sistema robusto, para grandes, pequeñas y medianas empresas, con intención de crecimiento, Magento es un sistema probado, eficaz y potente.

### 3. Análisis

#### 3.1. Diagrama de casos de uso



## 3.2. Casos de uso

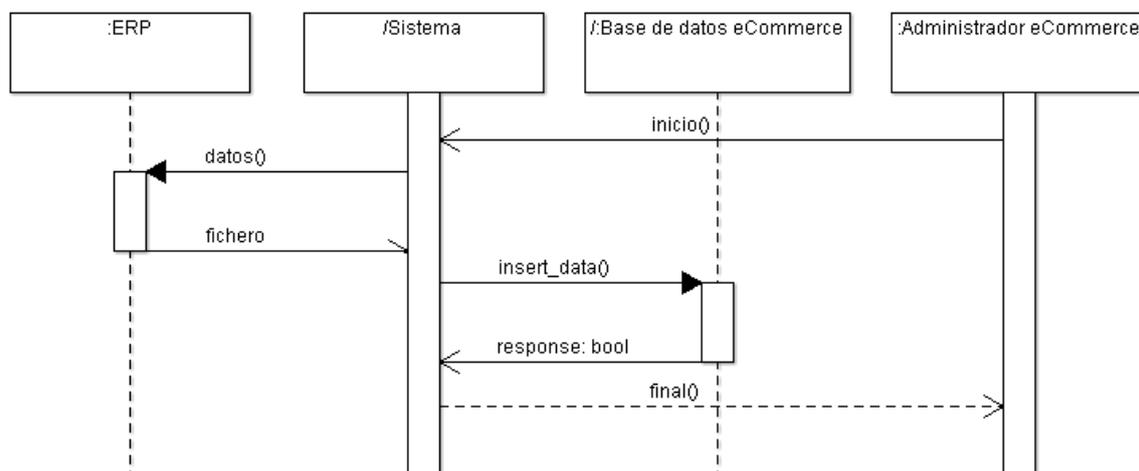
### UC1-Importación de datos



Nombre: UC1-Importación de datos
Actores: ERP, Administrador eCommerce, Base de datos eCommerce
Descripción: El administrador ejecuta el importador y se sincronizan los datos del ERP con los datos del eCommerce
Precondiciones: El ERP ha generado un archivo con el contenido de datos a importar y lo ha insertado en una carpeta predeterminada. El ERP ha analizado que las imágenes de todos los artículos a importar existen y están en la carpeta que corresponde.
Flujo normal: <ol style="list-style-type: none"> <li>1. Se extraen los datos del archivo y se colocan en variables de sistema</li> <li>2. Los datos se organizan y se construyen en variables con el formato que usa el eCommerce</li> <li>3. Introducimos los datos en la base de datos eCommerce</li> </ol>
Flujo alternativo: 1a. Datos incorrectos, se genera txt de errores y se para el proceso. 1b. Fichero inexistente o incorrecto, el importador se para y no procesa nada, esto queda registrado en un txt de errores. 2a. Faltan datos necesarios para poder crear un producto en el eCommerce, no procesamos el producto y lo anotamos en el txt de información. 2b. El eCommerce está realizando otros procesos y fallan las funciones de inserción i/o actualización, se genera un error y queda registrado en el txt de errores. 3a. Falla la conexión con la base de datos, queda registrado en el txt de información. 3b. Los datos están incompletos, se guarda el resultado en un archivo txt y se para el proceso.
Poscondiciones: Los productos del ERP, existen y están actualizados en el eCommerce.

Diagrama de secuencia del sistema:

- Flujo normal



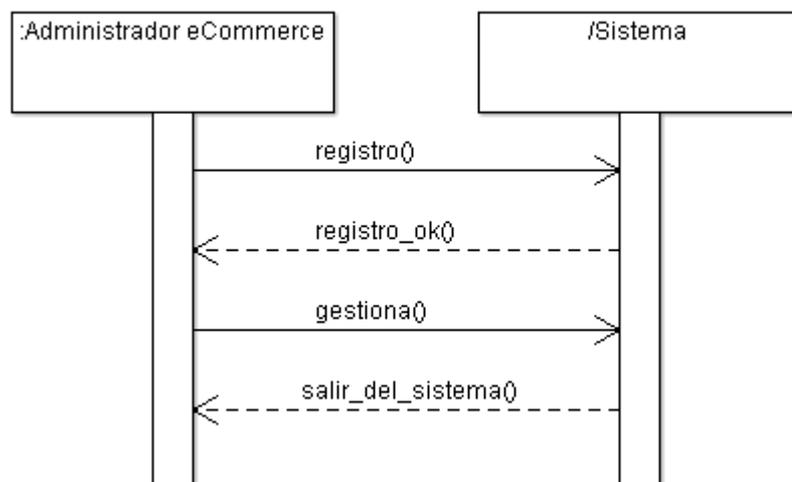
## UC2-Gestiona

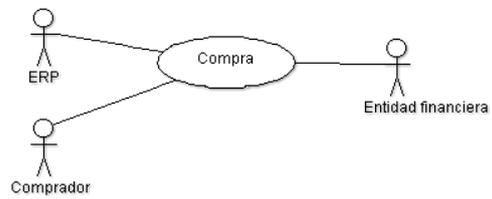


Nombre: UC2- Gestiona
Actores: Administrador eCommerce
Descripción: El Adminisitrador eCommerce gestiona el eCommerce para realizar los cambios que sean necesarios para garantizar el correcto funcionamiento de este.
Precondiciones: El Administador eCommerce debe tener un usuario y contraseña con permisos suficientes para la gestión total del eCommerce.
Flujo normal: 1. Administrador eCommerce accede al panel de administración 2. Gestiona las ventas, catalogo, usuarios y parámetros e información
Flujo alternativo: 1a. El usuario o contraseña son incorrectos, se debe solicitar un cambio de contraseña mediante el email que se relaciona con el eCommerce en el momento de su instalación
Poscondiciones: El Administrador eCommerce deja el sistema eCommerce configurado.

Diagrama de secuencia del sistema:

- Flujo normal



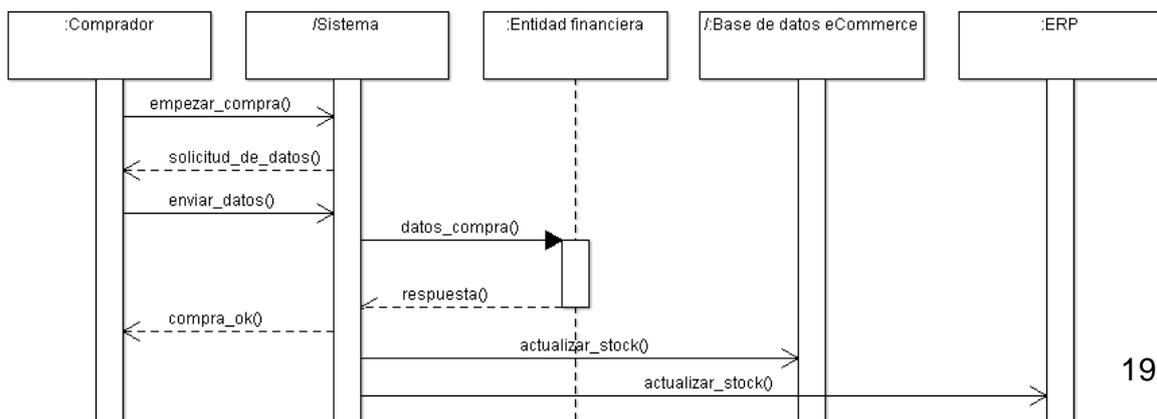


## UC3-Compra

Nombre: UC3-Compra
Actores: Comprador, ERP, Entidad financiera, Base de datos eCommerce
<p>Descripción:</p> <p>El comprador efectúa una compra en el eCommerce realizando un pago en la entidad financiera. El eCommerce actualiza los stocks de los productos vendidos e informa al ERP.</p>
<p>Precondiciones:</p> <p>El comprador ha seleccionado el producto/s que quiere comprar. Para que el comprador pueda realizar la compra de un producto, este debe tener stock.</p>
<p>Flujo normal:</p> <ol style="list-style-type: none"> <li>1. El comprador selecciona comprar en el sistema.</li> <li>2. El eCommerce solicita los datos del cliente para realizar la compra.</li> <li>3. El cliente introduce los datos.</li> <li>4. Se realiza el pago en la entidad financiera.</li> <li>5. Se confirma el pago en el eCommerce.</li> <li>6. Se actualizan los stocks de los productos vendidos en el eCommerce</li> <li>7. El eCommerce envía un email con los datos de los productos vendidos y la cantidad al ERP.</li> </ol>
<p>Flujo alternativo:</p> <p>3a. El cliente no introduce los datos requeridos para la compra, no se completa el proceso de compra y se anula el proceso.</p> <p>4a. El comprador no realiza el pago, se cancela la compra y el stock no se modifica.</p> <p>4a. El comprador realiza el pedido, pero no se confirma el pago en el eCommerce, se cancela la compra y el stock no se modifica.</p> <p>7a. El email no es correcto, como en el almacén si se va a realizar la actualización de stock a la hora de tramitar el pedido, cuando el ERP se actualice con los datos de almacén se van a actualizar los stocks correctamente.</p> <p>7b. El email no se envía, queda registrado en el log de sistema.</p>
<p>Poscondiciones:</p> <p>Para considerar la compra realizada y procesar la actualización de stock, se debe confirmar el pago del producto/s vendido/s. Una vez confirmada la compra, el stock de productos del eCommerce queda actualizado en el eCommerce y en el ERP al mismo tiempo.</p>

Diagrama de secuencia del sistema:

- Flujo normal



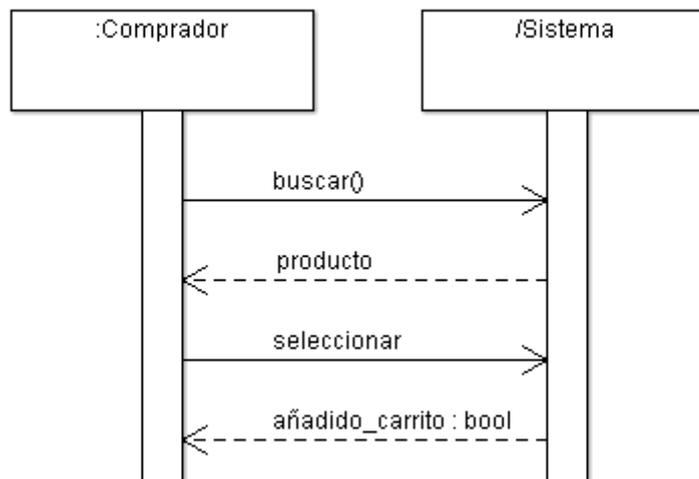
## UC4-Seleccionar productos



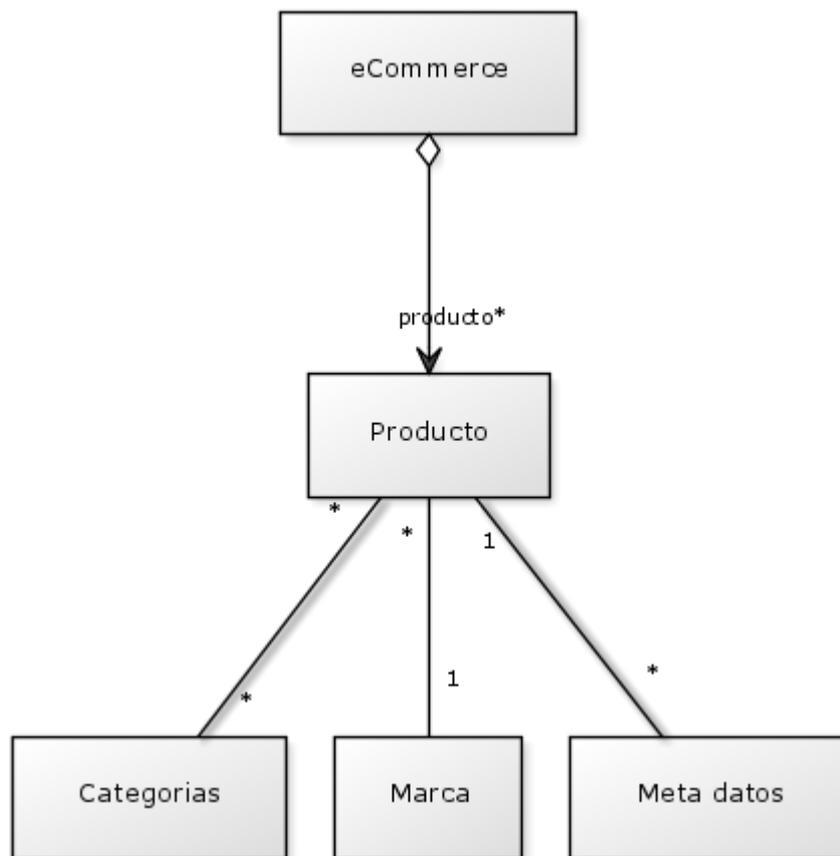
Nombre: UC4-Seleccionar productos
Actores: Comprador
Descripción: El comprador accede al eCommerce y puede visitar todo el catalogo del eCommerce. En este puede ver todo el producto de forma ordenada y estructurada.
Precondiciones: El comprador accede al eCommerce mediante un navegador web.
Flujo normal: 1. El comprador recorre el catalogo en busca del producto que le interesa. 2. El comprador selecciona los productos que desea comprar.
Flujo alternativo: 1a. El comprador no encuentra el producto que busca, no selecciona ningún producto y no realiza ninguna compra.
Poscondiciones: Una vez el comprador ha visitado el catalogo del eCommerce y ha seleccionado los productos que desea comprar, puede realizar la compra de estos, el proceso de compra lo podemos ver en el UC3.

Diagrama de secuencia del sistema:

- Flujo normal



### 3.3. Definición del modelo de dominio



### 3.4. Requerimientos generales: software y hardware

#### *Requerimientos de software:*

Para poder instalar y configurar el eCommerce con un sistema Magento, es imprescindible que la computadora donde se aloje el eCommerce tenga instalado un servidor apache(librerías y sistema encargado de interpretar y convertir código php<sup>1</sup> en html<sup>2</sup>), mysql<sup>3</sup>(sistema para la creación y gestión de base de datos), sistemas de cache<sup>4</sup>, como APC<sup>5</sup> (*Alternative PHP Cache*) cache, o usar un sistema CDN<sup>6</sup>(*content distribution network*), que permite montar el contenido de imágenes y css en un servidor y la base de datos y código fuente en otro. Al realizar esta separación se consigue una carga simultánea de ambas partes y así se incrementa notablemente el rendimiento del eCommerce.

<sup>1</sup>php: es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.

<sup>2</sup>html: (*HyperText Markup Language*) es el lenguaje de marcado para la elaboración de páginas web y el código que interpretan todos los navegadores web para mostrar las páginas.

<sup>3</sup>mysql: es un sistema de gestión de bases de datos relacional, multihilo y multiusuario.

<sup>4</sup>Cache: Parte de memoria de proceso, reservada para guardar el código que se debe usar, de esta forma su acceso es más rápido e inmediato.

<sup>5</sup>Cache APC: Sistema de cache de código libre, para proporcionar un funcionamiento robusto de mayor velocidad del código PHP.

<sup>6</sup>Cache CDN: Es un gran sistema distribuido de servidores desplegados en múltiples centros de datos a través de Internet. Sirven para poder separar datos de código, con datos de imágenes y estilos.

#### *Requisitos de hardware:*

Para la instalación de un sistema eCommerce con Magento y un catálogo con una envergadura de entre 20.000 y 50.000 productos, es necesaria una computadora remota que sea capaz de mover todo el sistema con fluidez, para esto es importante disponer de unos requisitos mínimos de configuración con más de 8GB de memoria RAM, ya que la cantidad de visitas y datos que procesa cargan mucho la memoria RAM. También es importante el uso de una CPU de varios núcleos que sea capaz de procesar grandes cantidades de datos y finalmente imprescindible un disco duro con más de 30GB de capacidad, para poder albergar la gran base de datos que se genera al contener un catálogo de más de 50.000 productos y todas sus imágenes.

Para este proyecto se ha hecho uso de un servidor SIP-300 de la empresa americana NEXCESS[14]. El servidor remoto donde se alberga Mountlegend dispone contiene las siguientes características:

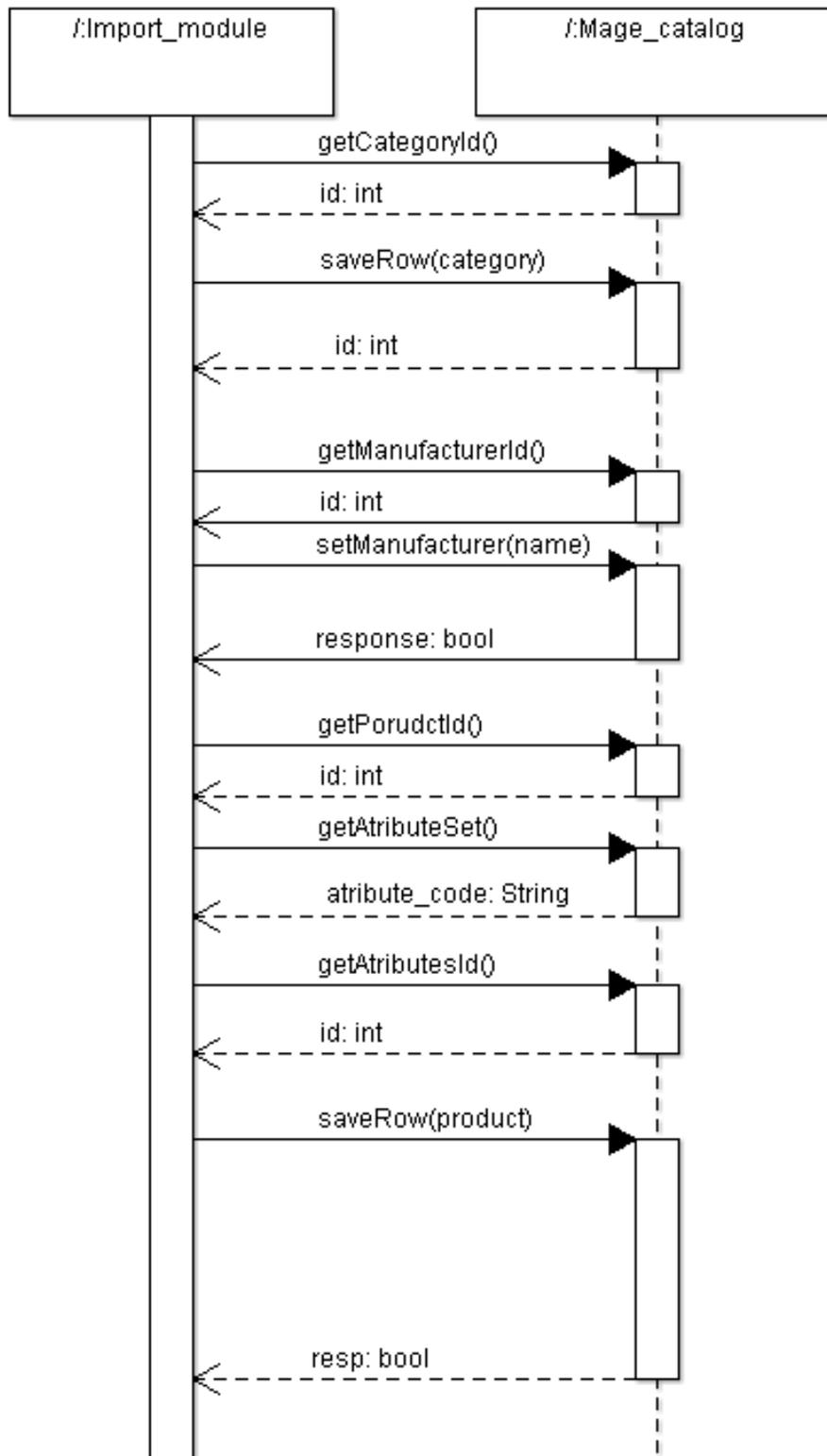
- Procesador: Dual E5-2420 Hex Core
- Disco duro: 55GB
- Memoria RAM: 24GB
- Sistema CDN: 250GB

Requisitos de hardware más que suficientes para el proyecto.

## 4. Diseño

### 4.1. Diagrama de secuencia

#### Diagrama de secuencia



En el diagrama de secuencia podemos ver la interacción entre el módulo de importación y la clase Mage\_catalog de Magento. Inicialmente el módulo de importación debe pedir a Magento el identificador de la categoría, si ya existe, la actualiza, en caso de llegar vacía se crea una categoría nueva. Los dos procesos los realiza la función saveRow.

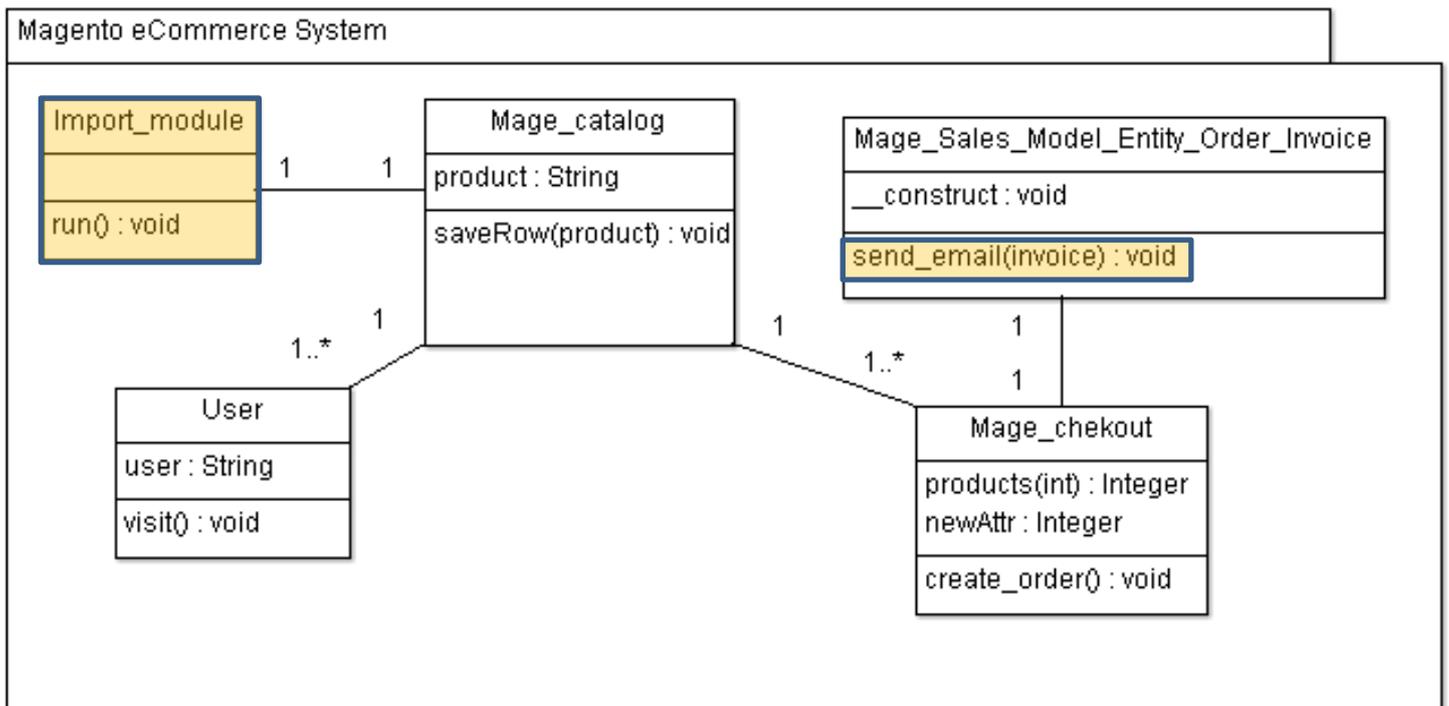
En segundo lugar pide el identificador del manufacturer y en caso de existir ya tenemos el identificador de este, en caso de no existir, se crea de nuevo mediante la función setManufacturer.

Finalmente, vemos la interacción para sincronizar productos. En primer lugar comprobamos la existencia del producto pidiendo el identificador de este a Magento. Tanto si existe como si no existe, necesitamos los identificadores de las estructuras de datos que forman el producto en Magento. Por este motivo, a continuación pedimos a Magento el identificador del grupo de atributos y los identificadores de los atributos que lo precisen. Una vez tenemos los datos necesarios mediante la función saveRow, creamos o actualizamos el producto en Magento.

## 4.2. Diagrama de clases

En este proyecto se hace uso de un sistema open source eCommerce, llamado Magento.

Mediante código PHP definimos una clase principal de importación que hace uso de las diferentes clases de entidades que importamos en el eCommerce.



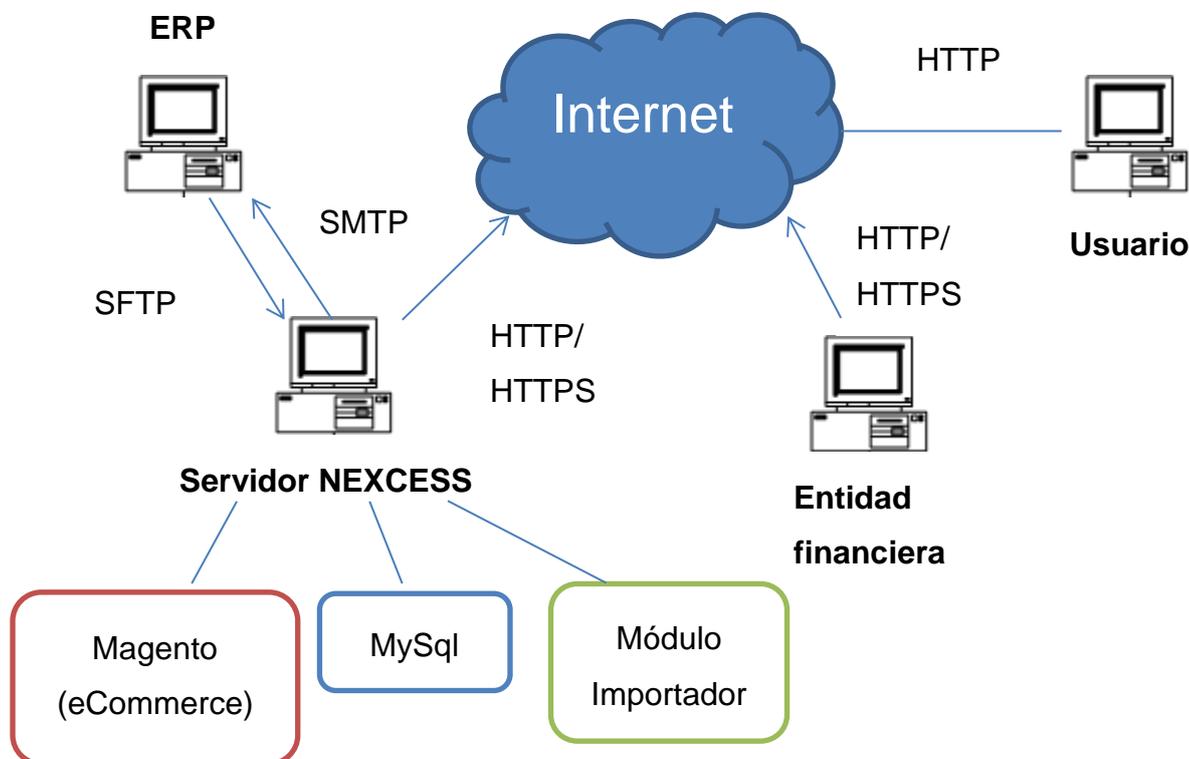
*Diagrama de clases simplificado*

*(Aparecen en amarillo los apartados que se han implementado en este proyecto)*

En este diagrama vemos la estructura interna que va a tener el eCommerce una vez integremos la importación. En el diagrama se puede apreciar la clase `import_module` que se encarga de actualizar el catálogo. Este es visitado por los usuarios los cuales realizan compras. Estas compras se encargan de generar facturas, con la cual hemos modificado la función básica “`send_email`” de Magento para integrar una funcionalidad nueva en el envío de email de factura. En esta enviamos un segundo email al ERP para informarle de la actualización de stock que se ha producido en la factura enviada.

## 5. Implementación

Para iniciar la implementación del proyecto, he realizado a continuación, un diagrama de la distribución del proyecto [Ilustración 1].



*Ilustración 1: Distribución del proyecto*

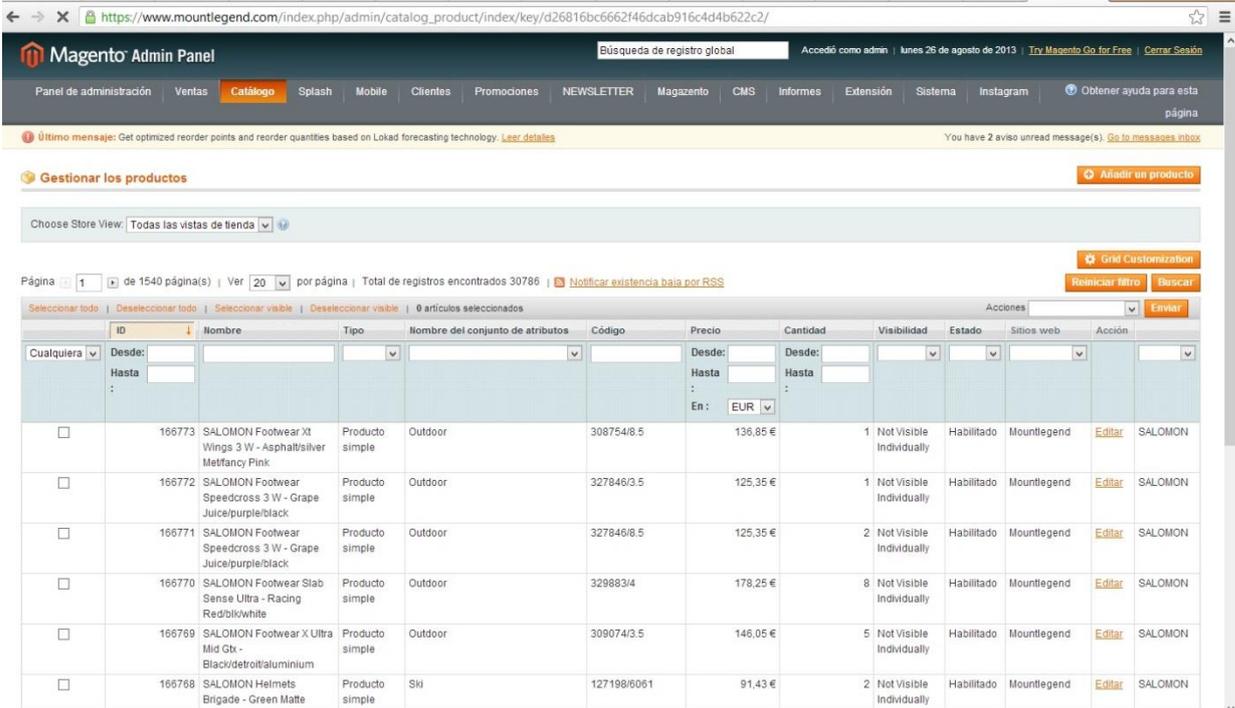
El ERP está instalado en local en una computadora externa. Por otro lado, el servidor de NEXCESS (se encuentra en EEUU) alberga el sistema eCommerce, MySQL y el módulo importador. Se genera una doble comunicación entre estos, el ERP envía por SFTP al servidor los ficheros que contienen la información a sincronizar, y el servidor de NEXCESS envía por email al ERP, la actualización de stock. A la vez está conectado a Internet mediante HTTP para que el eCommerce pueda ser visitado en Internet. Por este motivo, en la distribución se especifica la computadora de los usuarios, que se conecta a internet para poder visitar y realizar compras en el eCommerce. Cada vez que un cliente realiza una compra y desea pagar con tarjeta, el servidor de NEXCESS se comunica mediante http o https (en función de la entidad) con la entidad financiera, para que el usuario pueda realizar la compra en el servidor de esta.

## 5.1. Instalación eCommerce

El proceso inicia en el momento que el cliente hace petición de un proyecto eCommerce con sincronización con su sistema de almacén.

El cliente pone sobre la mesa el proyecto y sus necesidades, explicadas en los apartados de introducción y objetivos.

Inicialmente se realizó la instalación del Magento. Este proceso se realiza en el servidor de NEXCESS, donde se crea una base de datos y se lleva a cabo el instalador del sistema. En la instalación se define un usuario y contraseña que serán las credenciales para que el administrador del sistema pueda acceder a él y realizar la gestión de este. Con el Backend de Magento [Ilustración 2] implementamos el UC2.



The screenshot shows the Magento Admin Panel interface. At the top, there's a navigation bar with 'Magento Admin Panel' and a search bar. Below that, a menu bar includes 'Panel de administración', 'Ventas', 'Catálogo', 'Splash', 'Mobile', 'Clientes', 'Promociones', 'NEWSLETTER', 'Magazento', 'CMS', 'Informes', 'Extensión', 'Sistema', and 'Instagram'. A message bar indicates 'Último mensaje: Get optimized reorder points and reorder quantities based on Lokad forecasting technology'. The main content area is titled 'Gestionar los productos' and features a 'Choose Store View' dropdown set to 'Todas las vistas de tienda'. Below this, there's a pagination bar showing 'Página 1 de 1540 página(s)' and 'Ver 20 por página'. The main table displays a list of products with the following data:

ID	Nombre	Tipo	Nombre del conjunto de atributos	Código	Precio	Cantidad	Visibilidad	Estado	Sitios web	Acción
166773	SALOMON Footwear Xt Wings 3 W - Asphalt/silver Metfancy Pink	Producto simple	Outdoor	308754/8.5	136,85 €		1 Not Visible Individually	Habilitado	Mountlegend	Editar SALOMON
166772	SALOMON Footwear Speedcross 3 W - Grape Juice/purple/black	Producto simple	Outdoor	327846/3.5	125,35 €		1 Not Visible Individually	Habilitado	Mountlegend	Editar SALOMON
166771	SALOMON Footwear Speedcross 3 W - Grape Juice/purple/black	Producto simple	Outdoor	327846/8.5	125,35 €		2 Not Visible Individually	Habilitado	Mountlegend	Editar SALOMON
166770	SALOMON Footwear Slab Sense Ultra - Racing Red/blk/white	Producto simple	Outdoor	329883/4	178,25 €		8 Not Visible Individually	Habilitado	Mountlegend	Editar SALOMON
166769	SALOMON Footwear X Ultra Iliid Gtx - Black/detroit/aluminium	Producto simple	Outdoor	309074/3.5	146,05 €		5 Not Visible Individually	Habilitado	Mountlegend	Editar SALOMON
166768	SALOMON Helmets Brigade - Green Matte	Producto simple	Ski	127198/6061	91,43 €		2 Not Visible Individually	Habilitado	Mountlegend	Editar SALOMON

Ilustración 2: Backend de Magento

Una vez instalado se realiza una copia del template(conjunto de archivos que forman toda la imagen visible por el usuario mediante el navegador web) que se modificara para darle diseño al sistema.

El diseñador realiza unos diseños con Photoshop (programa de edición de imagen). Una vez el cliente (Mountlegend) acepta los diseños, el diseñador los monta en HTML. Una vez tenemos las paginas en HTML se integran en el template que hemos copiado. Esta tarea es larga y complicada, ya que se realiza una integración de HTML a PHP y se debe mantener todos los bloques y estructuras de Magento.

Una vez está el template integrado, tenemos que estructurar las categorías y los atributos de la tienda que se van a importar del ERP.

Todos estos procesos se realizan para el cliente pueda acceder al eCommerce y realizar su visita por el catalogo [Ilustración 3], buscando o investigando los productos que le puedan interesar para proceder a su compra. Así pues, de esta forma implementamos el UC4.

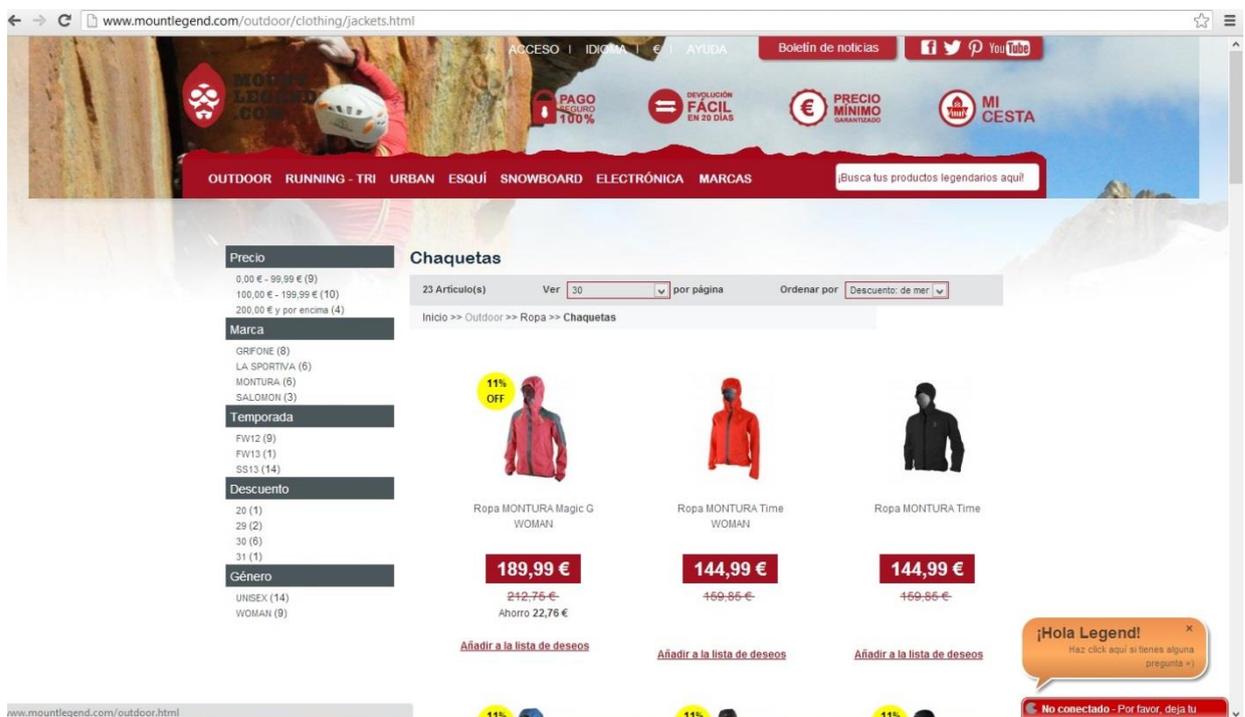


Ilustración 3: Frontend de Magento

Una vez esta implementada la imagen y la estructura del eCommerce, se configuran los métodos de pago. En este caso: Servired(pasarela de pago por tarjeta de crédito) y PayPal(banco por internet). También se configuran los métodos de envío, en este caso tenemos envío gratuito, internacional y Seur (empresa de transportes española).

Una vez tenemos gran parte del eCommerce a punto, falta por configurar las páginas estáticas, como son las políticas de privacidad, formas de pago, manual de uso; todas las páginas informativas que se deben poner en un eCommerce o que el cliente quiere poner para informar a los visitantes del eCommerce.

Estas son todas las fases que se realizaron para la construcción y configuración del eCommerce. En los siguientes apartados profundizo en los aspectos destacados de este proyecto.

## 5.2. Módulo de importación

En este apartado se desarrolla la implementación del caso de uso 1(UC1-Importación de datos). Este caso se ha llevado a cabo íntegramente con diseño, código y desarrollo propio.

La sincronización en el sentido ERP>eCommerce consiste en un proceso automático que actualiza únicamente los productos creados / modificados (Stocks incluidos) desde la sincronización anterior. El archivo de actualización genera el contenido necesario para la actualización en varios idiomas, SEO (meta datos). El proceso genera un log de proceso que permite hacer un seguimiento constante de las actualizaciones.

En la estructura de importación tengo que diseñar un archivo CSV (*comma-separated values*)<sup>1</sup>[15] que contenga toda la información necesaria para importar y estructurar el catalogo del eCommerce, de forma que tengo que crear estructuras para las categorías, atributos, metadatos y finalmente los productos. Este diseño se realizó en colaboración con el administrador del ERP.

La forma en que diseñamos los datos es extrapolar datos, es decir, guardar todos los datos ordenadamente en las tablas de importación y trabajar con ellos de forma ordenada y secuencial. Para guardar los datos cree una tabla temporal de productos, donde guardo todo el producto simple que contiene el CSV a importar. Así pues, tengo una tabla de productos, donde hay todo el producto que este en aquel momento en Magento, de esta forma puedo actualizar y/o crear el nuevo producto previamente en estas tablas. Este proceso se realiza porque Magento usa una base de datos orientada a objetos, así que usa muchas tablas para guardar todo el producto, cosa difícil y complicada para poder ver y comparar en caso de problemas con el CSV de importación. Con esto tengo una tabla muy semejante a los datos que recibimos del ERP y en caso de comprobación de errores me facilita mucho el trabajo. Con estas dos tablas tengo todo el producto simple, así que cree una tercera para guardar el producto configurable que se genera a partir de los simples.

<sup>1</sup> Son un tipo de documento que se representan en forma de tabla, como tal se estructuran con filas y columnas.

Por otro lado, cree una tabla para categorías y para marcas, donde guardo las categorías y controlo las marcas que contienen la tienda y su código que le pertenece en el eCommerce.

Y finalmente, cree otra tabla para los meta datos, que contienen los datos de SEO (meta datos)\* de cada producto en diferentes idiomas.

SEO (meta\_datos)\*: Los metadatos, son la información que se coloca en las etiqueta html, que Google lee y les da mucha importancia para posicionar una página de internet más arriba o más abajo en las búsquedas de Google. Por este motivo la práctica de ser experto en SEO se está convirtiendo en un trabajo de calidad y muy valorado.

### **5.2.1. Preparación de datos**

En este punto de la implementación vamos a realizar un resumen de la importancia de la preparación de datos (categorías, atributos y productos) antes del proceso de importación a Magento.

En la definición de categorías, teníamos que estructurar hasta tres niveles de categorías y con idiomas. El ERP funciona con un código interno con el cual denomina las categorías y lo usa para relacionar cada producto con su categoría o categorías.

Por otro lado, cada categoría puede tener categorías equivalentes, lo cual significa que un producto que forma parte de una categoría que tiene categorías equivalentes, también debe formar parte de las equivalentes.

El eCommerce como es independiente, usa otro identificador para poder relacionar productos y categorías, de forma que cree primeramente todas las categorías en el eCommerce y cogí todos los identificadores que crearon; en segundo lugar se introdujeron en el ERP y finalmente diseñe un CSV con todos los datos.

En el caso de este cliente es una tienda de productos de deportes, así que debía fijar un mínimo de atributos que en el catálogo del eCommerce ayuden al cliente a encontrar el producto que buscan lo más rápido posible, para poder favorecer a conseguir más ventas.

Así que, como atributos importantes definí el sku (identificador único), la marca, talla, color, genero, temporada, descuento, stock interno, stock externo,

stock mountlegend, meta datos, lenguaje, tipo de producto, descripción y descripción técnica.

Al definir estos atributos debía tener en cuenta que en el caso de los stocks Magento solo es capaz de manejar un stock y aquí tenía tres. Así que modifique Magento para trabajar con tres stocks. Esta modificación se explica en el apartado 5.3.

Por otra parte otros atributos importantes a tener en cuenta es la marca y temporada. Mediante la marca y temporada se define la estructura de carpetas que contienen todas las imágenes de los productos. Por ese motivo, necesitaba que las marcas no tuvieran caracteres especiales, y cree una importación de marcas que contenían el código de marca (denominado brandcode) y el nombre de la marca (manufacturer). La temporada ya contiene códigos que no contienen caracteres especiales.

Finalmente los atributos más importantes para la construcción del catálogo de Magento son la talla y color. Cuando se muestran estos productos en un eCommerce se construyen de forma que en el catálogo solo se muestra un producto, del cual se selecciona la talla y el color que el cliente desee y esté disponible en almacén. Para construir dicho producto se debe crear un producto configurable, así que mediante estos dos atributos a la hora de importar debía crear el producto configurable de cada grupo de productos simples.

En la definición del CSV, tenía que integrar en un solo fichero todos los productos simples y a ser posible con todos los datos necesarios para poder importarlos con Magento. Se plantearon varias dificultades, por ese motivo tuve que integrar todos los datos en este apartado y luego estructurarlos e importarlos por partes.

Una de las dificultades, fue los diferentes idiomas y los metas, con el formato que usa Magento para los idiomas, se debe adjudicar a cada vista de tienda(idioma) un valor diferente, así que para cada producto se generan tantas filas como idiomas tiene, cambiando solo los campos que contienen contenido con idioma diferente.

Para introducir diferentes idiomas se generan dos filas por producto, una contiene el contenido estándar y en inglés y el otro añade el contenido de metas y descripciones en español.

### **5.2.2. Sincronización eCommerce**

Una vez todos los datos estructurados y guardados en tablas, tenía que procesar todos los datos, así que lo estructuraré de forma secuencial: En primer lugar compruebo el archivo, si es de marcas, categorías o productos. Si es de productos mira la opción que plantea. Las opciones disponibles son: borrar toda una/s marca/s y volverla a importar, borrar todo un/os vendedor/es y volver a importar, reiniciar todo el catalogo o inserción/actualización simple. Una vez tenemos la opción a ejecutar, independientemente de esta, extraemos todos los datos del CSV y los colocamos en vectores para poder tratar la información.

A partir de aquí se ejecutan procesos diferentes. En caso de ser marcas se van a colocar los datos en la tabla de marcas y posteriormente se insertaran en el eCommerce con funciones de Magento. En caso de ser categorías, se insertaran en la tabla de categorías y se actualizara el id de cada categoría con el id del eCommerce. Finalmente si el archivo es de productos se procesa la opción que contenga el CSV. Las opciones de borrado, sirven principalmente para limpiar productos erróneos u obsoletos del eCommerce, por este motivo en caso de necesitar hacer una limpieza a fondo, tenemos la opción de reiniciar catálogo, con lo cual se elimina todo el producto del eCommerce y se reinserta de nuevo.

El caso típico es la actualización simple de producto. En este caso sencillamente se actualiza el producto que contiene el CSV en el eCommerce.

Se importan los productos simples, a continuación los configurables, que para poder importarlos, es imprescindible que los simples que los componen, existan en el eCommerce. Ya que, de lo contrario, se producirá un error en la importación. Finalmente, importamos todas las metas (atributos de información de los productos en varios idiomas).

Como ya he explicado, Magento dispone de una base de datos orientada a objetos, es decir, cada tabla contiene una parte de información y esto genera multitud de índices entre tablas. Cada vez que se genera una actualización de datos, debemos lanzar un proceso de Magento llamado reindexación, cuya misión es revisar todos los índices de las tablas que se enlacen correctamente, y en caso de haber índices incorrectos, eliminarlos. De esta forma una vez está la tabla reindexada, la información que se muestra en el eCommerce debe ser la misma que el CSV que nos provee el ERP.

Aquí finaliza el proceso de importación, y ya se puede comprobar que todo el producto se ha importado correctamente, en el archivo out.txt [Ilustración 4] que se genera en cada importación y tenemos todos los procesos que se han seguido.

Out.txt:

```

1 <h4>FICHERO ../var/import/ITM130823_2221A.CSV HA SIDO LEIDO</h4>
2 Se a eliminado correctamente los datos.
3 El producto : REDSTER D2 SL + NEOX TL12 OME BLA con categoria: 01.01.01 pertenece a la cats ids: 2671
4 METAS-----
5 METAS -> UPDATE CORRECTA!
6 El producto : RJ3 - BLACK/RED con categoria: 01.03.01 pertenece a la cats ids: 2695
7 METAS-----
8 METAS -> UPDATE CORRECTA!
9 El producto : RJ4 - BLACK/RED con categoria: 01.03.01 pertenece a la cats ids: 2695
10 METAS-----
11 METAS -> UPDATE CORRECTA!
12 El producto : RJ4 - BLACK/RED con categoria: 01.03.01 pertenece a la cats ids: 2695
13 METAS-----
14 METAS -> UPDATE SENSE CANVIS!
15 El producto : HACKETT JK con categoria: 01.05.01,02.04.01 pertenece a la cats ids: 2711,2792
16 METAS-----
17 METAS -> UPDATE CORRECTA!
18 El producto : KARLA con categoria: 01.05.01,02.04.01 pertenece a la cats ids: 2711,2792
19 METAS-----
20 METAS -> UPDATE CORRECTA!
21 El producto : WSXP NYLON RIPSTOP F con categoria: 01.05.01,02.04.01 pertenece a la cats ids: 2711,2792
22 METAS-----
23 METAS -> UPDATE CORRECTA!
24 El producto : ACTION POLY CANVAS STR con categoria: 01.05.01,02.04.01 pertenece a la cats ids: 2711,2792
25 METAS-----
26 METAS -> UPDATE CORRECTA!
27 El producto : FRACTURE YOUTH JKT con categoria: 01.05.01,02.04.01 pertenece a la cats ids: 2711,2792
28 METAS-----
29 METAS -> UPDATE CORRECTA!
30 El producto : DOME HOODIE con categoria: 01.05.01,02.04.01 pertenece a la cats ids: 2711,2792
31 METAS-----
32 METAS -> UPDATE CORRECTA!
33 El producto : VDLR PRINTED JKT con categoria: 01.05.01,02.04.01 pertenece a la cats ids: 2711,2792
34 METAS-----
35 METAS -> UPDATE CORRECTA!
36 El producto : TWC PRIZEFIGHTR PT con categoria: 01.05.02,02.04.02 pertenece a la cats ids: 2713,2794
37 METAS-----
38 METAS -> UPDATE CORRECTA!
39 El producto : MAJOR con categoria: 01.05.02,02.04.02 pertenece a la cats ids: 2713,2794
40 METAS-----
41 METAS -> UPDATE CORRECTA!

```

Journal text file length: 131072 lines: 1816

Ilustración 4: Fichero out.txt

### 5.3. Compra y actualización de stocks

En este apartado se implementa el caso de uso 3 (UC3 – Compra). En esta implementación se ha hecho uso de funcionalidad de Magento y programación propia.

Cuando se generan ventas en el eCommerce se actualizan los stocks de este, e indirectamente, el almacén puede saber el estado del stock.

En el caso de Mountlegend, tiene tiendas físicas en Andorra y el stock de los almacenes puede variar independientemente de las ventas del eCommerce, pero en el momento que se realiza una venta en el eCommerce, se envía el pedido al almacén y modifica su stock, de forma que cuando el ERP actualiza sus stocks mediante los almacenes, ya recibe el stock actualizado. Aun así este proceso es muy largo, ya que depende de cuánto tarde el almacén en preparar todo el proceso. Para poder evitar esta demora, he modificado Magento para que envíe un email al ERP con la actualización de stocks. De forma que, el ERP ya está actualizado antes de recibir los nuevos stocks, además puede generar estadísticas de ventas en tienda eCommerce.

No obstante, referente a los stocks, se ha creado otra modificación en Magento, para implementar tres stocks en Magento, en esta misma función donde enviamos el email con la actualización de stocks, también actualizamos los atributos de stock interno, Mountlegend y externo (atributos internos de Magento que se usan para informar al cliente el tiempo de entrega de cada producto [*Ilustración 5 y 6*], en función de donde provenga el stock que tiene cada uno).

De esta forma en la ficha de producto donde informamos al cliente del tiempo de entrega, comprobamos el stock que le queda a cada producto (en los productos configurables, miramos todos los hijos).

www.mountlegend.com/ofertas/timberland-footwear-chocorua-mid-gtx-brown-green-149665.html

ACCESO | IDIOMA | € | AYUDA | Boletín de noticias | f | t | p | YouTube

PAGO Seguro 100% | DEVOLUCIÓN FÁCIL EN 20 DÍAS | PRECIO MÍNIMO GARANTIZADO | MI CESTA

OUTDOOR | RUNNING - TRI | URBAN | ESQUÍ | SNOWBOARD | ELECTRÓNICA | MARCAS | ¡Busca tus productos legendarios aquí!

Inicio >> Destacados >> Calzado TIMBERLAND Chocorua Mid Gtx - Brown/green

Se el primero en opinar sobre este producto

SKU : 15130  
Género : UNISEX  
Temporada : SS13

\*1. Talla  
Selecione una opción...

### Calzado Timberland Chocorua Mid Gtx - Brown/green

Disponible para envío: **Inmediato\***

**114,99 €** ~~126,49 €~~  
IVA incl.

1 **Añadir al carrito**

Pasar por caja  
 [Añadir a la lista de deseos](#)

Tiempo de entrega que se modifica en función del stock

Información adicional | Opiniones | Share on

¡Hola Legend! Haz click aquí si tienes alguna pregunta ->

Conectado - Haz click aquí si tienes

Ilustración 5: Vista de producto con stock interno o mountlegend

www.mountlegend.com/snowboard/boards/freeride/burton-boards-restricted-social-148921.html

ACCESO | IDIOMA | € | AYUDA | Boletín de noticias | f | t | p | YouTube

PAGO Seguro 100% | DEVOLUCIÓN FÁCIL EN 20 DÍAS | PRECIO MÍNIMO GARANTIZADO | MI CESTA

OUTDOOR | RUNNING - TRI | URBAN | ESQUÍ | SNOWBOARD | ELECTRÓNICA | MARCAS | ¡Busca tus productos legendarios aquí!

Inicio >> Snowboard >> Tablas >> Freeride >> Tablas BURTON Restricted Social

Se el primero en opinar sobre este producto

SKU : 276082  
Género : WOMAN  
Temporada : FW12

\*1. Talla  
Selecione una opción...

### Tablas Burton Restricted Social

Disponible para envío: **10-15 días\***

**328,99 €** ~~366,70 €~~  
Ahorro **36,71 €**

IVA incl.

1 **Añadir al carrito**

Pasar por caja  
 [Añadir a la lista de deseos](#)

Detalles | Información adicional | Opiniones | Share on

BEND: V-Rocker SHAPE: Twin FLEX: Twin CORE: Women's-Specific True Flex Fly® Core FIBERGLASS: Blax

¡Ho

Ilustración 6: Vista de producto con stock externo

Para realizar el mail de actualización de stocks, se ha creado una plantilla de email usando el generador de plantillas de Magento [Ilustración 7 y 8], y se ha generado un código en la función que captura el evento de facturación de Magento para mandar el email en el momento que se genera la factura, ya que en Magento cuando se genera la factura es en el momento que certificamos la venta y realmente se confirma la actualización de stock.



Ilustración 7: Vista del listado de emails en el backend de Magento

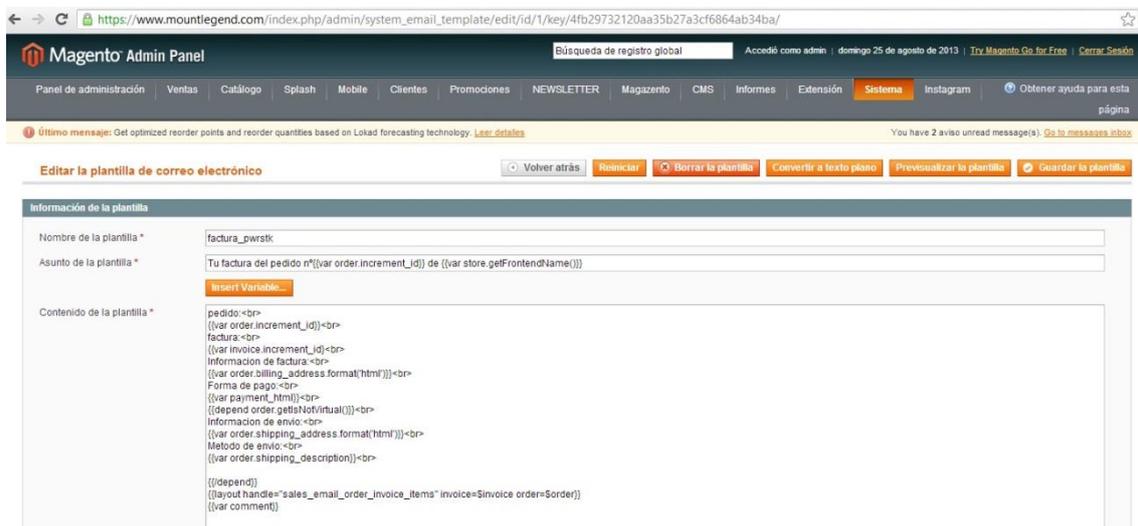
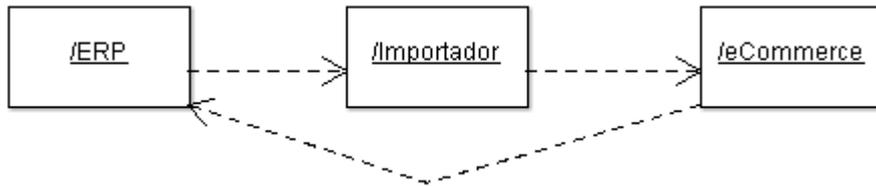


Ilustración 8: Vista del editor de emails en el backend de Magento

## 5.4. Diagrama de componentes



En este diagrama se ve los componentes que tenemos en el proyecto y las comunicaciones que se generan entre ellos. El ERP se comunica con el importador para transferirle datos mediante un archivo CSV. El importador se comunica con el eCommerce para transferirle mediante funciones de inserción/actualización los datos del CSV del ERP, y finalmente cada vez que se genera una venta en el eCommerce, este informa al ERP para que actualice los stocks de los productos vendidos.

En los apartados anteriores, implementamos los casos de uso 1 y 3, para la implementación de los casos de uso 2(UC2-Gestiona) y 4(UC4-Visita catalogo), se hace uso íntegramente de Magento, ya que son funcionalidades totalmente integradas en el sistema y ya vienen implementadas una vez se instala el eCommerce. Por esta razón con la instalación de Magento y la implementación y modificación propia realizada, consigo solucionar todos los casos de uso que plantea el proyecto.

Hasta aquí todo el proceso de implementación, una vez realizado todo el proceso, se realizó la fase de comprobación y pruebas, donde se realizaron importaciones de todo tipo y se comprobó producto por producto la visualización y corroboración de datos.

## 6. Mejoras propuestas

Después de la realización de pruebas y puesta en marcha del proyecto, me planteo un paquete de mejoras que me permitirán ampliar el abasto de tecnologías y procesos.

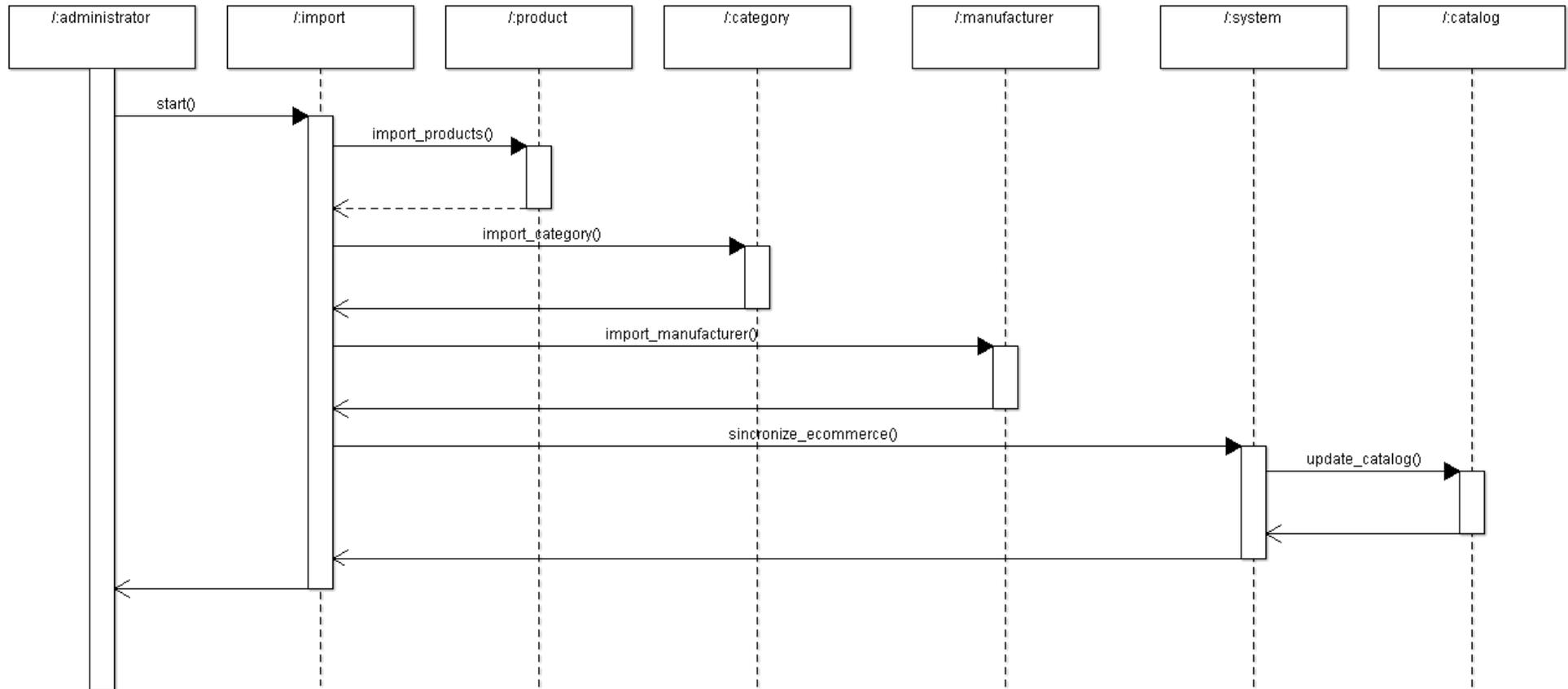
La realización de esta mejora es imprescindible para poder ampliar los clientes que puedan usar esta herramienta. Actualmente, dispongo de clientes que usan Magento y otros que usan Prestashop. Uno de los clientes que usan Prestashop me ha encargado el desarrollo de la sincronización entre PowerStock y Prestashop. Por este motivo, con esta ampliación quiero poder ofrecer la conexión entre eCommerce y ERP para ambas plataformas (Magento / Prestashop). Como he explicado en el proyecto, PowerStock es un ERP que ya están usando muchos clientes, así que los que ya disponen de una plataforma eCommerce se les podrá plantear una conexión, y los que no lo tengan se les puede plantear todo el proyecto.

En esta propuesta planteamos un módulo de importación orientado a objetos, que estructuran los datos de forma que flexibiliza el trabajo de estructuración de datos. Una vez tengo los datos en objetos, puedo manejar estos como necesite, y puedo crear diferentes conversiones para diferentes plataformas, en este caso para Magento y Prestashop.

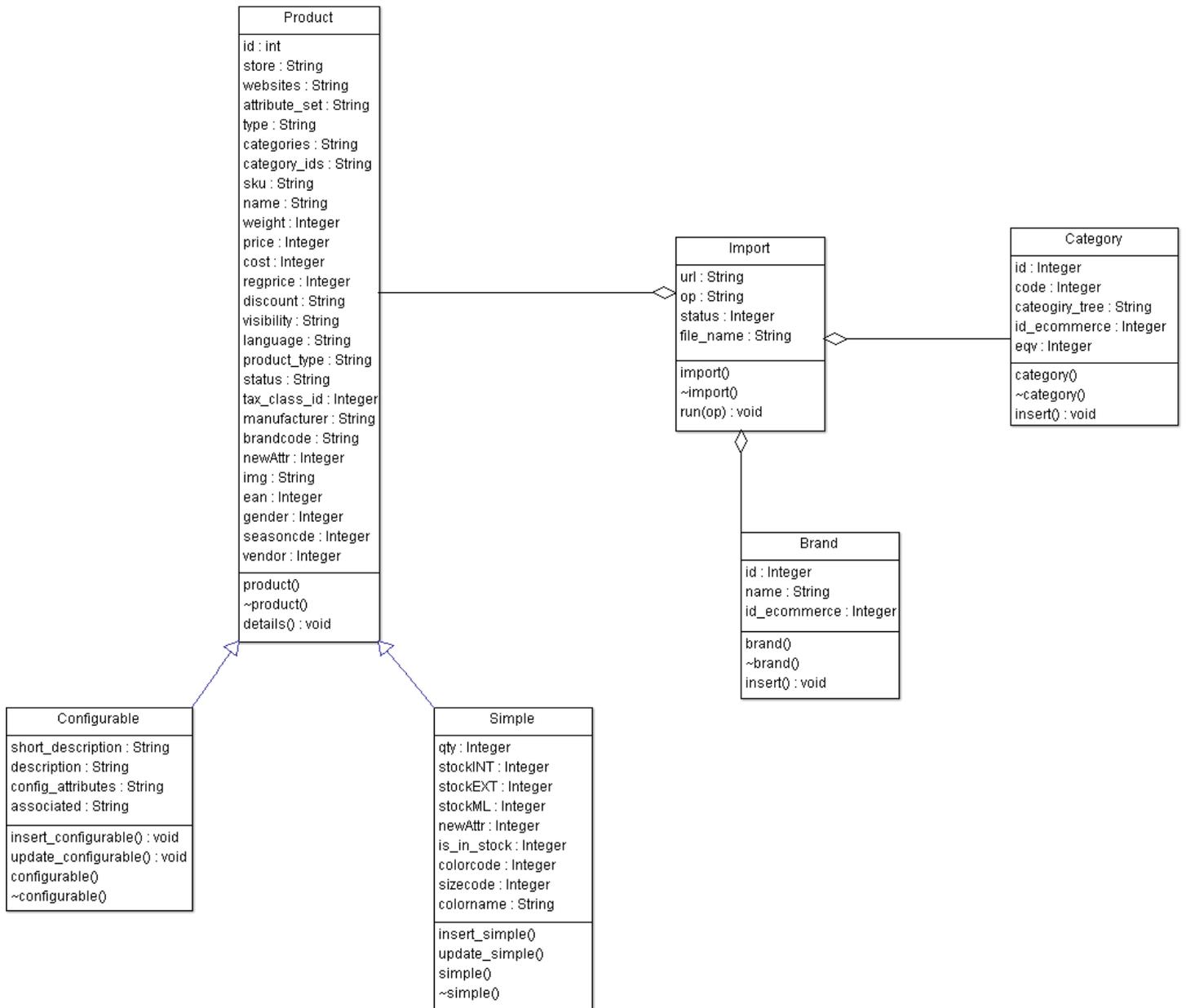
En estos momentos estoy en la fase de diseño del proceso con Prestashop. A continuación tenemos el diagrama de secuencia de las clases para poder preparar los datos para las plataformas.

## 6.1. Diseño

### 6.1.1. Diagrama de secuencia



## 6.1.2. Diagrama de clases



## 7. Conclusión

En este proyecto se ha creado un eCommerce y se ha diseñado un enlace de sincronización entre el eCommerce Mountlegend y el ERP PowerStock que contiene todo el catálogo de productos que se vende en Mountlegend. En la estructura entre los tres componentes, eCommerce, importador y ERP, se consigue optimizar el tiempo de inserción de datos. En negocios que ya estén haciendo uso de ERP, les permite optimizar recursos y añadir un punto de venta a su negocio.

En el proyecto inicialmente se ha realizado la instalación del eCommerce. Por una parte, se ha implementado la extracción de datos del ERP, se ha diseñado y desarrollado clases y funciones externas para guardar y estructurar los datos, para que sean compatibles con el eCommerce. Por otro parte, se ha modificado funciones de Magento para poder insertar y actualizar los datos de productos del eCommerce de forma automática.

Esta sincronización se ha realizado en doble sentido. En sentido ERP-eCommerce, todos los datos que existen en el ERP, también existen en el eCommerce. En sentido opuesto el eCommerce informa al ERP de las ventas y actualizaciones de stock.

No obstante, durante el desarrollo del proyecto me he encontrado con varias dificultades y problemas de comunicación. Cuando aparecen varias marcas y varias estructuras de identificadores, se crean semejanzas y en muchos casos se da el problema de identificadores idénticos entre marcas. Para poder abordar todos estos problemas, se ha diseñado nuevos identificadores que concatenan marca e identificador.

Una vez realizado el proyecto se ha conseguido un diseño de datos y canal de comunicación lo suficientemente óptimo para llevar a cabo la finalidad del proyecto. Pero como en toda tarea, en esta se podrían llevar a cabo varias mejoras. A parte de la propuesta en el apartado 6 (actualmente en desarrollo), otra propuesta es añadir funcionalidades al ERP para poder crear y eliminar categorías y atributos del eCommerce y además, permitir insertar nuevas descripciones de productos en la página web con formato html.

En conclusión, he conseguido cumplir todos los objetivos propuestos, y actualmente sigo desarrollando mejoras para el proyecto Mountlegend.

## 8. Referencias bibliográficas

- [1] E-commerce (*electronic commerce*)  
[http://es.wikipedia.org/wiki/Comercio\\_electr%C3%B3nico](http://es.wikipedia.org/wiki/Comercio_electr%C3%B3nico) , 2013.
- [2] ERP (enterprise resource planning)  
[http://es.wikipedia.org/wiki/Sistema\\_de\\_planificaci%C3%B3n\\_de\\_recursos\\_em\\_presariales](http://es.wikipedia.org/wiki/Sistema_de_planificaci%C3%B3n_de_recursos_em_presariales) , 2013.
- [3] MountLegend.  
<http://www.mountlegend.com/> , 2012.
- [4] Introducción a los metadatos: Estándares y aplicación. Eva Méndez y José A. Senso. Unidad de Autoformación. SEDIC 2004.  
<http://www.sedic.es/autoformacion/metadatos/tema1.htm> , 2004
- [5] SAGE  
<http://sage.es/> , 2013
- [6] SAP  
<http://www.sap.com/spain/index.epx> , 2013
- [7] OpenERP Open Source Business Applications  
<https://www.openerp.com/> , 2013
- [8] PayPal  
<https://www.paypal.com/es/webapps/mpp/home> , 2013
- [9] Magento an ebay inc company  
<http://www.magentocommerce.com/> , 2013
- [10] PHP  
<http://php.net/> , 2013
- [11] Zend Framework  
<http://framework.zend.com/> , 2013
- [12] osCommerce  
<http://www.oscommerce.com/> , 2013
- [13] PrestaShop  
<http://www.prestashop.com/es/> , 2013
- [14] NEXCESS  
<http://www.nexcess.net/> , 2013