



UNIVERSITAT
ROVIRA I VIRGILI

Departament d'Enginyeria Informàtica i Matemàtiques

DISEÑO Y DESARROLLO DE UNA TIENDA VIRTUAL COMO HERRAMIENTA DE AMPLIACIÓN DE NEGOCIO

TITULACIÓ: Enginyeria Tècnica Informàtica de Gestió

AUTORS: Alejandra Gazulla Gascón.

DIRECTORS: María Ferre Bergada.

DATA: Juny / 2011.

INDICE

1. Objetivos del proyecto.....	5
2. Análisis del proyecto.....	6
2.1 Descripción del sistema deseado.....	6
2.2 Propósito, alcance y objetivos.....	6
3. Herramientas utilizadas.....	6
3.1 Lenguajes de programación.....	7
3.1.1 PHP.....	7
3.1.2 JavaScript.....	7
3.1.3 HTML.....	8
3.2 Tipos de bases de datos.....	8
3.2.1 Oracle.....	8
3.2.2 Microsoft SQL Server.....	9
3.2.3 MySQL.....	9
3.2.4 HeidiSQL.....	9
3.3 Sistemas operativos.....	10
3.3.1 Microsoft Windows.....	10
3.3.2 Linux.....	10
3.4 Servidores.....	10
3.4.1 Apache.....	11
3.5 Dreamweaver.....	11
3.6 Elección y justificación de las soluciones adoptadas.....	11
4. Patrones de diseño.....	12
4.1 Abstract Factory.....	12
4.2 Adapter.....	12
4.3 Bridge (Handle/Body).....	13
4.4 Bulider (Constructor).....	13
4.5 Chain of Responsibility.....	13
4.6 Command (Petición de operación).....	14
4.7 Composite.....	14
4.8 Decorator.....	14
4.9 Facade.....	15
4.10 Factory Method.....	15
4.11 Flyweight.....	15
4.12 Interpreter.....	16
4.13 Iterator.....	16
4.14 Mediator.....	16
4.15 Memento.....	17
4.16 Observer.....	17
4.17 Prototype.....	17
4.18 Proxy.....	17
4.19 Singleton.....	18
4.20 State.....	18
4.21 Strategy.....	18
4.22 Template Method.....	19
4.23 Visitor.....	19
4.24 Modelo vista controlador.....	20
4.24.1 Definición.....	20

4.24.2 Descripción del patrón.....	20
4.24.3 Flujo de control del modelo vista controlador.....	20
4.24.4 Ventajas.....	21
4.24.5 Inconvenientes.....	21
5. Análisis y diseño del software.....	21
5.1 Diagrama de casos de uso.....	22
5.1.1 Casos de uso del invitado.....	22
5.1.2 Casos de uso del usuario y del administrador.....	22
5.1.3 Casos de uso del administrador.....	23
5.2 Descripción de los casos de uso.....	23
5.3 Diagrama de actividades.....	28
6. Bases de datos.....	36
6.1 Diagrama de Chen.....	36
6.2 Descripción de las tablas.....	36
6.2.1 Consideraciones generales.....	36
6.2.2 Detalle de las tablas.....	36
6.2.3 Detalle de las tablas utilizadas para la interfície gráfica.....	39
7. Planificación.....	43
7.1 Introducción.....	43
7.2 Tareas generales.....	45
7.3 Tareas específicas.....	46
7.4 Gestión de riesgos.....	47
7.5 Esfuerzo.....	48
7.6 Planificación desglosada del proyecto.....	49
7.7 Diagrama de PERT.....	51
7.8 Líneas base del proyecto.....	52
8. Manual de usuario.....	52
8.1 Requisitos hardware y software.....	52
8.1.1 Hardware.....	52
8.1.2 Software.....	54
8.2 Manual de instalación.....	54
8.2.1 Instalación de la base de datos.....	54
8.2.2 Instalación del FileZilla.....	54
8.3 Manual de desinstalación.....	55
8.4 Manual de uso de la aplicación.....	55
9. Juegos de pruebas.....	58
9.1 Login usuario.....	58
9.2 Gestión de categorías.....	59
9.3 Contacto.....	62
9.4 Enlaces.....	65
9.5 Menú administración.....	68
9.6 Menú general.....	70
9.7 Menú clientes.....	71
9.8 Formas de pago.....	72
9.9 Formas de transporte.....	77
9.10 Productos.....	80
9.11 Carrito / proceso de compra.....	82

9.12 Stock.....	86
9.13 Pedidos.....	86
10. Bibliografia.....	89
11. Conclusiones.....	89

Objetivos Del Proyecto

Este trabajo se ha realizado como proyecto de final de carrera en Ingeniería Técnica Informática de Gestión (ETIG).

Mi objetivo en el momento de plantearme en realizar un proyecto, era buscar una situación real con alguna necesidad y resolver el problema que se me planteaba. Con esta idea tuve contacto con diferentes pequeñas empresas para ver que problemas me planteaban. Finalmente, después de reunirme con varias de ellas, decidí estudiar y dar una propuesta viable al problema que me planteaba una de ellas. Una de estas empresas, me propuso realizarles una intranet para gestionar sus aplicaciones. Se trataba de una empresa privada de recolección y selección de frutas. Otra empresa me propuso realizar un proyecto para expandir su negocio. Finalmente me decanté por esta última y el resultado de esta propuesta es hoy mi proyecto de final de carrera. Este problema consistía en que la empresa quería expandir su mercado pero no sabía muy bien cómo. Decidimos hacer varias reuniones y estudiar el mercado de hoy en día, de las necesidades que hay cubiertas y necesita cubrir el mercado. Finalmente decidimos hacer una tienda online, ya que la situación geográfica donde está ubicada la empresa no es la más adecuada y además no es muy común este tipo de artesanía.

La empresa se trata de una pequeña empresa, (compuesta de 3-4 empleados) que se dedica a realizar accesorios para el cabello. Estos accesorios se realizan principalmente con pelo natural lazos y telas. Estos productos están distribuidos en varias categorías, como pueden ser diademas, tocados, prendidos, pasadores.. y se realizan modelos para adulto y modelos infantiles. Los productos finales, están realizados 100% a mano. Por tanto, al trabajar con pelo natural, cosa que es muy costosa y poco vista, decidimos hacer una tienda online para expandir las barreras.

Los requisitos que me plantearon fueron los siguientes: querían una tienda online, donde sus productos siempre pertenecerían a una categoría. Lógicamente querían que cada producto tuviera una foto y su descripción. También decidimos las formas de pago, inicialmente serían contra-rembolso y por transferencia y posteriormente, si el negocio funciona correctamente ya añadiremos el pago por tarjeta. En cuanto al diseño querían algo sobrio, no querían estridencias, dado que lo que ellas querían era que destacaran sus productos. Después de varias reuniones, y de comentar algunos aspectos generales, propusimos entre todos una serie de características que poseería la tienda online. Intentamos hacer una lista clara y concisa ya que la persona responsable de la empresa no tiene demasiados conocimientos informáticos. Las características son las siguientes:

1. La página puede ser gestionada por una persona o más, que tendrán el rango de administrador. Se entiende por gestión, la capacidad de poder insertar o eliminar categorías y productos además de poder cambiar el título de la página y la interfaz gráfica de la misma.
2. Los productos deben pertenecer a una categoría y deben tener una fotografía.
3. Los usuarios podrán registrarse para que posteriormente puedan acceder a realizar una compra. Además tener acceso a modificar su contraseña, consultar facturas de compra realizadas en el pasado, agregar una o más direcciones y contactar por la tienda vía email.
4. Los productos que el usuario registrado quiera comprar se irán insertando en un

- carrito , cuya función será almacenar esos productos hasta que finalice la compra.
5. El usuario registrado debe de ser capaz de eliminar productos del carrito en cualquier momento antes de finalizar al compra.
 6. El usuario registrado tiene que poder elegir a donde quiere que se le envíe su pedido.
 7. Habrá una página de confirmación donde se muestre la factura y la dirección de envío antes de que se haga efectiva
 8. Se enviará un email al comprador con la factura y la dirección de envío en el momento en que se haga la compra.
 9. Acceso restringido a varias páginas por rango (no se podrá acceder a ciertas partes si no se tienen permisos para ello)
 10. La página web de la tienda deberá poderse ver desde cualquier sistema con un navegador.

Análisis del Proyecto

2.1 Descripción del sistema deseado

Se pretende realizar una aplicación web que facilite a los dueños de tiendas la posibilidad de crear una tienda web de la manera más sencilla posible, en la que no haga falta muchos conocimientos sobre informática. Para ello se usarán tecnologías orientadas a la creación y gestión de páginas webs, y una base de datos que sea eficiente.

2.2 Propósito, alcance y objetivos

1. **Propósito:** Estimular los negocios de compra-venta en internet, de una forma fácil y sencilla.
2. **Alcance:** El proyecto es aplicable a cualquier negocio en el que se venda cualquier cosa, ya sea un particular o una empresa. La página no posee plugins de ningún tipo porque no podemos esperar que el usuario los tenga. Esto hace que la página pueda ser cargada desde cualquier navegador.
3. **Objetivos:** El proyecto tiene pues como objetivo permitir la creación de tiendas a cualquier usuario, permitiendo, mediante el tratamiento de la información y el uso de interfaces gráficas, un acceso a los datos sencillo, rápido, libre de fallos y seguro.

Otro importante avance que se desea obtener con esta aplicación es el mantenimiento dinámico de toda la información de cada usuario.

Estos dos objetivos sumados al hecho de poder realizar todas estas actividades vía web, hace que la aplicación alcance un nivel de eficiencia máximo

Herramientas Utilizadas

Para la aplicación vamos a establecer una arquitectura que es la forma de seleccionar e interconectar componentes hardware y software según los requerimientos de la aplicación y de los creadores de esta.

3.1 Lenguaje de programación

El lenguaje de programación es un lenguaje que puede ser utilizado para controlar el comportamiento de una máquina. Consiste en un conjunto de reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos.

Un lenguaje de programación permite a un programador especificar sobre qué datos debe operar una computadora, cómo deben ser almacenados y transmitidos y qué acciones debe tomar dependiendo de las circunstancias.

Los procesadores usados en las computadoras son capaces de actuar según lo indican programas en lenguaje fijo llamado lenguaje de máquina. Los programas escritos en cualquier otro lenguaje podrán ser ejecutados de dos formas:

- Mediante un programa que adapta las instrucciones.
- Traduciendo este programa al programa equivalente escrito en lenguaje máquina.

Hoy en día existen numerosos lenguajes de programación, los que yo he utilizado han sido PHP, JavaScript y HTML.

3.1.1 PHP:

Es un lenguaje de programación interpretado. Fue diseñado para la creación de páginas web dinámicas.

Ventajas de PHP:

- Es un lenguaje multiplataforma.
- Es un lenguaje orientado a la creación de páginas web dinámicas y con acceso a la información que se encuentra en la base de datos.

Capacidad de conexión con la mayoría de las bases de datos.

- En su sitio oficial existe mucha documentación sobre el lenguaje.
- Es libre.
- Permite aplicar técnicas de programación orientada a objetos.
- No es necesaria la definición de las variables.
- A partir de PHP5, permite el tratamiento de excepciones.

Inconvenientes:

- Como es un lenguaje que se interpreta en la ejecución, para determinados usos puede ser un inconveniente que el código fuente no pueda estar oculto.

3.1.2 JavaScript

Se trata de un lenguaje de programación interpretado. Es un lenguaje orientado a objetos, débilmente tipado y dinámico.

Características de JavaScript:

- Es simple, no hace falta tener conocimientos de programación para poder hacer un programa en JavaScript.
- Maneja objetos dentro de nuestra página web y sobre ese objeto podemos definir eventos.
- Es dinámico, responde a eventos en tiempo real.

3.1.3 HTML

Se trata de un lenguaje creado para la elaboración de páginas web. Se usa para definir la estructura y el contenido (en forma de texto) y para complementar el texto con objetos, como pueden ser imágenes, formularios, botones...

Características del HTML:

- Es reconocido en todo el mundo.
- Es utilizado únicamente para dar estructura a una página web.

3.2 Tipos de base de datos

Una base de datos es un conjunto de información almacenada en memoria auxiliar que nos permite el acceso directo a esos datos para poder manipularlos. Para acceder a los datos existen sistemas gestores de bases de datos.

Estos sistemas gestores son conocidos como DBMS y son los encargados de manejar los datos y de servir de interfaz entre la base de datos, el usuario y las aplicaciones que utilizan la base de datos. Se compone de tres lenguajes: un lenguaje de definición de datos, un lenguaje de manipulación de datos y un lenguaje de consulta, este último es el SQL.

El propósito de los sistemas gestores es el de definir las estructuras que utilizaremos para almacenar los datos y el manejo de forma clara, ordenada y sencilla de un conjunto de datos. En la actualidad existen muchos gestores de bases de datos. Hay gestores libres y gestores comerciales.

3.2.1 Oracle

Oracle es un sistema gestor de bases de datos relacional. Está fabricado por Oracle Corporation. Ha sido uno de los gestores más utilizados en el mercado. Es uno de los gestores más importantes. Ha añadido soporte SQL para muchas funciones y es capaz de optimizar estas funciones en la base de datos.

Hasta hace poco ha sido el gestor más importante y más extendido en el mercado, pero su coste, que es bastante alto, y la entrada de nuevas competencias, como puede ser Microsoft SQL, y competencias de libre comercio, como puede ser MySQL, han causado que ya no sea el gestor más extendido.

Características:

- Las últimas versiones están certificadas para poder trabajar con Linux.
- Soporta funciones orientadas a objetos.
- Es uno de los gestores más completos.
- Tiene un elevado precio en el mercado, incluso para algunas empresas.
- Es más complejo para usar que otros gestores.
- Su seguridad no es de las mejores.

3.2.2 Microsoft SQL server

Es un sistema de gestión de bases de datos relacionales. Microsoft SQL Server ofrece seguridad y disponibilidad mayores a las aplicaciones de datos empresariales. Ofrece una solución integrada de administración y análisis. Ofrece seguridad, escalabilidad y disponibilidad mayores a las aplicaciones de datos de empresas.

Características:

- Estable
- Seguro
- Posee un potente entorno gráfico de administración.
- Permite trabajar en modo cliente-servidor.
- Permite administrar información de otros servidores de datos.
- Su manejo es fácil.
- No es multiplataforma, es decir, solo está disponible para sistemas operativos creados por Microsoft.

3.2.3 MySQL

Es un sistema gestor de base de datos relacional, multihilo y multiusuario. MySQL es un software libre. Funciona con multiplataformas

Características:

- Posee de múltiples motores de almacenamiento. Esto permite que el usuario escoja la que sea más adecuada para cada tabla que tenemos en la base de datos.
- Es muy seguro. Tiene un sistema de contraseñas y privilegios seguro, que está basado en el encriptado.
- Esta disponible en muchas plataformas y sistemas.
- Soporta mucha cantidad de datos.
- Es un software libre.

3.2.4 HeidiSQL

HeidiSQL es una interfaz gráfica opensource. Gestiona de manera simplificada las bases de datos MySQL locales y remotas. Es gratuita. Nos ofrece muchas funcionalidades para que la gestión de las bases de datos sea una tarea más fácil.

Características:

- Genera exportaciones SQL.
- Sincroniza las tablas entre dos bases de datos.
- Administra los privilegios del usuario.
- Importa los archivos de texto.
- Exporta las tablas de los datos en los formatos CSV, HTML, XML y SQL.
- Inserta archivos ASCII o binarios por lotes en las tablas.
- Escribe consultas con resaltado de sintaxis personalizado y auto completado
- Monitoriza y termina los procesos activos de los clientes.
- Conecta a servidores mediante la línea de comandos.

3.3 Sistema Operativo

Un sistema operativo es un programa que ha sido diseñado para que el usuario y el ordenador se comuniquen. También hace más fácil al usuario gestionar recursos de forma eficaz. El sistema operativo arranca cuando se enciende el ordenador. Gestiona el hardware de los ordenadores desde los niveles más básicos.

Los sistemas operativos realizan las funciones básicas para la gestión del equipo como puede ser gestionar de manera eficiente los recursos del equipo.

3.3.1 Microsoft Windows

Microsoft Windows es un sistema operativo con interfaz gráfica para ordenadores personales creados por Microsoft. Es el sistema operativo más utilizado en la actualidad dada su sencillez. Además casi todas las empresas fabricantes de hardware y software desarrollan sus programas para windows.

Aunque windows es utilizado principalmente para ordenadores personales, también existen algunas versiones para servidores y dispositivos móviles.

Características:

- Es el sistema operativo más utilizado del mundo ya que es el más sencillo de utilizar.
- Casi todos los programas son compatibles con este sistema operativo.
- Su coste es alto ya que necesita pagar una licencia por cada máquina donde se quiera utilizar.
- Se necesitan gran cantidad de parches para conseguir un buen funcionamiento del sistema y un buen nivel de seguridad.

3.3.2 Linux

Linux es un sistema operativo denominado de software libre, ya que su código fuente está disponible a todo el público. Linux se originó en 1992.

Aunque Linux se ha utilizado como sistema operativo en algunos ordenadores personales y dispositivos móviles, su mayor desarrollo ha sido en servidores.

Características:

- Es un sistema operativo gratuito.
- Es un sistema multitarea.
- Es muy estable y se usa en muchos servidores.
- Es más complejo que otros sistemas operativos.
- La instalación y la configuración es más complicada que en windows.
- Muchas aplicaciones no son compatibles con este sistema operativo.

3.4 Servidores

Un servidor es una aplicación que realiza unas tareas en beneficio de los clientes. Son servicios habituales los servicios de archivos, los cuales permiten a los usuarios almacenar y acceder a los archivos de un ordenador. Un ordenador puede realizar a la vez las funciones de cliente y de servidor.

3.4.1 Apache

Apache es un servidor web de HTTP de código abierto. Se utiliza en plataformas Unix, Microsoft Windows y Macintosh.

Ventajas:

- Modular
- Código abierto
- Multi-plataforma
- Extensible
- Popular (fácil de conseguir ayuda / soporte)

3.5 Dreamweaver

Dreamweaver es un programa en forma de estudio que está enfocado en la construcción y edición de sitios y aplicaciones web. Es el programa más utilizado para la creación de aplicaciones y sitios web.

3.6 Elección y justificación de las soluciones adoptadas

Una vez analizadas las múltiples soluciones para la aplicación web, ya podemos determinar qué vamos a utilizar y porqué se va a utilizar.

Tras evaluar los diferentes lenguajes que se pueden utilizar, he elegido que el lenguaje en el que se va a realizar la aplicación va a ser PHP. Además PHP fue creado para el fin de realizar páginas web. También hemos utilizado HTML para realizar los formularios y JavaScript para la validación de formularios.

El gestor escogido para gestionar las bases de datos ha sido HeidiSQL. Se ha elegido este gestor porque está considerado el mejor gestor de bases de datos MySQL. Además este gestor es muy seguro, muy fácil de usar y de comprender por lo que un usuario con escaso conocimiento podrá utilizarlo.

La elección del sistema operativo no debería afectar al rendimiento, puesto que la aplicación ha sido diseñada en PHP para que la portabilidad sea máxima. A pesar de ello, tras comprobar las características de cada uno de estos sistemas operativos he preferido elegir como sistema operativo Microsoft Windows. Se ha escogido este sistema operativo porque básicamente es el más utilizado del mundo, por la facilidad en que los usuarios podrán utilizar la aplicación y porque el desarrollador del proyecto conoce a la perfección este sistema operativo.

En cuanto al servidor web, he decidido contratar uno. Primero instalé el servidor apache pero debido a mis continuos desplazamientos semanales a causa de mi procedencia y el lugar de estudios, decidí contratar un servicio de hosting. De esta manera el servidor es un servidor contratado.

El programa utilizado para realizar la programación ha sido dreamweaver. He decidido utilizar este programa por la facilidad y comodidad de ser la vista del navegador desde el propio programa sin tener que acceder al navegador. A parte de esto dreamweaver uno de los programas para la creación de webs.

Patrones de Diseño

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces.

Un patrón de diseño es una solución a un problema de diseño. Para que una solución sea considerada un patrón debe poseer ciertas características:

- Proporcionar catálogos de elementos reusables en el diseño de sistemas software.
- Evita la reiteración en la búsqueda de soluciones a problemas ya conocidos y solucionados anteriormente.
- Formalizar un vocabulario común entre diseñadores.
- Estandarizar el modo en que se realiza el diseño.
- Facilitar el aprendizaje de las nuevas generaciones de diseñadores condensando conocimiento ya existente.

Hay muchos tipos de patrones de diseño utilizados en la creación de páginas web. A continuación ofrezco un listado con estos patrones, sus características y ventajas e inconvenientes.

4.1 Abstract Factory

Este patrón intenta solucionar el problema de crear diferentes familias de objetos. Está aconsejado cuando se prevé la inclusión de nuevas familias de productos, pero sería contraproducente cuando se añaden nuevos productos o cambian los existentes, ya que esto afectaría a todas las familias creadas.

Ventajas:

- Aísla las clases de implementación.
- Hace fácil el intercambio de familias de productos sin mezclarse, permitiendo configurar un sistema como una de entre varias familias.
- Fomenta la consistencia entre productos

Inconvenientes:

- Soportar nuevas clases de productos es difícil. Porque la interfaz del Abstract Factory arregla el bloque de productos que pueden ser creados. Para soportar nuevas clases se requiere extender la interfaz factory.

4.2 Adapter

El patrón adapter convierte la interfaz de una clase en otra interfaz que el cliente espera. Adapter permite a las clases trabajar juntas, lo que de otra manera no podría hacerlo debido a sus interfaces incompatibles.

Ventajas:

- Permite que un único Adapter trabaje con muchos Adaptees.
- El Adapter también puede agregar funcionalidad a todos los Adaptees de una sola vez.
- Incrementa el rehuso de de clases por del desacoplamiento de la interfaz desde la implementación.

Inconvenientes:

- Una clase adaptadora no funcionará cuando se desea adaptar una clase y todas sus subclases
- Hace difícil sobrescribir el comportamiento de Adaptee.
- Inflexibilidad, puesto que un sólo adaptador no puede trabajar con una clase y sus hijos a la vez.

4.3 Bridge (Handle/body)

El patrón Bridge desacopla una abstracción de su implementación para que ambas puedan cambiar independientemente.

Ventajas

- Desacopla interfaz e implementación, pudiendo configurarse o cambiarse durante la ejecución.
- Favorece la división en niveles de un sistema.
- Ambas jerarquías pueden ampliarse independientemente mediante herencia.
- Oculta a los clientes los detalles de las implementaciones concretas.

Inconvenientes:

- En un entorno de ventanas que puede funcionar en dos sistemas de ventanas diferentes (por ejemplo Windows y Mac). Permite desarrollar aplicaciones que funcionen en los dos sistemas pero la implementación de esta abstracción deberá ser distinta en uno que en otro.

4.4 Builder (Constructor)

Separa la construcción de un objeto complejo de su representación, de modo que el mismo proceso de construcción permite crear representaciones.

Ventajas:

- Reduce el acoplamiento
- Permite variar la representación interna de estructuras compleja, respetando la interfaz común de la clase Builder.
- Se independiza el código de construcción de la representación.
- Permita un mayor control en el proceso de creación del objeto.

Inconvenientes:

- Extremadamente complejo y costoso de implementar
- Incrementa el “heap” de memoria en procesos de construcción
- Lentifica el tiempo de construcción al hacer participar más clases en el proceso de instanciación.

4.5 Chain of Responsibility

Evita acoplar el emisor de una petición a su receptor, al dar a más de un objeto la posibilidad de responder a la petición. Crea una cadena con los objetos receptores y pasa la petición a través de la cadena hasta que esta sea tratada por algún objeto.

Ventajas:

- Reduce el acoplamiento.

- Mayor flexibilidad porque se pueden añadir o modificar la capacidad de tender una petición simplemente haciendo cambios en la cadena de responsabilidades dinámicamente.

Inconvenientes:

- No se garantiza la recepción y satisfacción de una petición.

4.6 Command (Petición de Operación)

Encapsula una petición en un objeto, permitiendo así parametrizar a los clientes con distintas peticiones, encolar o llevar un registro de las peticiones y poder deshacer las operaciones.

Ventajas:

- Incrementa la reutilización de las clases por la disociación de la interfaz de la aplicación.
- Prevé un acoplamiento débil entre los clientes y los subsistemas
- Le permite variar el comportamiento de una clase independiente de su contexto.

Desventajas:

- Necesidad de modificar las conductas objeto de forma dinámica sin afectar a los objetos existentes
- Necesidad de hacer cola las solicitudes de ejecutar en el futuro (como deshacer).

4.7 Composite

Combina objetos en estructuras de árbol para representar jerarquías de parte-todo. Permite que los clientes traten de manera uniforme a los objetos individuales y a los compuestos.

Ventajas:

- Se crea una jerarquía de objetos básicos y de composiciones de estos
- Simplifica el cliente al tratar todos de igual forma.
- Facilita agregar nuevas clases insertándolas en un contenedor compatible.
- Generaliza el diseño.

Inconvenientes:

- La necesidad de proteger a clientes e saber si un objeto es un todo o una parte.

4.8 Decorator

Añade dinámicamente nuevas responsabilidades a un objeto, proporcionando una alternativa flexible a la herencia para extender la funcionalidad.

Ventajas:

- Aporta una mayor flexibilidad que la herencia estática, permitiendo añadir una funcionalidad dos o más veces.
- Evita concentrar en lo alto de la jerarquía clases “guiadas por las responsabilidades”.
- Las nuevas funcionalidades se componen de piezas simples que se crean y se combinan con facilidad, independientemente de los objetos cuyo comportamiento

extienden.

Inconvenientes:

- La transparencia tiene el inconveniente de que un decorador y un componente son indistinguibles, por lo que no se puede confiar en la identidad de los objetos.
- Gran fragmentación, el sistema se llena de gran cantidad de objetos pequeños y se dificulta el aprendizaje de su funcionamiento aunque el que lo domine puede adaptarlo fácilmente.

4.9 Facade

Proporciona una interfaz unificada para un conjunto de interfaces de un subsistema. Define una interfaz de alto nivel que hace que el subsistema sea más fácil de usar.

Ventajas:

- Puede reducir las dependencias de compilación.
- Permite a los clientes de varias habilidades para acceder a la capa adecuada del sistema
- Promueve la estratificación de los sistemas, proporciona bajo acoplamiento entre los clientes y los subsistemas.

Inconvenientes:

- Necesidad de subsistemas de capas
- Necesidad de proporcionar abstracción entre el subsistema y el cliente.

4.10 Factory Method

Define una interfaz para crear un objeto, pero deja que sean las subclases quienes decidan qué clase instanciar. Permite que una clase delegue en sus subclases la creación de objetos.

Ventajas:

- Se unen jerarquías de clase paralelas proporcionando relaciones para la creación de subclase y la extensión.

Inconvenientes:

- No saben de antemano todas las clases que deben ser creadas aplazando la creación de objetos a las clases con el contexto, la hoja que necesita para simular la construcción de objetos virtuales.

4.11 Flyweight

Usa el compartimiento para permitir un gran número de objetos de grano fino de forma eficiente

Ventajas:

- Mejora en la eficiencia, sobretodo en relación con el ahorro en el almacenamiento, que aumenta paralelamente con la compartición de objetos.

Inconvenientes:

- Se introducen costes en tiempo de ejecución asociados a las transferencias, búsquedas y/o computación del estado y su almacenamiento.

4.12 Interpreter

Dado un lenguaje, define una representación de su gramática junto con un intérprete que usa dicha representación para interpretar las sentencias del lenguaje.

Ventajas:

- Facilidad para cambiar o extender la gramática, mediante herencia.
- Facilidad para interpretar la gramática, dado que las implementaciones de las clases nodo del árbol sintáctico son similares, pudiendo usarse para ello generadores automáticos de código.
- Facilidad para introducir nuevas formas de interpretar las expresiones en la gramática.

Inconvenientes:

- Limitación en el tipo de gramáticas para las que sirve esta aproximación: simples y sin grandes necesidades de eficiencia.

4.13 Iterator

Proporciona un modo de acceder secuencialmente a los elementos de un objeto agregado sin exponer su representación interna.

Ventajas:

- Se incrementa la flexibilidad, dado que para permitir nuevas formas de recorrer una estructura basta con modificar el iterador en uso, cambiarlo por otro o definir uno nuevo.
- Se facilitan el paralelismo y la concurrencia, puesto que, como cada iterador tiene consciencia de su estado en cada momento, es posible que dos o más iteradores recorran la misma estructura simultánea solapadamente.

Inconvenientes:

- La necesidad de proteger a los clientes de saber si un objeto es un todo o una parte.

4.14 Mediator

Define un objeto que encapsula como interactúan un conjunto de objetos. Promueve bajo acoplamiento al evitar que los objetos se refieran unos a otros explícitamente, y permite variar la interacción entre ellos de forma independiente.

Ventajas:

- Se limita el uso de la herencia.
- Al reducir el acoplamiento entre los objetos que interactúan es más fácil variar o reutilizar dichos objetos y el mediador, independientemente.
- Se simplifican los protocolos de comunicación entre los objetos, a sustituir las relaciones “muchos a muchos” por relaciones “uno a muchos”, que son más sencillas de entender, mantener y extender.

Inconvenientes:

- Se cambia complejidad en la interacción por complejidad en el mediador, que se puede convertir en el elemento más complejo del conjunto de objetos.

4.15 Memento

Representa y externaliza el estado interno de un objeto sin violar la encapsulación, de forma que éste puede volver a dicho estado más tarde.

Ventajas:

- Se salvaguarda el encapsulamiento.

Inconvenientes:

- Puede resultar problemático en ciertos lenguajes el mantenimiento de dos interfaces distintas.

4.16 Observer

Define una dependencia de uno a muchos entre objetos, de forma que cuando un objeto cambia de estado se notifica y actualizan automáticamente todos los objetos.

Ventajas:

- Dado que el “Asunto” no conoce la clase de “Observadores” que tiene, el acoplamiento que existe es mínimo y abstracto. Por ello, tanto unos como otros pueden variar y evolucionar en diferentes niveles de detalle de forma independiente.
- Se facilita y simplifica el “radiado” de las notificaciones.
- Gran libertad a la hora de dar de alta o baja a los “Observadores”.

Inconvenientes:

- Dado el desconocimiento que unos “Observadores” tienen de los otros, cualquier operación aparentemente inofensiva puede originar una cascada de costosas actualizaciones. Además, sin un protocolo preciso, en el que se especifique lo que cambia, el esfuerzo se multiplica.

4.17 Prototype

Especifica los tipos e objeto a crear por medio de una instancia prototípica, y crear nuevos objeto copiando este prototipo.

Ventajas:

- Aísla clases concretas hace familias de producto de cambio permisos fáciles que añaden y productos de quitar en permisos de tiempo de ejecución.

Inconvenientes:

- Necesita una familia de objetos de producto relacionados de ser usado juntos.

4.18 Proxy

Proporciona un sustituto o representante de otro objeto para controlar el acceso a éste.

Ventajas:

- Mantenimiento mayor y la extensibilidad del sistema.
- Puede reducir las dependencias de compilación.
- Promueve la estratificación de los sistemas.
- Proporciona acoplamiento débil entre los clientes y los subsistemas

- Facilita otra optimización, relacionada con la creación de objetos por demanda.

4.19 Singleton

Garantiza que un clase sólo tenga una instancia, y proporciona un punto de acceso global a ella.

Ventajas:

- Proporciona el control para singularizar o en múltiples casos reducen el espacio de nombres global
- El acceso a la “InstanciaÚnica” está más controlado
- Se reduce el espacio de nombres.
- Permite refinamientos en las operaciones y en la representación, mediante la especialización por herencia de “Solitario”.
- Es fácilmente modificable para permitir más de una instancia y, en general, para controlar el número de las mismas.

Inconvenientes:

- Debe controlar el número de los casos de una clase
- En la versión estática una vez que el/los objetos están instanciados, no se eliminan nunca de memoria. La garbage collector no podrá alcanzarlos ya que están referenciados por la clase y esta no los desreferenciará nunca hasta que el sistema se resetee.
- Si bien el patrón permite la herencia de los objetos singleton, la utilización de la misma debe ser controlada y planificada.
- La versión registro obliga a tener un objeto más en memoria.
- En programas multi-threads su utilización puede ser delicada, si al mismo tiempo dos hilos intentan instanciar un objeto a la vez, solo uno de ellos podría hacerlo.

4.20 State

Permite que un objeto modifique su comportamiento cada vez que cambia su estado interno. Parecerá que cambia la clase el objeto.

Ventajas:

- Dado que el comportamiento específico de cada estado está autocontenido en cada estado concreto, existe una gran flexibilidad para añadir nuevos estados y transiciones, mediante la definición de nuevas subclases
- Se hacen más explícitas las transiciones entre estados que si todo estuviera concentrado en una sola clase

Inconvenientes:

- necesidad de romper muchos de los comportamientos con varias condiciones, la necesidad de modificar las conductas objeto de forma dinámica sin afectar a los objetos existentes

4.21 Strategy

Define una familia de algoritmos, encapsula uno de ellos y los hace intercambiables permite que un algoritmo varíe independientemente de los clientes que lo usan.

Ventajas:

- La herencia puede ayudar a sacar factor común a la funcionalidad de los algoritmos.
- Las familias jerárquicas de estrategias definen algoritmos y comportamientos que enfatizan la reutilización.
- Se eliminan las costosas definiciones de comportamiento multicondicionales
- Se posibilita ofrecer diferentes implementaciones del mismo comportamiento, en función de restricciones como el espacio en memoria o el tiempo de respuesta.

Inconvenientes:

- Los clientes deben tener un cierto conocimiento de cada estrategia, para así poder elegir en cada situación cual es la más apropiada.
- Dado que todas las estrategias comportan una interfaz común, si las diferencias entre ellas es grande, es probable que mucha de la información que se les pasa no sea de utilidad más que a las más complejas
- Puede producirse una gran explosión en el número de objetos del sistema.

4.22 Template Method

Define el esqueleto de un algoritmo en una operación, delegando algunos pasos a subclases de forma que estas pueden redefinir ciertos pasos de un algoritmo sin cambiar la estructura del mismo.

Ventajas:

- Los “MetodosPlantilla” son una técnica fundamental para reutilizar código, especialmente en las librerías de clases, donde son la razón de ser de la “factorización de comportamiento común”.

Inconvenientes:

- Necesidad de flexibilidad en la creación de varios objetos compleja necesidad de muchos algoritmos con una variación.

4.23 Visitor

Representa una operación que debe ser llevada a cabo sobre los elementos de la estructura de un objeto. Permite definir una nueva operación sin cambiar las clases de los elementos con los que opera.

Ventajas:

- Es fácil añadir nuevas operaciones a la “Estructura De Objetos”.
- Se juntan las operaciones relacionadas entre sí, separándose las que nada tienen que ver.
- Se pueden recorrer jerarquías formadas por objetos de distintos tipos (distinta clase padre), mientras que un Iterador no puede hacer esto.
- Se puede ir acumulando el estado a medida que se recorre la estructura, en vez de tener que pasarlo como parámetro o guardarlo en variables globales.

Inconvenientes

- Añadir un nuevo tipo de elemento a la estructura de objetos crea problemas, pues hay que tocar toda la estructura jerárquica de “Visitantes”, lo cual es muy costoso.

- Frecuentemente se fuerza a los “Elementos” a proporcionar operaciones públicas que den acceso a su estado interno, con lo que se pone en peligro el encapsulamiento.

4.24 Modelo Vista Controlador

4.24.1 Definición

El Modelo Vista Controlador es un patrón de diseño de arquitectura software usado en aplicaciones que manejan gran cantidad de datos y transacciones complejas donde se requiere una mejor separación de conceptos para que el desarrollo esté estructurado de una mejor manera. El MCV separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos: el modelo, la vista y el controlador.

4.24.2 Descripción del patrón

- **Modelo:** Esta es la representación específica de la información con la cual el sistema opera. El modelo se limita a lo relativo de la vista y su controlador facilitando las presentaciones visuales complejas. El sistema también puede operar con más datos no relativos a la presentación, haciendo uso integrado de otras lógicas de negocio y de datos afines con el sistema modelado.
- **Vista:** Es la representación del modelo en forma gráfica disponible para la interacción con el usuario. En el caso de una aplicación Web, la vista es una página HTML con contenido dinámico sobre el cuál el usuario puede realizar operaciones.
- **Controlador:** es la capa encargada de manejar y responder las solicitudes del usuario, procesando la información necesario y modificando el modelo en caso e ser necesario.

4.24.3 Flujo de control del modelo vista controlador

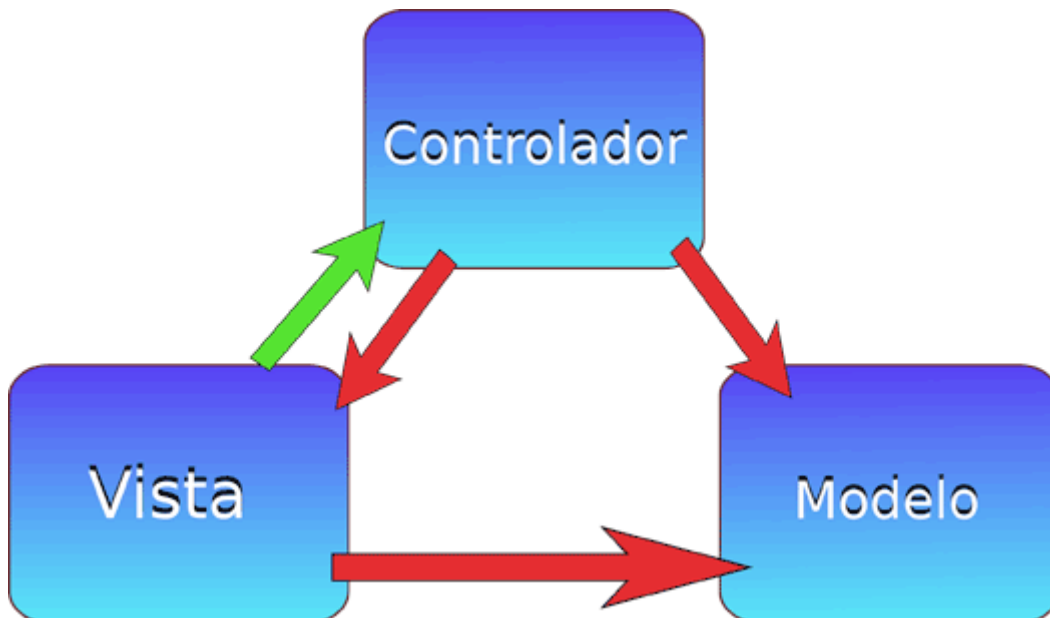


Figura 1: Flujo de control del modelo vista controlador.

1. El usuario interactúa con la interfaz del usuario de alguna forma.
2. El controlador recibe la notificación de la acción solicitada por el usuario y gestiona el evento que llega.
3. El controlador accede al modelo, actualizándolo y posiblemente modificándolo de forma adecuada a la acción solicitada por el usuario.
4. El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene sus datos del modelo para generar la interfaz apropiada para el usuario donde se reflejan los cambios en el modelo. El modelo no debe tener conocimiento directo sobre la vista.
5. La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente.

4.24.4 Ventajas

Las principales ventajas de utilizar el patrón modelo vista controlador (MCV) son las siguientes:

- La separación del modelo de la vista, es decir, separar los datos de la representación visual de la misma.
- Es mucho más sencillo agregar múltiples representaciones de los mismos datos o información.
- Facilita agregar nuevos tipos de datos según sea requerido por la aplicación ya que son independientes del funcionamiento de las otras capas.
- Crea independencia de funcionamiento.
- Facilita el mantenimiento en caso de errores.
- Ofrece maneras más sencillas para probar el correcto funcionamiento del sistema.
- Permite el escalamiento de la aplicación en caso de ser requerido

4.24.5 Inconvenientes

Los principales inconvenientes de utilizar el patrón modelo vista controlador (MCV) son las siguientes:

- La separación de conceptos en capas agrega complejidad al sistema.
- La cantidad de archivos a mantener y desarrollar se incrementa considerablemente.
- La curva de aprendizaje del patrón de diseño es más alta que usando otros modelos más sencillos

Análisis y Diseño del Software

En nuestra tienda podemos tener varios tipos de usuarios, concretamente tres. Los tipos de usuarios dependerá de los privilegios que tenga un usuario respecto de otro en las acciones de la tienda online.

Los tipos de actores serán:

- Invitado: es la persona que entra a nuestra tienda sin estar registrado.
- Usuario: es la persona que está registrada en nuestra tienda y en este momento está logueado. Este usuario podrá realizar más acciones que el invitado.
- Administrador: es la persona que administra la tienda. En este caso yo. El administrador tiene control total sobre la tienda, es decir, puede cambiar la

disposición de la tienda, añadir o eliminar acciones, cambiar la interficie...

5.1 Diagrama de Casos de Uso

5.1.1 Casos de uso del invitado

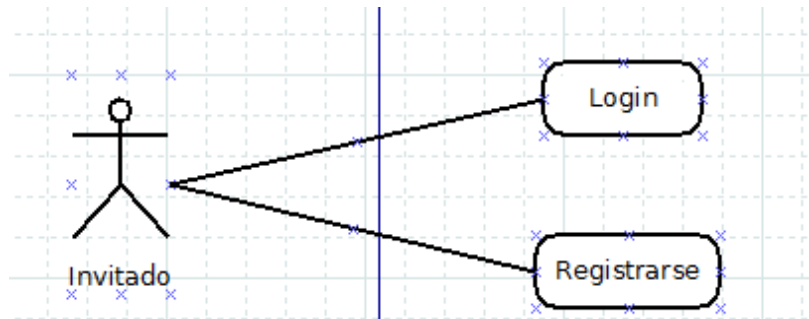


Figura 2: Caso de uso del invitado

5.1.2 Casos de uso del usuario y administrador

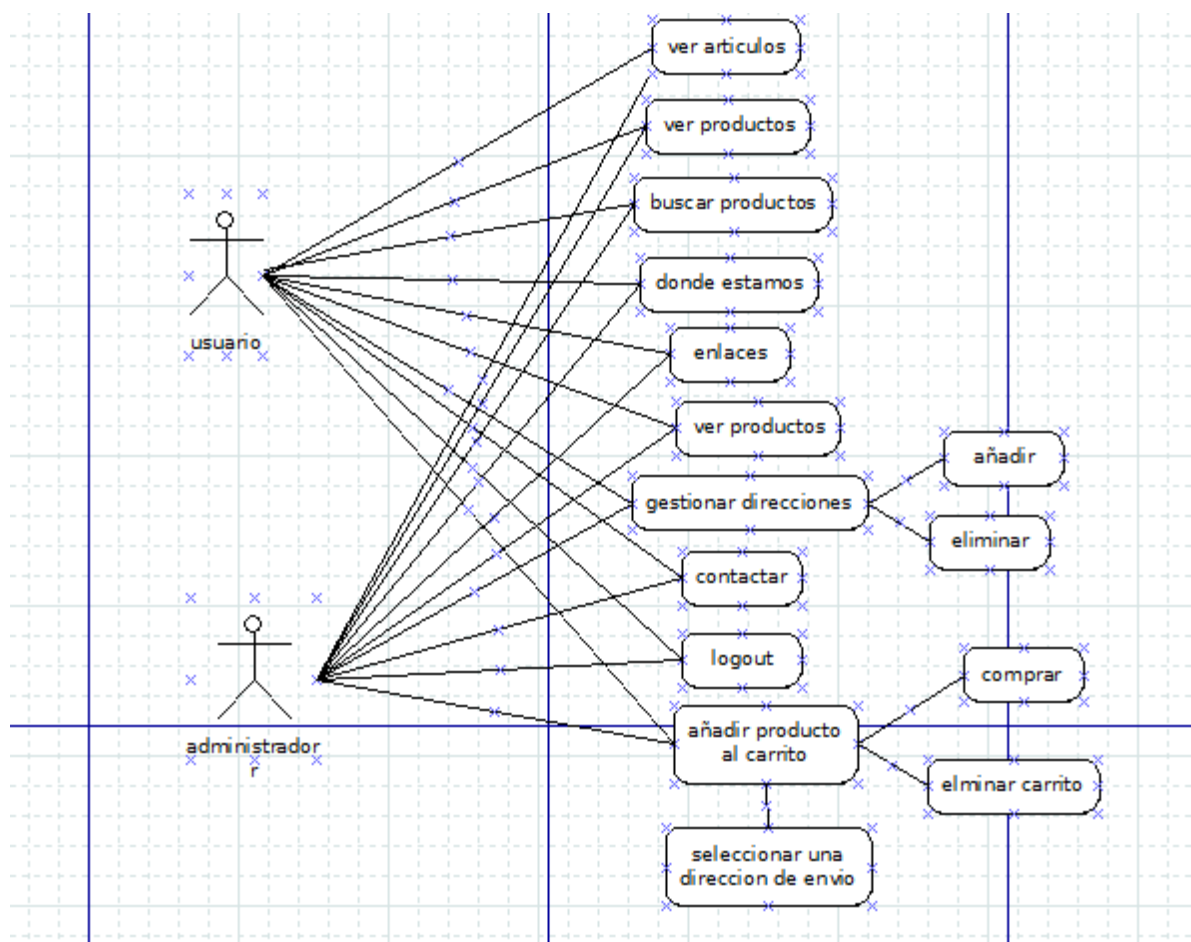


Figura 3: Caso de uso del usuario y del administrador

5.1.3 Casos de uso del administrador

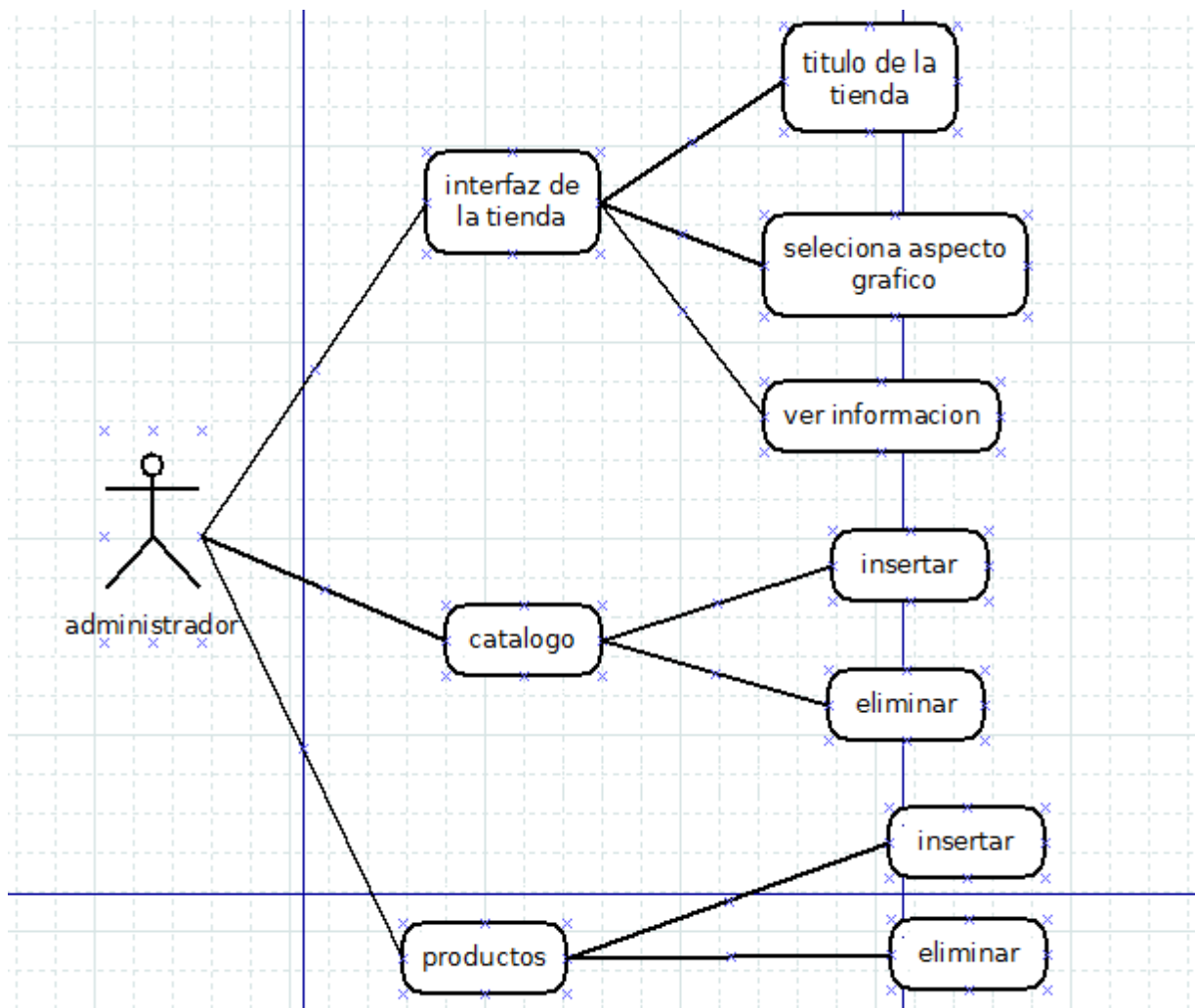


Figura 4: Caso de uso del administrador

5.2 Descripción de Casos de Uso

Descripción de actores:

- Invitado: son los usuarios que no se han registrado.
- Usuario: son los usuarios que se han registrado.
- Administrador: es el encargado de gestionar todo el mantenimiento de la tienda.

Productos:

- Descripción: el actor podrá ver los productos de una determinada categoría.
- Actores: invitado, usuario y administrador.
- Precondición: ninguna.
- Postcondición: se mostrarán los productos por categorías.
- Flujo de eventos:
 1. Flujo básico: el actor debe hacer click en la categoría.

Novedades:

- Descripción: el actor podrá ver un listado de los últimos productos insertados en la base de datos.
- Actores: invitado, usuario y administrador.
- Precondición: ninguna.
- Postcondición: mostrará tantos productos como el administrador haya definido en el panel de administración.
- Flujo de eventos:
 1. Flujo básico: el actor deberá hacer click en 'novedades'

Buscar productos

- Descripción: el actor podrá buscar productos de la tienda que contengan una cadena de caracteres predefinida.
- Actores: invitado, usuario y administrador.
- Precondición: ninguna.
- Postcondición: el sistema mostrará los productos que coincidan con el nombre buscado.
- Flujo de eventos:
 1. Flujo básico: el actor deberá introducir un nombre y pulsar el botón 'buscar'.
 - a) Flujo alternativo: si no se introduce ningún nombre en el campo, aparecerán todos los productos registrados.

Donde estamos:

- Descripción: el actor podrá ver la ubicación de la tienda.
- Actores: invitado, usuario y administrador.
- Precondición: ninguna.
- Postcondición: el sistema mostrará un mapa con la ubicación y la dirección de las tienda.
- Flujo de eventos:
 1. Flujo básico: el actor deberá hacer click en 'donde estamos'.

Enlaces:

- Descripción: el actor podrá ver un listado con enlaces a otras páginas.
- Actores: invitado, usuario y administrador.
- Precondición: ninguna.
- Postcondición: el sistema mostrará un listado con los enlaces que tenemos en nuestra tienda.
- Flujo de eventos:
 1. Flujo básico: el usuario deberá hacer click en 'enlaces'
 - a) Flujo alternativo: cuando el actor linka sobre un enlace, este enlace se abrirá en una nueva pestaña.

Añadir al carrito

- Descripción: el actor podrá ir añadiendo productos a su carrito de la compra con el fin de realizar la compra en cualquier momento.
- Actores: usuario y administrador
- Postcondición: estar logueado.
- Precondición: se agregará al menú del carrito el producto señalado.

- Flujo de eventos:
 1. Flujo básico: el actor deberá introducir una cantidad y darle al botón 'añadir al carrito'.

Registrarse

- Descripción: el actor podrá acceder a los listados de los productos, a loguearse o registrarse.
- Actores: usuario y administrador.
- Precondición: debe rellenar los campos correctamente y si el email ya está registrado se dará un aviso de que no está disponible.
- Postcondición: nada mas registrarse se logueará automáticamente.
- Flujo de eventos:
 1. Flujo básico: el usuario rellenará los campos y se registrará en la base de datos.
 - a) Punto abierto: si el usuario rellena el campo email con un nombre ya utilizado, aparecerá un aviso que indicará que no te puedes registrar con esa nueva cuenta de correo.

Login:

- Descripción: el actor podrá loguearse en la tienda y acceder a todas las funciones a las que da permiso.
- Actores: usuario y administrador.
- Precondición: el actor ya se ha registrado previamente.
- Postcondición: el actor habrá accedido a los productos de la tienda.
- Flujo de eventos:
 1. Flujo básico: el actor rellenará los campos con el email y la contraseña con la que se registró, y accederá al sistema.
 - a) Primer punto abierto: si el actor rellenase los campos con datos incorrectos aparecerá un aviso informando que los datos son erróneos.

Datos personales:

- Descripción: el actor podrá cambiar su propia contraseña.
- Actores: usuario y administrador.
- Precondición: El actor debe de estar logueado.
- Postcondición: El sistema cambiará la contraseña del actor.
- Flujo de eventos:
 1. el actor deberá introducir su antigua contraseña y la nueva contraseña dos veces.
 - a) Primer punto abierto: si el actor introduce datos incorrectos aparecerá un aviso informando que los datos son erróneos.

Salir:

- Descripción: el actor podrá cerrar la sesión del usuario para finalizar la sesión.
- Actores: usuario y administrador.
- Precondición: debe haberse logueado anteriormente.
- Poscondición: el sistema sacará al actor de la tienda virtual.
- Flujo básico de eventos:

1. Flujo básico: el actor deberá pulsar el botón 'salir'.

Ver pedidos:

- Descripción: el actor tendrá acceso a su historial de compras.
- Actores: usuario y administrador.
- Precondición: el actor debe haberse logueado.
- Postcondición: ninguna.
- Flujo de eventos:
 1. flujo básico: pulsando en 'pedidos' el actor podrá ver todas las compras que ha hecho.

Contacto:

- Descripción: el actor podrá mandar un correo electrónico a la dirección de email asociada a la web.
- Actores: usuario y administrador.
- Precondición: el actor debe haberse logueado.
- Postcondición: Ninguna
- Flujo de eventos:
 1. Flujo básico: el actor accederá haciendo click en 'contacto'.
 - a) Primer flujo alternativo: escribirá el nombre, el email, el teléfono y la consulta y luego hará click en enviar.

Cesta (paso 1):

- Descripción: el actor podrá continuar el proceso de compra, eliminar algún producto del carrito y seleccionar la dirección de envío.
- Actores: usuario y administrador.
- Precondición: el actor deberá haberse logueado y tener un producto o más en el carrito.
- Postcondición: el actor deberá seleccionar la dirección de envío.
- Flujo de eventos:
 1. Flujo básico: el actor podrá ver todos los productos del carrito.
 - a) Primer flujo alternativo: el actor podrá eliminar los productos seleccionados dando al botón eliminar.
 - b) Segundo flujo alternativo: el actor podrá añadir o quitar unidades de cada producto que hay en el carrito.

Cesta (paso 2):

- Descripción: el actor que seleccionar la dirección de envío.
- Actores: usuario y administrador.
- Precondición: el actor deberá estar logueado y haber sido redireccionado desde el paso anterior.
- Postcondición: el actor deberá seleccionar el tipo de transporte.
- Flujo de eventos:
 1. Flujo básico: el actor podrá introducir una dirección alternativa de facturación.

Cesta (paso 3):

- Descripción: el actor deberá elegir el tipo de transporte.
- Actores: usuario y administrador.
- Precondición: el actor deberá estar logueado y haber sido redireccionado desde el paso anterior.
- Postcondición: el actor deberá seleccionar la forma de pago.
- Flujo de eventos:
 1. Flujo básico: El actor deberá seleccionar un tipo de transporte y pulsar el botón siguiente.

Cesta (paso 4):

- Descripción: el actor deberá elegir la forma de pago
- Actores: usuario y administrador.
- Precondición: el actor deberá estar logueado y haber sido redireccionado desde el paso anterior.
- Postcondición: el sistema nos mostrará el resumen de la compra.
- Flujo de eventos:
 1. Flujo básico: el actor deberá seleccionar una forma de pago y pulsar el botón siguiente.

Cesta (paso 5):

- Descripción: El actor tendrá que confirmar la compra.
- Actores: usuario y administrador.
- Precondición: el actor deberá estar logueado y haber sido redireccionado desde el paso anterior.
- Postcondición: el sistema enviará un email con la factura a la dirección asociada al proyecto y a la del usuario.
- Flujo de eventos:
 1. Flujo básico: se le mostrará al actor un resumen de la compra y la dirección de envío. El actor deberá confirmar.

Interfaz de la tienda:

- Descripción: el actor podrá cambiar una serie de opciones generales de la tienda.
- Actores: administrador.
- Precondición: el actor debe haberse logueado y ser administrador.
- Postcondición: el sistema mostrará la página actualizada.
- Flujo de eventos:
 1. Flujo básico: el actor podrá cambiar el título de su página, elegir el estilo gráfico e información sobre la tienda.

Modificar catálogo:

- Descripción: el actor podrá agregar o eliminar categorías.
- Actores: administrador.
- Precondición: el actor debe haberse logueado y ser administrador.
- Postcondición: el sistema mostrará la misma página con la información actualizada.
- Flujo de eventos:
 1. Flujo básico: el actor dispone de un campo de texto para insertar una categoría y una lista de categorías ya insertadas para poder eliminarlas.

Modificar productos:

- Descripción: insertar productos a una categoría.
- Actores: administrador.
- Precondición: el actor debe haberse logueado y ser administrador.
- Postcondición: el sistema mostrará la misma página con la información actualizada.
- Flujo de eventos:
 1. Flujo básico: El actor podrá seleccionar la categoría a la que pertenece el producto que se desee ingresar. También podrá eliminar productos de una lista.
 - a) Primer flujo alternativo: se selecciona una categoría de una lista y se insertan los datos del producto y su fotografía, si se desea, se aprieta el botón para insertar.
 - b) Segundo flujo alternativo: se selecciona un elemento de la lista y se pulsa el botón de eliminar.

5.3 Diagrama de Actividades

Catálogo:

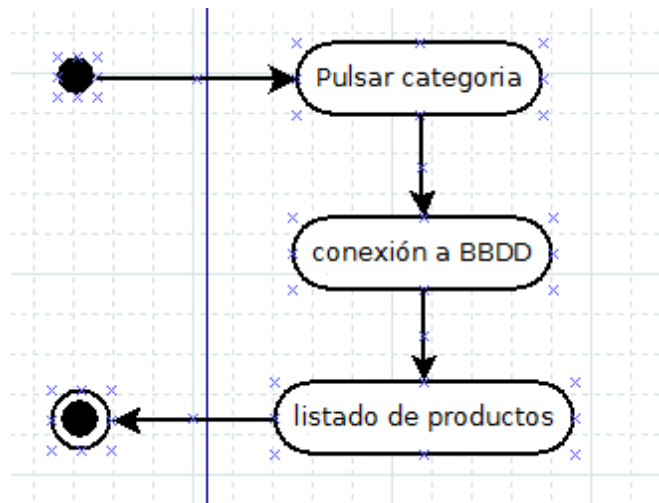


Figura 5: Diagrama de actividades del catálogo

Novedades:

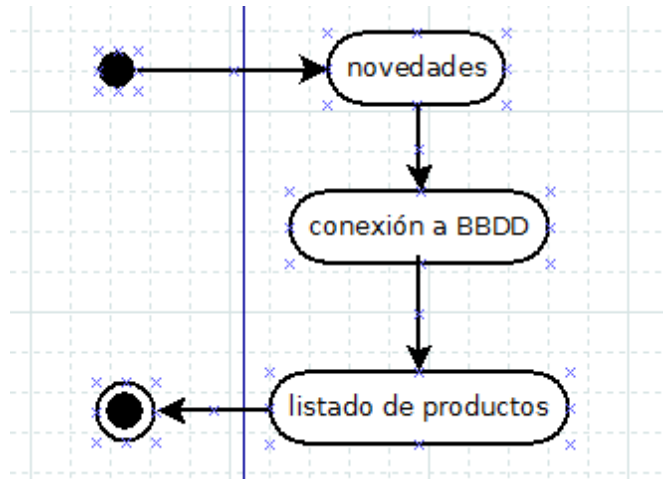


Figura 6: Diagrama de actividades de las novedades

Donde estamos:

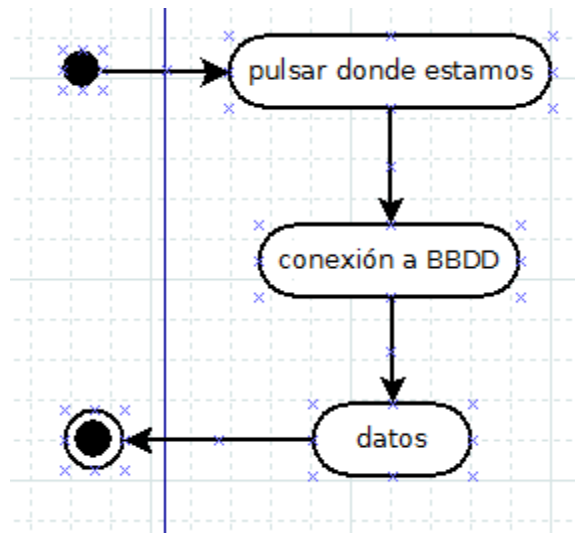


Figura 7: diagrama de actividades de donde estamos

Enlaces:

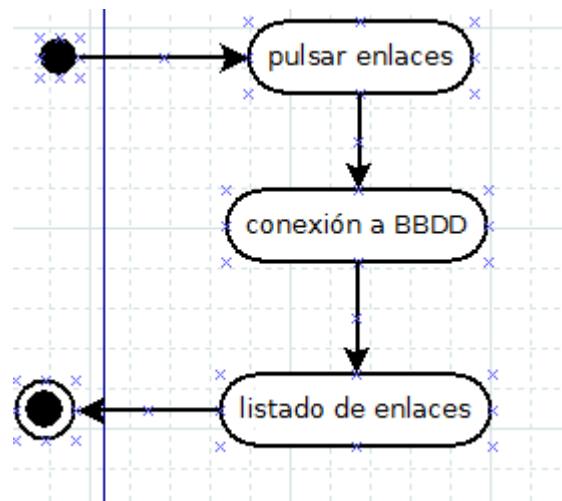


Figura 8: Diagrama de actividades de enlaces

Buscar productos:

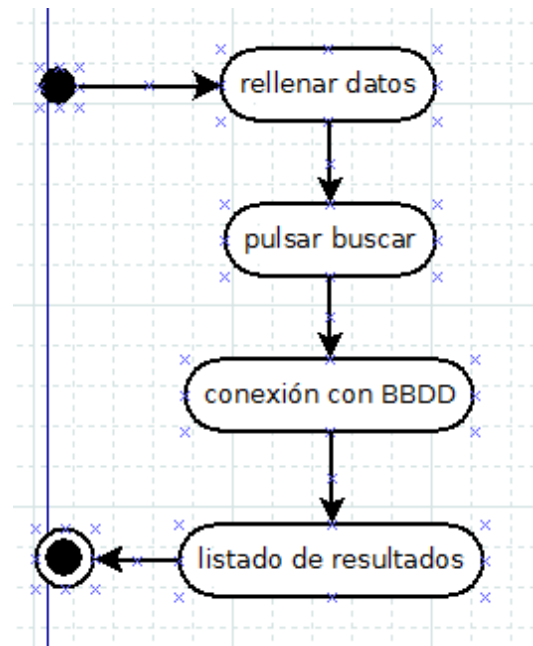


Figura 9: Diagrama de actividades e buscar productos

Añadir al carrito:

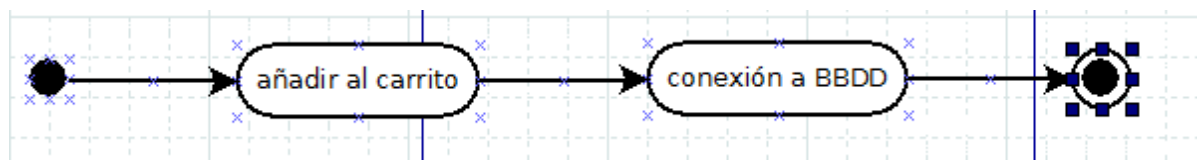


Figura 10: Diagrama de actividades de añadir al carrito

Registrarse:

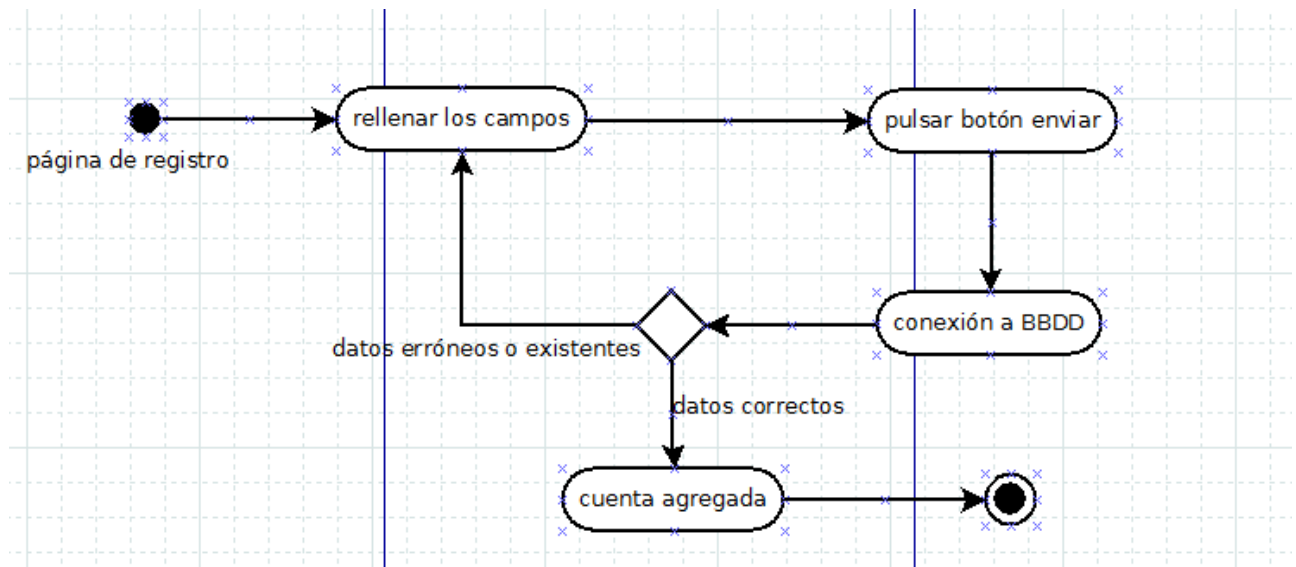


Figura 11: Diagrama de actividades de registrarse

Login:

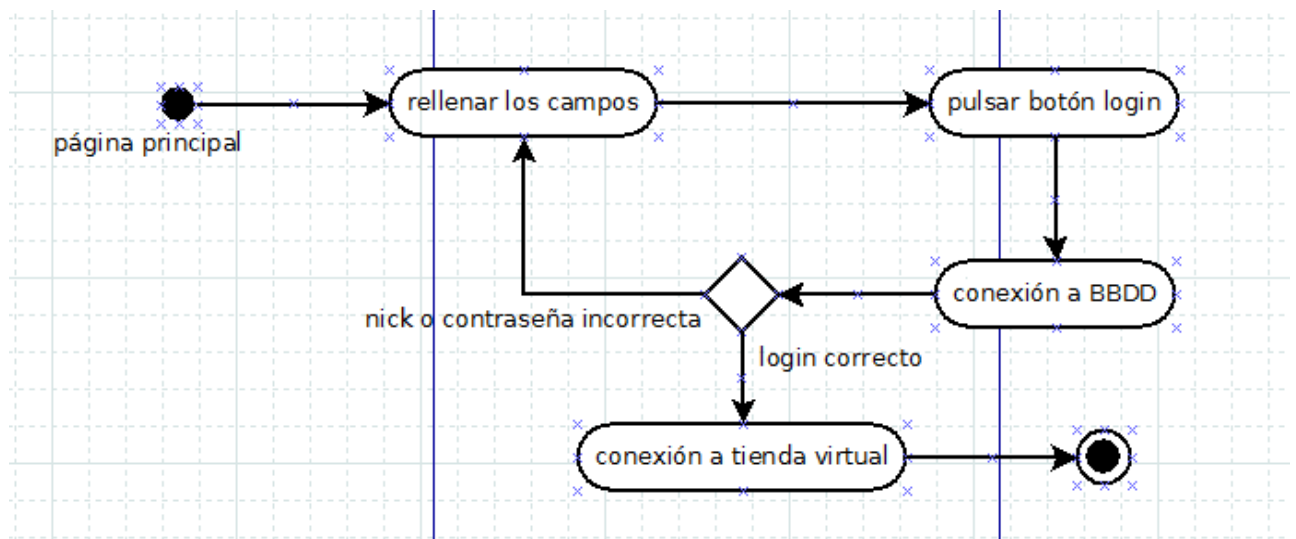


Figura 12: Diagrama de actividades de login

Datos personales:

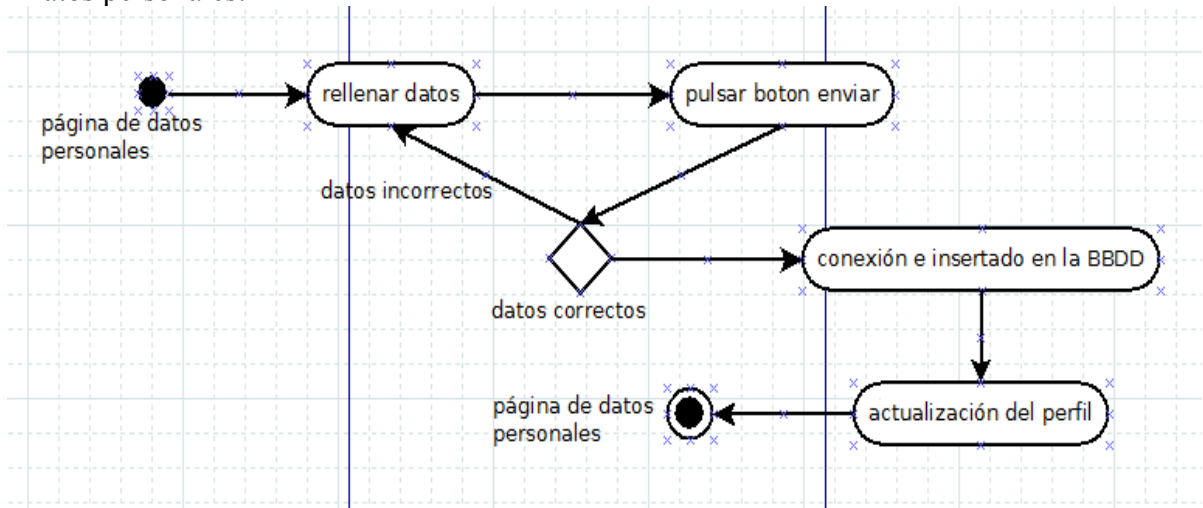


Figura 13: Diagrama de actividades de datos personales

Salir:

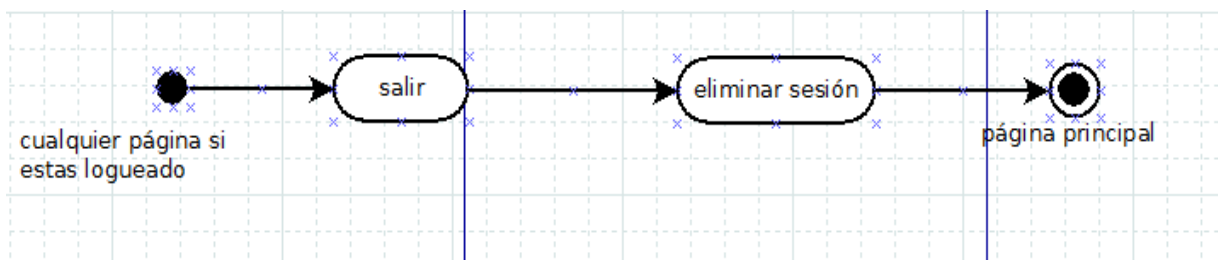


Figura 14: Diagrama de actividades de salir

mostrar pedidos:

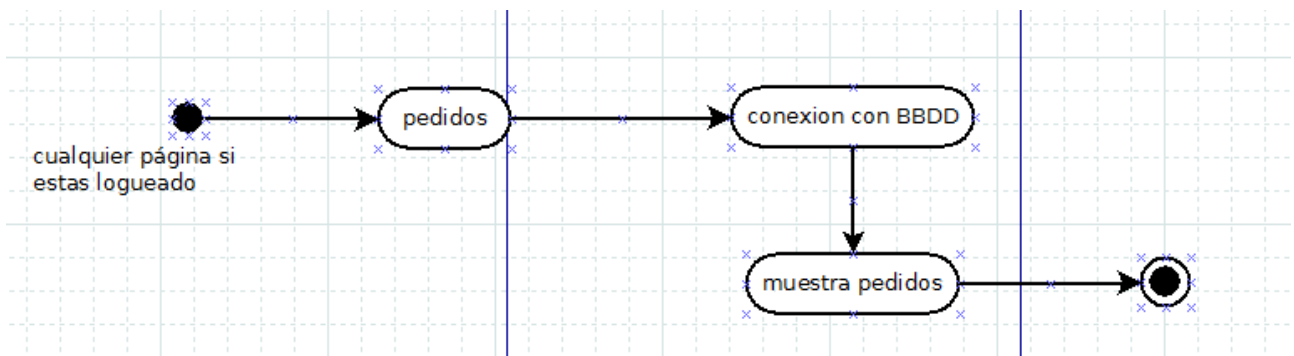


Figura 15: Diagrama de actividades de mostrar pedidos

Contacto:

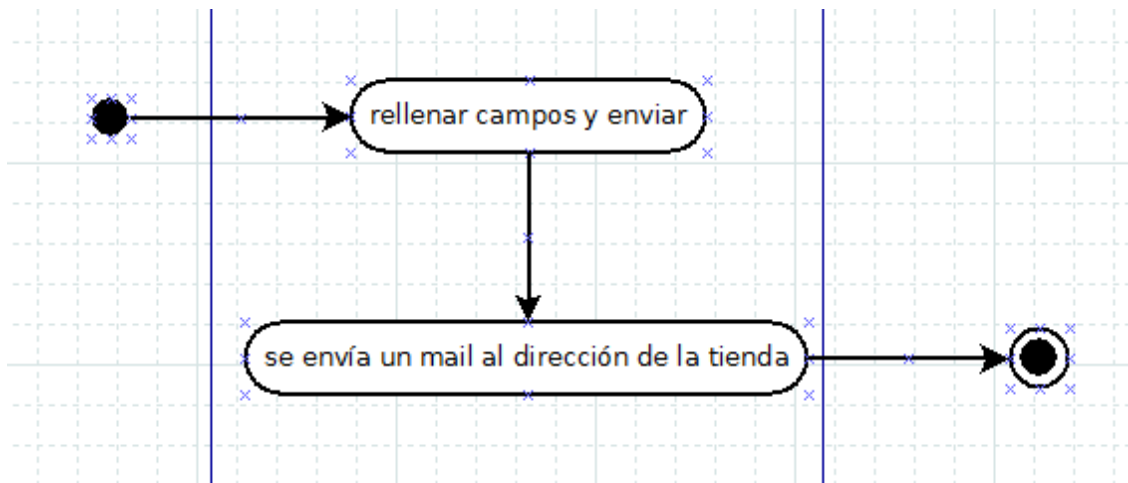


Figura 16: Diagrama de actividades de contacto

Cesta (paso 1):

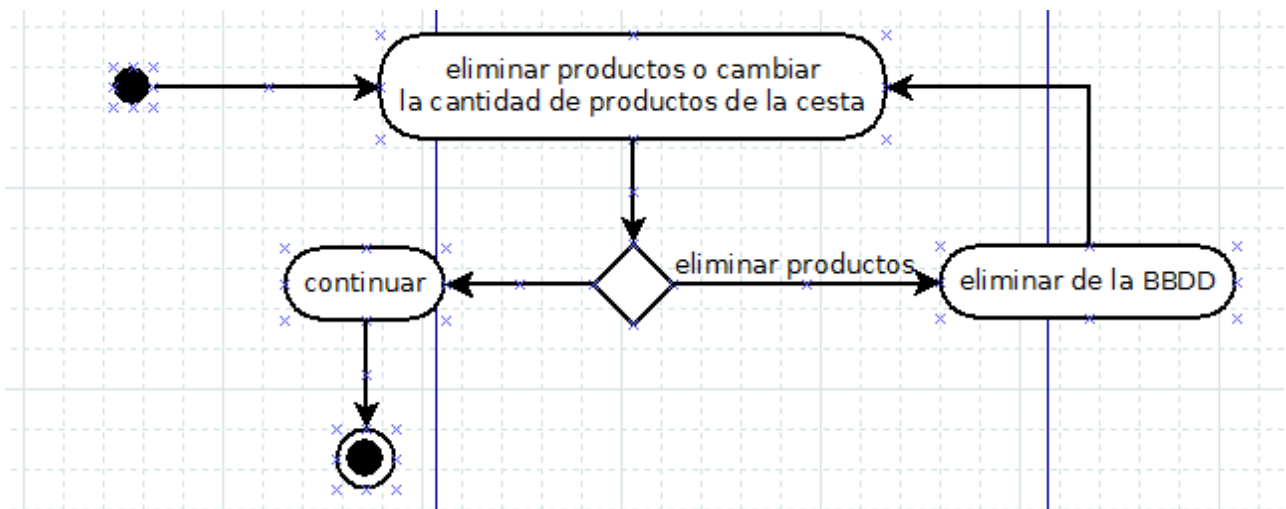


Figura 17: Diagrama de actividades de la cesta (paso 1)

Cesta (paso 2):

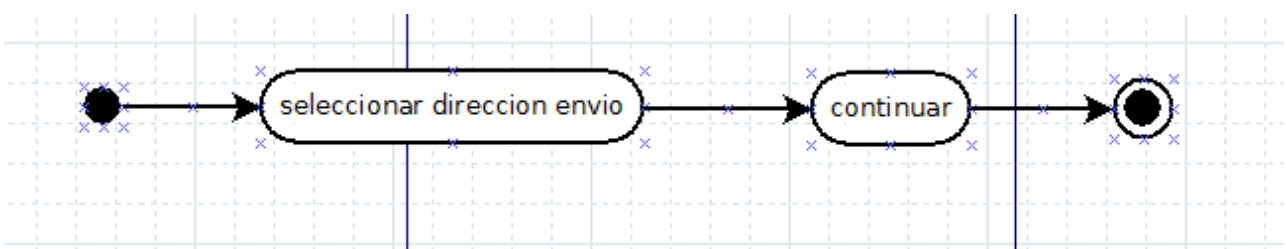


Figura 18: Diagrama de actividades de la cesta (paso 2)

Cesta (paso 3):

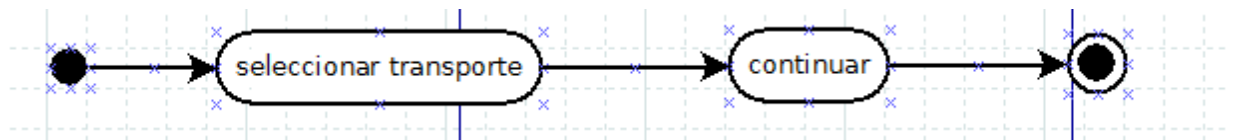


Figura 19: Diagrama de actividades de la cesta (paso 3)

Cesta (paso 4):

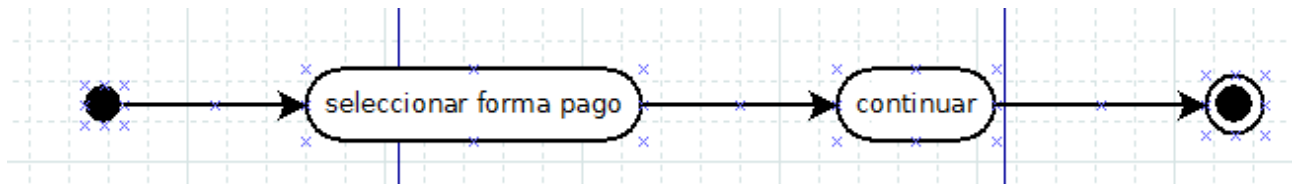


Figura 20: Diagrama de actividades de la cesta (paso 4)

Cesta (paso 5):

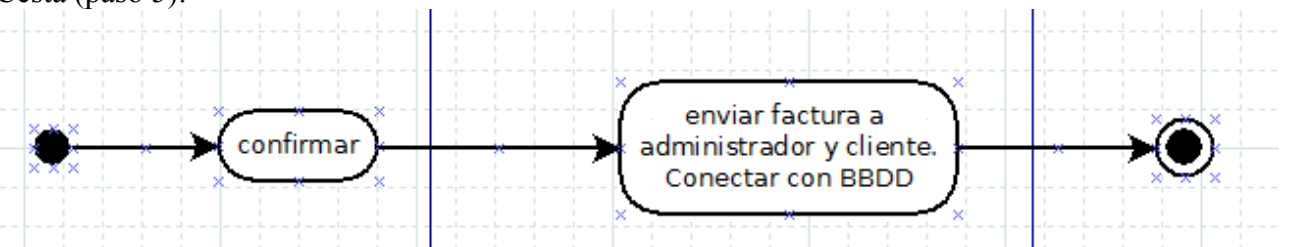


Figura 21: Diagrama de actividades de la cesta (paso 5)

Interfaz de la tienda:

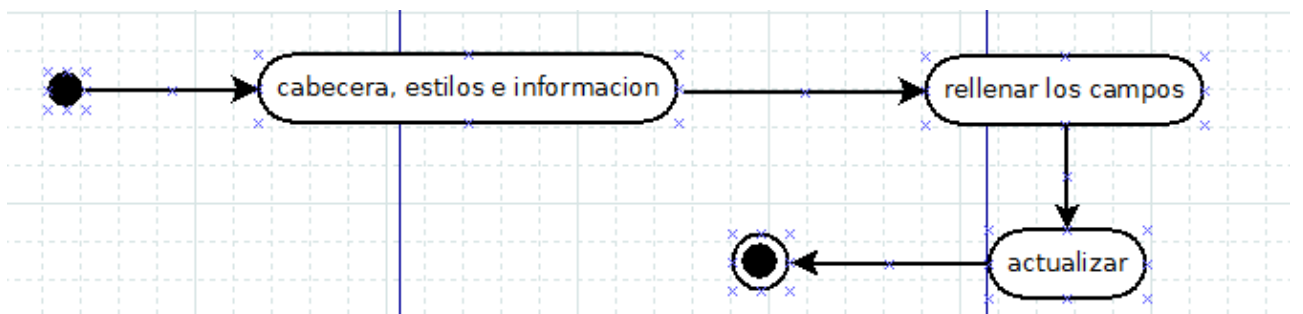


Figura 22: Diagrama de actividades de la interfaz de la tienda

Modificar catálogo:

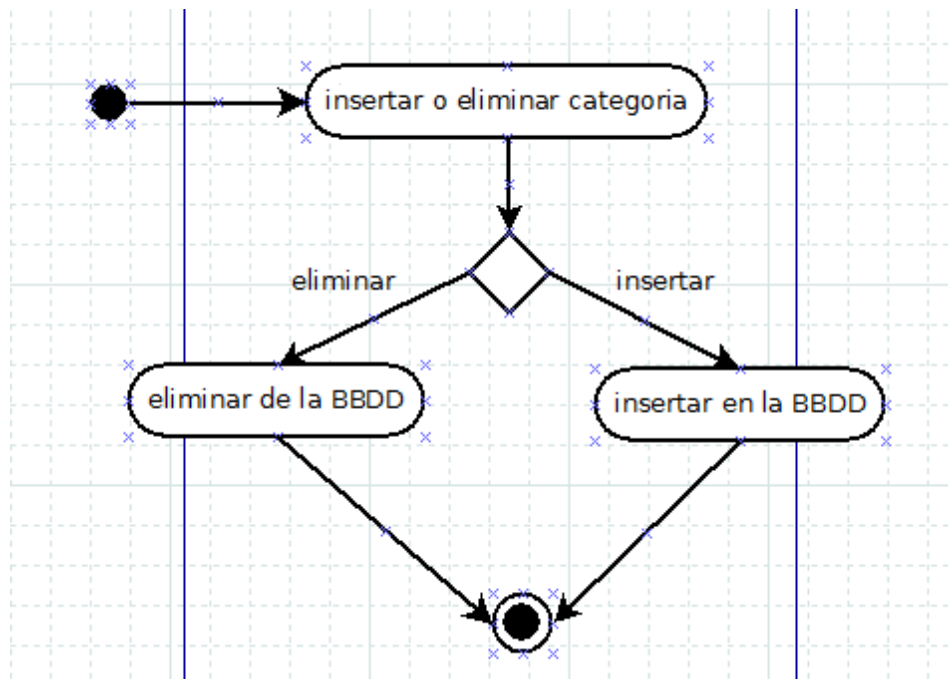


Figura 23: diagrama de actividades de modificar catálogo

Modificar productos:

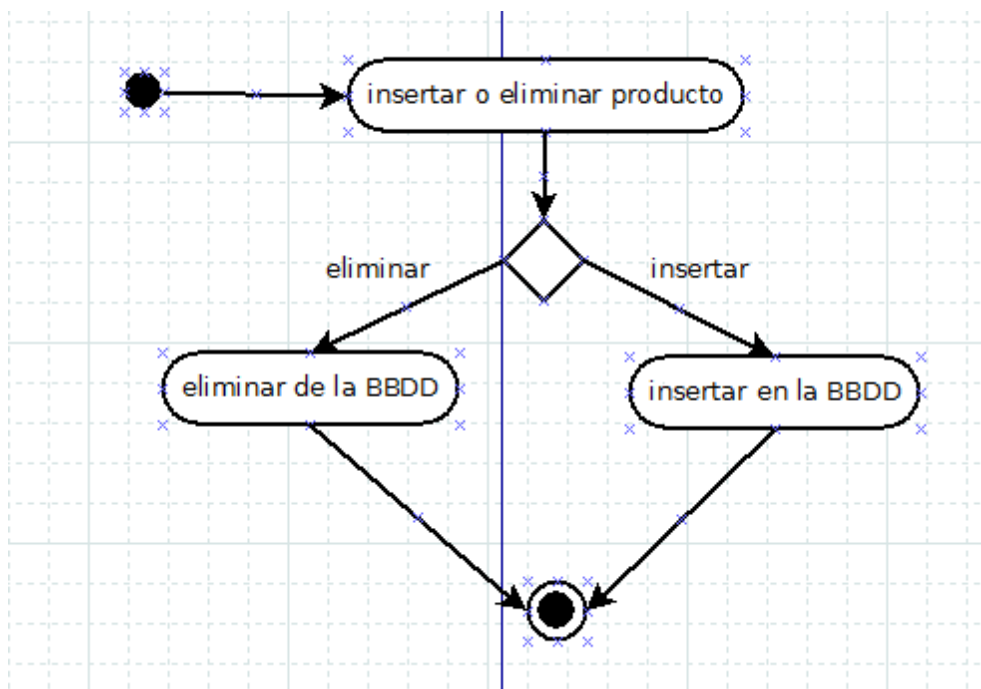


Figura 24: Diagrama de actividades de modificar productos

Bases de Datos

6.1 Diagrama de Chen

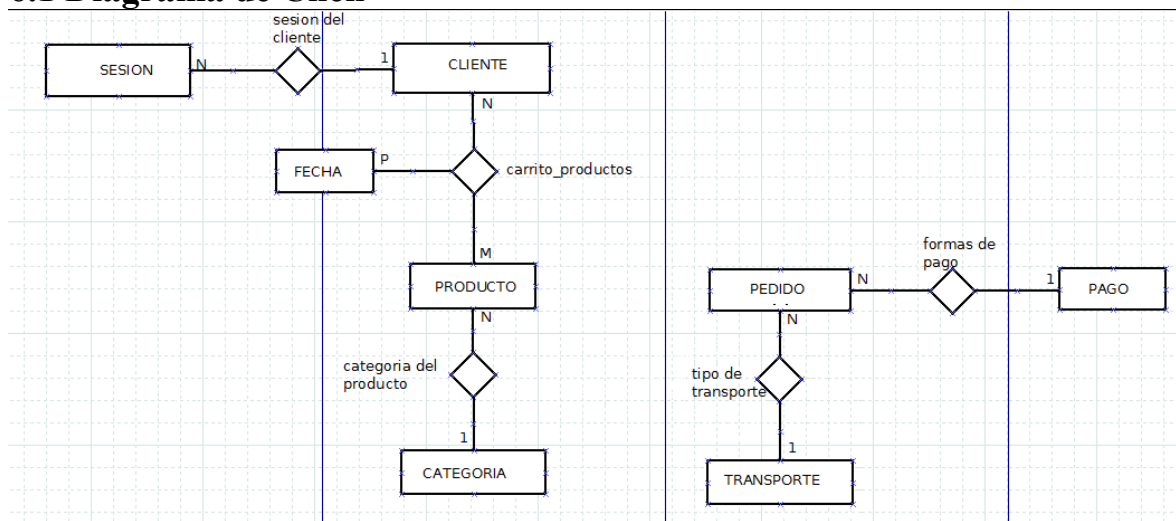


Figura 25: Diagrama de Chen

6.2 Descripción de las Tablas

6.2.1 Consideraciones generales

El campo activo lo he creado por seguridad. Algunos registros no se deben eliminar completamente y este campo impide que se muestren por la aplicación. Esto puede ser útil por ejemplo si hay un error y se elimina un pedido que no se debería eliminar, así se podría recuperar sin que cause mayor contratiempo.

También he creado una serie de tablas en la base de datos para poder administrar la interfaz de la tienda. Estas tablas pueden ser el menú de administración, el menú general, el menú de clientes, entre otras tablas. Así se facilita el trabajo de modificar al interfaz o añadir o eliminar secciones de la parte pública sin tener que modificar el código de la aplicación.

6.2.2 Detalle de las tablas

1. Carrito: en esta tabla tenemos los campos necesarios para guardar el carrito. Ponemos el campo de pendiente, porque es el campo que podemos modificar nosotros desde la parte de administración. Además de esto guardamos la sesión del cliente, la fecha en la que se ha realizado y el identificador del cliente.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	sesion	VARCHAR	50		<input checked="" type="checkbox"/>		"
3	id_cliente	INT	10	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0
4	fecha_alta	DATETIME			<input checked="" type="checkbox"/>		NULL
5	pendiente	INT	11	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	1
6	activo	INT	10	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	1

Figura 26: Tabla de carrito

2. Carrito_productos: en esta tabla tenemos los campos necesarios para guardar la cantidad de cada producto que se añade al carrito. Así pues, en esta tabla tenemos el identificador del carrito, el identificador del producto, la cantidad y la fecha en la que se ha realizado la compra.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	id_carrito	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
3	id_producto	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
4	cantidad	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
5	fecha_alta	DATETIME			<input type="checkbox"/>		No default
6	activo	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 27: Tabla de los productos del carrito

3. Categoría: en esta tabla tenemos todos los campos necesarios para guardar las categorías. Podemos apreciar, que el nombre de una categoría no puede ser nulo, en cambio su descripción si que lo puede ser.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	nombre	VARCHAR	80		<input type="checkbox"/>		No default
3	descripcion	TEXT			<input checked="" type="checkbox"/>		No default
4	activo	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 28: Tabla de las categorías

4. Cliente: En esta tabla tenemos todos los campos necesarios para guardar al información de un cliente. Podemos ver que todos los campos son obligatorios excepto los datos correspondientes a la dirección de facturación y la fecha de nacimiento.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO INCREMENT
2	nombre	VARCHAR	80		<input type="checkbox"/>		No default
3	apellidos	VARCHAR	80		<input type="checkbox"/>		No default
4	dni	VARCHAR	9		<input type="checkbox"/>		No default
5	env_direccion	VARCHAR	80		<input type="checkbox"/>		No default
6	fac_direccion	VARCHAR	80		<input checked="" type="checkbox"/>		NULL
7	env_localidad	VARCHAR	50		<input type="checkbox"/>		No default
8	fac_localidad	VARCHAR	50		<input checked="" type="checkbox"/>		NULL
9	env_provincia	VARCHAR	80		<input type="checkbox"/>		No default
10	fac_provincia	VARCHAR	80		<input checked="" type="checkbox"/>		NULL
11	env_codigo_postal	VARCHAR	5		<input type="checkbox"/>		No default
12	fac_codigo_postal	VARCHAR	5		<input checked="" type="checkbox"/>		NULL
13	fecha_nac	DATE			<input checked="" type="checkbox"/>		NULL
14	email	VARCHAR	40		<input type="checkbox"/>		No default
15	usuario	VARCHAR	50		<input type="checkbox"/>		No default
16	password	VARCHAR	12		<input type="checkbox"/>		No default
17	telefono	VARCHAR	9		<input type="checkbox"/>		No default
18	activo	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 29: Tabla del cliente

5. Formas de pago: en esta tabla guardamos los campos correspondientes a los tipos de las formas de pago. He añadido un campo comisión en el que indicamos la comisión que dependiendo de la forma de pago se incrementa en el importe total.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO INCREMENT
2	nombre	VARCHAR	50		<input type="checkbox"/>		No default
3	descripcion	TEXT			<input checked="" type="checkbox"/>		No default
4	comision	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default
5	activo	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 30: Tabla de formas de pago

6. Pedidos: En esta tabla guardamos todos los datos correspondientes al pedido. Estos datos se resumen a la dirección de envío, a la dirección de facturación (que puede ser nula), al identificador del carrito, al identificador del transporte, al identificador de la forma de pago y al estado del carrito.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	id_carrito	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
3	fac_direccion	VARCHAR	50		<input checked="" type="checkbox"/>		NULL
4	env_direccion	VARCHAR	50		<input type="checkbox"/>		No default
5	fac_localidad	VARCHAR	50		<input checked="" type="checkbox"/>		NULL
6	env_localidad	VARCHAR	50		<input type="checkbox"/>		No default
7	fac_codigo_postal	VARCHAR	50		<input checked="" type="checkbox"/>		NULL
8	env_codigo_postal	VARCHAR	50		<input type="checkbox"/>		No default
9	fac_provincia	VARCHAR	50		<input checked="" type="checkbox"/>		NULL
10	env_provincia	VARCHAR	50		<input type="checkbox"/>		No default
11	id_transporte	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
12	id_forma_pago	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
13	estado	VARCHAR	20		<input type="checkbox"/>		No default
14	activo	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1
15	fecha	DATE			<input checked="" type="checkbox"/>		NULL

Figura 31: tabla de pedidos

7. Producto: en esta tabla tenemos todos los campos necesarios para guardar un productos. Estos campos son la categoría a la que pertenece, en nombre, la dirección URL donde se encuentra la imagen, la descripción, el precio y el stock.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	id_categoria	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
3	nombre	VARCHAR	50		<input type="checkbox"/>		No default
4	direccion_URL	VARCHAR	50		<input type="checkbox"/>		No default
5	descripcion	TEXT			<input checked="" type="checkbox"/>		No default
6	precio	DOUBLE		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
7	stock	INT	11	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0
8	activo	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 32: Tabla del producto

8. Sesión: es esta tabla tenemos todos los campos correspondientes para guardar la sesión de un cliente. Estos campos son el identificador del usuario, el identificador de la sesión, la fecha de alta y la fecha de baja.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	id_user	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default
3	id_sesion	TINYTEXT			<input type="checkbox"/>		No default
4	fecha_alta	DATETIME			<input checked="" type="checkbox"/>		NULL
5	fecha_baja	DATETIME			<input checked="" type="checkbox"/>		NULL
6	activo	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 33: Tabla de la sesión

9. Transporte: En esta tabla tenemos todos los campos correspondientes para guardar los diferentes tipos de transporte. Estos campos son el nombre, el precio que tiene cada tipo de transporte y los días de envío que tardan.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	nombre	VARCHAR	80		<input type="checkbox"/>		No default
3	precio	FLOAT		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
4	dias_envio	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
5	activo	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 34: Tabla del transporte

6.2.3 Detalle de las tablas utilizadas para la interfície gráfica

Como he dicho anteriormente, tengo una serie de tablas que las utilizo para la interfície gráfica. Estas tablas las he creado porque me ha parecido una forma sencilla de que el administrador de la tienda pueda modificar parte de la apariencia sin tener que modificar el código fuente. A continuación voy a hacer un listado de las tablas que utilizo.

1. Contacto: En esta tabla tenemos los campos necesarios para poder guardar las consultas que nos lleguen de los clientes a través de “Contacto”. Estos campos son el nombre del cliente, su email, su teléfono, la consulta que nos realiza y la fecha en que realiza la consulta.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	nombre	VARCHAR	80		<input checked="" type="checkbox"/>		NULL
3	email	VARCHAR	50		<input type="checkbox"/>		No default
4	telefono	VARCHAR	9		<input type="checkbox"/>		No default
5	consulta	VARCHAR	400		<input type="checkbox"/>		No default
6	fecha	DATETIME			<input checked="" type="checkbox"/>		NULL
7	activo	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 35: Tabla de contacto

2. Donde estamos: en esta tabla tenemos todos los campos necesarios para guardar la información que aparece por pantalla en el apartado “Donde estamos”. Esta información es la dirección, el código postal, la provincia, la localidad y un mapa de localización.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	direccion	VARCHAR	50		<input type="checkbox"/>		No default
3	codigo_postal	VARCHAR	5		<input type="checkbox"/>		No default
4	provincia	VARCHAR	50		<input type="checkbox"/>		No default
5	localidad	VARCHAR	50		<input type="checkbox"/>		No default
6	mapa	TEXT			<input type="checkbox"/>		No default
7	activo	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 36: Tabla de donde estamos

3. Artículos: Esta tabla guarda los campos necesarios para que se muestre por pantalla un artículo con su título y su texto.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	titular	VARCHAR	50		<input checked="" type="checkbox"/>		NULL
3	contenido	LONGTEXT			<input checked="" type="checkbox"/>		No default
4	activo	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 37: Tabla de artículos

4. Ofertas: En esta tabla guardamos los campos necesarios para crear una oferta. En ofertas, hago una referencia al artículo que he creado (mediante el identificador id_reg) para que muestre la oferta por pantalla. Esta oferta tiene titular y contenido y una dirección URL por si se tuviera que mostrar algún producto.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	titular	VARCHAR	50		<input type="checkbox"/>		No default
3	contenido	LONGTEXT			<input type="checkbox"/>		No default
4	direccion_URL	VARCHAR	50		<input type="checkbox"/>		No default
5	id_reg	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
6	activo	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 38: Tabla de las ofertas

5. Enlaces: En esta tabla tenemos todos los campos necesarios para guardar los enlaces que tendremos en nuestra tienda. De estos enlaces guardamos el título que queremos que tengan y el enlace al cual nos van a llevar mediante una nueva pestaña.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	titulo	VARCHAR	50		<input type="checkbox"/>		'0'
3	enlace	TEXT			<input type="checkbox"/>		No default
4	activo	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 39: Tabla de enlaces

6. Facebook: En esta tabla guardamos los campos necesarios para que nos aparezca un icono del facebook en un lateral de nuestra tienda, que sea un link que nos lleve hasta la página de facebook de la empresa.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	nombre	VARCHAR	80		<input checked="" type="checkbox"/>		NULL
3	link	VARCHAR	250		<input type="checkbox"/>		No default
4	activo	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 40: Tabla de facebook

7. Menú administración: En esta tabla guardamos todos los campos necesarios para poder crear el menú de la parte de administración. Título hace referencia al título que nos saldrá en cada opción del menú, opción hace referencia a la carpeta que queremos acceder, vista a la vista que queremos mostrar y orden al orden que queremos en que aparezca en el menú.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	titulo	VARCHAR	50		<input type="checkbox"/>		No default
3	opcion	VARCHAR	50		<input type="checkbox"/>		No default
4	vista	VARCHAR	50		<input type="checkbox"/>		No default
5	orden	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
6	activo	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 41: Tabla del menú de administración

8. Menú cliente : En esta tabla guardamos todos los campos necesarios para poder crear el menú del cliente. Título hace referencia al título que nos saldrá en cada opción del menú, opción hace referencia a la carpeta que queremos acceder, vista a la vista que queremos mostrar y orden al orden que queremos en que aparezca en el menú.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	titulo	VARCHAR	50		<input type="checkbox"/>		No default
3	opcion	VARCHAR	50		<input type="checkbox"/>		No default
4	vista	VARCHAR	50		<input type="checkbox"/>		No default
5	orden	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
6	activo	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 42: Tabla del menú del cliente

13. Menú general: En esta tabla guardamos todos los campos necesarios para poder crear el menú general de la parte pública. Título hace referencia al título que nos saldrá en cada opción del menú, opción hace referencia a la carpeta que queremos acceder, vista a la vista que queremos mostrar y orden al orden que queremos en que aparezca en el menú.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	titulo	VARCHAR	50		<input type="checkbox"/>		No default
3	opcion	VARCHAR	50		<input type="checkbox"/>		No default
4	id_reg	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default
5	vista	VARCHAR	50		<input type="checkbox"/>		No default
6	orden	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0
7	activo	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 43: Tabla del menú general

18. Datos empresa: En esta tabla guardamos los campos necesarios para obtener los datos de la empresa. Estos datos son la razón social, el cif, la dirección, el código postal, la localidad y la provincia, y los utilizaremos al generar la factura en pdf para que salgan reflejados en ella.

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	razon_social	VARCHAR	50		<input type="checkbox"/>		No default
3	cif	VARCHAR	50		<input type="checkbox"/>		No default
4	direccion	VARCHAR	50		<input type="checkbox"/>		No default
5	codigo_postal	VARCHAR	50		<input type="checkbox"/>		No default
6	localidad	VARCHAR	50		<input type="checkbox"/>		No default
7	provincia	VARCHAR	50		<input type="checkbox"/>		No default
8	activo	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1

Figura 44: Tabla de los datos de la empresa

Planificación

7.1. Introducción

Por definición la planificación es la predeterminación de un curso de acción para alcanzar los objetivos impuestos. La planificación involucra la especificación de objetivos y metas para un proyecto y las estrategias, políticas, planes y procedimientos para alcanzarlos.

Todo proyecto de ingeniería de software debe partir con un buen plan. La planificación es necesaria por la existencia de incertezas sobre el ambiente del proyecto software y sobre fuentes externas. La planificación enfoca su atención en las metas del proyecto, riesgos potenciales y problemas que puedan interferir con el cumplimiento de esas metas.

Los principales problemas en la planificación de un proyecto de software incluyen los siguientes:

1. Requerimientos incorrectos e incompletos.
2. Muchas especificaciones de requerimientos son inestables y sujetas a cambios erróneas.
3. La planificación no se lleva a cabo por la creencia errónea de que es una pérdida de tiempo y los planes cambiarán de todos modos.
4. La planificación de costos y plazos no es actualizada y se basa en necesidades de mercadeo y no de los requerimientos del sistema.
5. Es difícil estimar el tamaño y complejidad del proyecto de software de modo de realizar una estimación de costos y plazos realista.
6. Los costos y plazos no son re estimados cuando los requerimientos del sistema o el ambiente de desarrollo cambia.
7. No se manejan factores de riesgo.

Actividades que se derivan de la planificación:

1. Fijar los objetivos y metas.
2. Desarrollar estrategias.
3. Desarrollar políticas.

4. Anticipar futuras situaciones.
5. Conducir un establecimiento de riesgos.
6. Determinar posibles cursos de acción.
7. Tomar decisiones de planificación.
8. Fijar procedimientos y reglas.
9. Desarrollar los planes del proyecto.
10. Preparar presupuestos.
11. Documentar los planes del proyecto.

A lo largo de todo este apartado se va a exponer la manera de planificar el proyecto. El primer paso será realizar un estudio de las tareas en las cuales se divide el proyecto, as como la distribución de las tareas para los empleados y desarrolladores del proyecto con el de asignar unas fechas de inicio y fin a cada tarea involucrados en el mismo.

También se estudiarán los riesgos que pueden afectar al proyecto, en que medida lo harán y cual será su probabilidad de aparición as como los factores indicativos y los planes de actuación.

Por ultimo veremos el esfuerzo total del proyecto y por cada fase del proceso unificado de desarrollo de software y también una calendarización. Establece un ciclo de vida formado por cuatro fases:

1. Inicio
2. Elaboración
3. Construcción
4. Transición

Cada fase esta formada por los siguientes flujos de trabajo:

1. Reevaluación de los requisitos
2. Requisitos
3. Análisis
4. Diseño
5. Implementación
6. Prueba
7. Evaluación de la iteración

Por cuestiones de brevedad de tiempo en el plazo asignado para sacar adelante un proyecto software de enormes dimensiones, el plan de iteraciones se ha visto reducido en gran medida, así como las exigencias a nivel de código de la aplicación. Se ha realizado para ello una iteración en las fases de construcción y transición, mientras que en las dos primeras fases se ha dedicado mas tiempo para definir bien el modelo conceptual del sistema.

7.2. Tareas generales

Se va a dividir el proyecto en tareas para poder estudiarlo de un modo más sencillo.

Inicio

En esta fase es donde comienza el proyecto; es donde encontramos el problema y lo

refinamos para poder darle una solución.

Aprendizaje

En esta fase es donde invertiremos tiempo en varios aspectos, por una parte estará la parte tecnológica que nos ayudara para dar una solución de calidad a nuestro proyecto, y por otro lado estará la parte teórica para poder hallar la solución.

Diseño

En esta fase trataremos de diseñar nuestra solución, ya conocemos el problema y además hemos estudiado técnica para resolverlo; por lo tanto ya tenemos una formación para dar nuestra solución. De este modo el desarrollador no se perderá entre toda la información recopilada y podrá implementar la solución de una manera eficiente, sin perdidas de productividad. Se utilizara en este apartado la notación para el modelado de UML, utilizaremos estos diagramas para diseñar la solución y además para evitar dejarnos algún cabo suelto en nuestra solución.

Implementación

Crearemos el código fuente de la solución, basándonos en lo obtenido en el diseño.

Pruebas

Se creara una batería de pruebas para asegurar el correcto funcionamiento. Si se encuentran fallos se corregirán y se repetirán las pruebas para que si al arreglar un fallo hemos sacado otro fallo nuevo, darnos cuenta y poder arreglarlo.

Documentación

Se incluyen en esta fase tanto la creación de un manual del usuario, como la documentación referente al programa. Esto ocurre por varios motivos:

- El manual del usuario debe ser claro, y de este modo podremos ir haciendo nosotros las cosas para asegurarnos de que funciona correctamente si se cumplen los requisitos del manual del usuario.
- Si alguna vez el proyecto se debe retomar, es imprescindible una buena documentación del código para no perder el hilo de lo que se estaba haciendo.

Por todos estos motivos se definen ya una serie de acciones que el desarrollador del proyecto llevara a cabo:

- Dejar el manual del usuario para el momento final, probablemente sea el mejor momento de realizarlo.
- El código se documentara mientras se va escribiendo.

7.3. Tareas específicas

A continuación dividiremos las tareas generales en tareas específicas que contienen cada una de las diferentes fases.

Inicio

- La fase de nacimiento del proyecto.
- El usuario se da cuenta del problema que existe.
- Se refina el problema para tener un problema mas concreto.
- Se evalúan los riesgos y la arquitectura candidata.

- Se revisan las pautas.
- Fase de Inicio completada.

Aprendizaje

- Se exponen los pasos que se van a seguir en la fase de Aprendizaje.
- Adquisición de libros y/o manuales dedicados al tema.
- Consultas a páginas web especializadas en la materia.
- Instalación y puesta a punto de los equipos para las nuevas tecnologías estudiadas.
- Realizar pequeñas pruebas sobre implementaciones sencillas.
- Evaluar el nivel de conocimientos adquiridos.
- Aprendizaje completado.

Diseño

- Se exponen los pasos que se van a seguir en la fase de Diseño.
- Desarrollar especificaciones de funcionamiento.
- Especificar los escenarios.
- Desarrollar varias arquitecturas.
- Evaluar la arquitectura seleccionada.
- Diseño completado.

Implementación

- Se exponen los pasos que se seguirá en la fase de implementación.
- Revisar el diseño y comprender perfectamente el diseño.
- Asignar el personal de desarrollo.
- Desarrollar la primera versión del producto.
- Desarrollo completado.

Pruebas

- Se exponen los pasos que se van a seguir en la fase de Pruebas.
- Desarrollar planes de prueba de unidad usando las especificaciones del producto.
- Desarrollar planes de prueba de integración usando las especificaciones del producto.
- Pruebas de unidad:
 - Revisión de la modularidad del código.
 - Revisión de la calidad del código.
 - Probar si los módulos se ajustan a las especificaciones y a los escenarios.
 - Identificar anomalías y anotarlas en las especificaciones del producto.
 - Modificar el código.
 - Repetición de las pruebas con el código modificado. Si es correcto seguiremos adelante; si no lo es, repetiremos la revisión.
 - Pruebas de integración completadas.

Documentación

- Manual del usuario.
- Se exponen los pasos que se van a seguir en esta fase.
- Desarrollar las especificaciones del manual de usuario.
- Desarrollar el manual de usuario.
- Revisar la documentación para el usuario, si es correcta continuar, en caso contrario

- modificaremos hasta que creamos que es correcta...
- Incorporar comentarios a la documentación de usuario.
- Documentación completada.

7.4. Gestión de riesgos

El análisis de riesgos no es un proceso estático. Las conclusiones deben revisarse cuando se obtiene información adicional o cuando las circunstancias varían. Es normal comparar entre diferentes opciones posibles, contrastando los riesgos y las rentabilidades potenciales; la hipótesis de contraste que se emplea siempre es la llamada 'hipótesis cero', esto es, la opción de no hacer nada.

El análisis de riesgos consiste sobre todo en un cálculo de probabilidades de ocurrencia de sucesos de valoración diversa. Las opciones más rentables son siempre las más arriesgadas, por ello la decisión final dependerá de la aversión al riesgo del directivo o agente decisivo. Si es alta, elegirá la opción que implique las menores pérdidas posibles.

A la hora de afrontar un proyecto de cualquier tipo siempre se verá amenazado por riesgos, habrá riesgos particulares de cada proyecto y otros comunes a cualquier tipo de proyecto.

Evaluaremos la pérdida y la incertidumbre de cada caso, aquí planteamos los riesgos que nos preocupan. Conforme el método 80-20, dice la estadística que los riesgos con 80% o más de incertidumbre y pérdida, son solo un 20% de los riesgos posibles. Debemos acotar nuestra planificación, puesto que de no hacerlo a la propia gestión de riesgos del proyecto se convertirá en otro proyecto.

Si los riesgos que se hacen realidad afectan al proyecto, por norma general, afectan al plazo de entrega. Por este motivo será conveniente no fijar un plazo definitivo de entrega hasta no haber planteado las medidas a tomar en caso de cumplimiento de riesgos.

Riesgos del proyecto:

- Incremento en costes.
- Desbordamiento organizativo.
- Riesgos técnicos.

Si el proyecto tuviera salida comercial, se deberá considerar también la aparición de riesgos relacionados con el negocio en sí. Los más importantes serán:

1. Riesgos del negocio:

- De mercado.
- De estrategia.
- De ventas.
- De gestión.
- De presupuesto.

2. A su vez, se identifican cuatro componentes del riesgo en el proyecto:

- Rendimiento.

- Coste.
- Mantenibilidad.
- Planificación.

7.5 Esfuerzo

Método de estimación mediante UUCP sin ajuste: los casos de uso son dar una explicación de lo que se quiere hacer con el sistema (desde el punto de vista del usuario).

Son casos concretos del funcionamiento de la aplicación. Los clasificamos en:

Tipo	Descripción
Simple	Menor o igual a 3 transiciones
Medio	De 4 a 7 transiciones
Compuesto	Mayor de 7 transiciones

Calculamos el UUCP (puntos de casos de uso sin ajustar):

Tipo	Factor
Simple	5
Medio	10
Complejo	15

Número	Caso de uso	Complejidad
1	registrarse	complejo
2	login	medio
3	catálogo	simple
4	novedades	simple
5	Donde estamos	simple
6	enlaces	simple
7	Buscar productos	medio
8	Modificar catálogo	medio
9	Modificar productos	complejo
10	contacto	medio
11	Datos personales	medio
12	Cambiar interfaz de la tienda	complejo
13	Mostrar pedidos	medio
14	Añadir al carrito	complejo
15	Cesta (paso 1)	complejo
16	Cesta (paso 2)	complejo
17	Cesta (paso 3)	complejo
18	Cesta (paso 4)	complejo

19	Cesta (paso 5)	complejo
20	salir	simple

$UUCP = CS * NSimples + CM * NMedios + CC * NComplejos$

$UUCP = 5*5 + 10*6 + 15*9 = 220$ Puntos de caso de uso.

Factores de complejidad tecnica: Ajustamos (WCP) para reflejar la complejidad y la experiencia.

Numero	Factor	Peso	Valoración
1	Sistema distribuido	2	0
2	Rendimiento	1	4
3	Eficiencia para el usuario final	1	2
4	Procesado interno complejo	1	5
5	Código reusable	1	5
6	Fácil de instalar	1	4
7	Fácil de usar	0,5	5
8	Portable	0,5	4
9	Fácil de modificar	2	2
10	Concurrente	1	0
11	Características especiales	1	2
12	Da acceso directo a terceras partes	1	4
13	Ayudas especiales para la formación de usuarios	1	0

$TCF = 0,6 + (0,01 * \sum_{i=1}^n Valoracion_i * Peso_i)$

$TCF = 0,6 + (0,01 * (2*0 + 1*4 + 1*2 + 1*5 + 1*5 + 1*4 + 0,5*5 + 0,5*4 + 2*2 + 1*0 + 1*2 + 1*4 + 1*0)) = 0,6 + (0,01 * 34,5) = 0,945$ Puntos de Complejidad Técnica.

7.6 Planificacion desglosada del Proyecto

Tiempo estimado por cada fase del proceso unificado. En este apartado se van a agrupar las distintas fases del proyecto en las fases del proyecto unificado. La jornada es de 4 horas diarias no pudiendo ser 8 debido a otros compromisos que tienen los desarrolladores.

Fase Inicio

Tarea	Descripción	Duración
T1	Análisis de negocio	2 días

T2	Especialización de requisitos	3 días
T3	Se evalúan los riesgos y la arquitectura candidata	2 días
T4	Se revisan las pautas	2 días
T5	Fin de fase de inicio	0 días

TOTAL 9 días.

Fase Elaboración: aquí incluiremos tanto el aprendizaje como el diseño.

Tarea	Descripción	Duración
T6	Adquisición de libros y/o manuales dedicados al tema	5 días
T7	Formación en nuevas herramientas	10 días
T8	Instalación y puesta a punto de equipos	1 día
T9	Realizar pequeñas pruebas sobre implementaciones sencillas	5 días
T10	Evaluar el nivel de conocimientos adquiridos	3 días
T11	Aprendizaje completado	0 días
T12	Desarrollar varias arquitecturas	25 días
T13	Planificación del ciclo de elaboración	2 días
T14	Desarrollar varias arquitecturas	25 días
T15	Refinamiento de los casos de uso	7 días
T16	Evaluar la arquitectura seleccionada	2 días
T17	Diseño completado	0 días

TOTAL 63 das.

Fase Construcción: incluimos aquí la fase de Implementacion y la fase de Pruebas.

Tarea	Descripción	Duración
T18	Revisar el diseño y comprenderlo.	1 día
T19	Asignar el personal de desarrollo	0 días
T20	Desarrollar la primera versión del producto.	40 días
T21	Prueba alfa	15 días
T22	Diseño completado	0 días
T23	Gestión y optimización del proceso	4 días
T24	Diagrama de PERT	1 día

Total 61 días

Fase Transición

Tarea	Descripción	Duración
T25	Desarrollar las especificaciones	1 día

T26	Desarrollar el manual de usuario	4 días
T27	Revisar la documentación del usuario	2 días
T28	Generar la documentación del código	1 día
T29	Revisar la documentación del código	1 día
T30	Documentación completada	0 días

Total 11das.

7.7 Diagrama de PERT

Antes de empezar, vamos a transformar cada etapa en una tabla para que haga de nodo y simplificarlo para evitar tener mas de 30 estados con actividades entre ellos y complicar demasiado el diagrama PERT.

Etapas Nodo:

- Inicio A
- Aprendizaje B
- Diseño C
- Implementación D
- Pruebas E
- Corrección Errores F
- Documentación G
- Entrega Final H

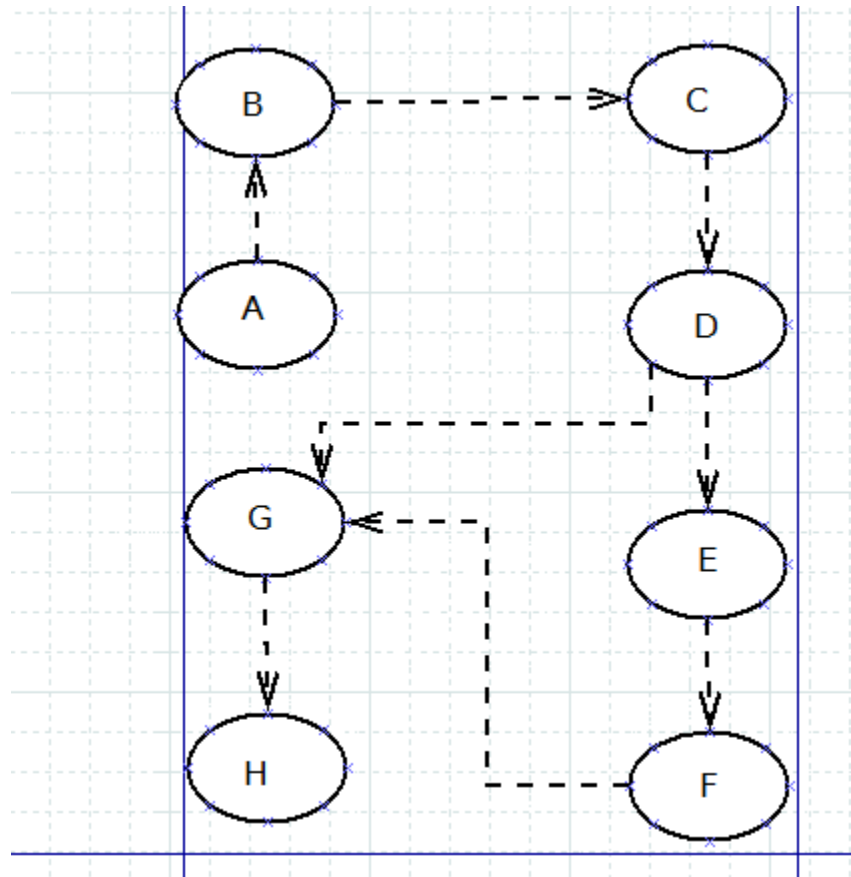


Figura 45: Diagrama de Pert

7.8 Líneas base del proyecto

Las especificaciones de la arquitectura para la aplicación van a ser las ocupadas por el servidor, ordenador que gestionara la aplicación, que a continuación se detallan.

Requisitos del servidor (mínimos):

- Windows 98/2k/ME/XP, Linux, Solaris.
- Pentium III 1000 Mhz o AMD equivalente.
- 512 MB RAM
- Tarjeta de red 10/100 Ethernet
- Lector de CD/DVD
- Espacio libre en disco duro de 40 GB.

Manual de Usuario

8.1 Requisitos Hardware y Software

8.1.1 Hardware

Servidor: Requisitos mínimos y recomendados.

- Procesador 233 MHz (recomendado de 2 GHz o superior) Intel Pentium/Celeron o AMD K6/Athlon/Duron o compatible.
- 128 MB de RAM (recomendado 1GB).

- 200 MB disponibles en el disco duro (se recomienda mas de 10GB).
- Adaptador de vídeo Super VGA (800 x 600).
- Unidad de CDROM.
- Internet Explorer, Firefox o cualquier navegador (se recomienda FireFox version 3.0).
- Conexión a internet (recomendado internet de alta velocidad).

Cliente: Requisitos mínimos y recomendados.

Cualquier dispositivo con navegador y conexión a internet es suficiente, como por ejemplo PDA's, Play Station Portable, móviles, portátiles, sobremesas, etc. No obstante si accedes a través de un sobremesa estos son los requisitos:

Windows:

- Requisitos del sistema mínimos:
 - Procesador a 233 Mhz
 - 64 MB de RAM
 - 50 MB de espacio libre en disco
- Requisitos recomendados:
 - Procesador a 500 Mhz²
 - 56 MB de RAM
 - 100 MB de espacio libre en disco

Mac

- Requisitos del sistema mínimos:
 - Procesador PowerPC G3
 - 128 MB de RAM
 - 75 MB de espacio libre en disco
- Requisitos recomendados:
 - Procesador PowerPC G4 o Intel
 - 512 MB de RAM
 - 150 MB de espacio libre en disco

Linux

- Requisitos del sistema mínimos:
 - Procesador a 233 Mhz
 - 64 MB de RAM
 - 50 MB de espacio libre en disco
- Requisitos recomendados:
 - Procesador a 500 Mhz
 - 256 MB de RAM
 - 100 MB de espacio libre en disco

8.1.2 Software

- Filezilla
- HeidiSQL
- Windows XP SP2

8.2 Manual de Instalación

La instalación la vamos a dividir en dos partes. La parte de la base de datos y la parte del filezilla.

8.2.1 Instalación de la base de datos

Vamos a instalar nuestro gestor de la base de datos. Nuestro gestor es HeidiSQL. A continuación voy a enumerar los pasos necesarios para conseguir realizar la instalación satisfactoriamente:

1. Descargamos el programa HeidiSQL y le damos a instalar.
2. Nos saldrá una pantalla de bienvenida. Pulsamos next.
3. Nos sale una nueva pantalla. Seleccionamos la opción de que aceptamos la licencia y le volvemos a dar a next.
4. Dejamos el directorio de instalación que nos muestra el programa por defecto.
5. Dejamos el nombre de la carpeta también por defecto.
6. Nos sale una pantalla con la opción de crear un icono en el escritorio. La dejamos seleccionada y seleccionamos la opción también de asociación de ficheros SQL (en principio está opción tiene que venir seleccionada). Pulsamos next
7. Nos aparecerá en la pantalla un resumen con los datos de instalación y le damos a la opción “install” para proceder a la instalación del programa.
8. Una vez tenemos instalado el programa, tenemos que introducir los datos la empresa de hosting que tenemos contratado nos ha proporcionado.
9. Cargar el script para crear las bases de datos.

8.2.2 Instalación del FileZilla

El FileZilla es un programa que nos permite subir ficheros a nuestra FTP. Nos va a permitir transferir archivos desde nuestro ordenador a nuestro servidor FTP y viceversa.

A continuación voy a detallar los pasos necesarios para la correcta instalación de este programa.

1. Descargamos el programa FileZilla y le damos a instalar.
2. Nos aparecerá una pantalla pidiendo que aceptemos la licencia. Pulsamos el botón I Agree.
3. A continuación nos aparece una pantalla pidiendo los privilegios de administrador. Seleccionamos la opción all users.
4. Ahora nos muestra los componentes del instalador. Lo dejamos como está por defecto y le damos a next.
5. Nos pide la carpeta de destino de la instalación. Dejamos también la que aparece por defecto y pulsamos next.
6. Finalmente nos pide el nombre que queremos que tenga. Dejamos el nombre que aparece por defecto (FileZilla FTP Client) y pulsamos instalar.
7. Pulsamos finalizar y ya tenemos nuestro programa instalado.
8. Subir los ficheros al servidor.

8.3 Manual de desinstalación

Si queremos eliminar lo instalado tendremos que hacer click en inicio, configuración y panel de control. Una vez dentro de panel de control hacemos doble click en agregar o quitar programas y borramos los siguientes archivos:

- HeidiSQL 6.0
- FileZilla Client 3.5.0

8.4 Manual de Uso de la Aplicación

Podemos acceder de dos maneras a la tienda. Podemos acceder mediante una dirección a la parte pública y con otra dirección podemos acceder a la parte de administrador.

A continuación voy a hacer un resumen detallado de las acciones que se pueden realizar como usuario y las que se pueden realizar como administrador.

USUARIO



Figura 46: Cabecera de la tienda

Para poder acceder a todos los menús deberás estar registrado. Para registrarse tenemos como un cliente nuevo hay que pulsar en cliente nuevo. Si ya estas registrado, simplemente tienes que introducir tu usuario y tu contraseña y pulsar en enviar.

Una vez pulsas cliente nuevo, aparece un formulario que tienes que rellenar con los datos pedidos. Hay una serie de datos que son obligatorios y otros que son opcionales.

Registrar cliente nuevo

Nombre:	<input type="text"/>
Apellidos:	<input type="text"/>
D.N.I.:	<input type="text"/>
Fecha nacimiento:	<input type="text"/>
Telefono:	<input type="text"/>
Email:	<input type="text"/>
usuario:	<input type="text"/>
password:	<input type="text"/>
Dirección de envío:	
Dirección:	<input type="text"/>
Localidad:	<input type="text"/>
Provincia:	<input type="text"/>
Codigo Postal::	<input type="text"/>
Dirección de facturación (opcional):	
Dirección:	<input type="text"/>
Localidad:	<input type="text"/>
Provincia:	<input type="text"/>
Codigo Postal::	<input type="text"/>

Figura 47: Formulario para un nuevo cliente

Como podemos apreciar en la fotografía los datos opcionales son los campos de la dirección de facturación.

Una vez estamos dentro de la tienda virtual como usuario identificado, ya podemos acceder a todas las funcionalidades que posee la tienda.



Figura 48: Menú para un cliente registrado o logueado

Una vez ya estamos dentro de la tienda, podemos ir consultando productos e insertándolos en nuestro carrito haciendo click en 'añadir al carrito'. Cuando queramos realizar la compra tenemos que confirmar el carrito y seguir los pasos que nos van apareciendo uno detrás de otro mediante la pulsación del botón siguiente. También podemos ver los pedidos que tiene realizado un cliente con los productos que ha comprado en cada pedido.

Para poder entrar en la aplicación como usuario tendremos que acceder a la siguiente URL: <http://tiendavirtual.zierzoware.es/>

ADMINISTRADOR

Para entrar en la parte de administración tenemos que entrar en la URL: <http://tiendavirtual.zierzoware.es/administrador/login.php>

LOGIN

Usuario:

Contraseña:

Figura 49: Pantalla para acceder como administrador

En esta página nos pedirá el usuario y la contraseña para poder acceder. El usuario es admin y la contraseña es 1234.

Una vez accedemos como administrador del sistema podemos ver un menú con todas las opciones que tenemos:

- Artículos: nos permite crear un artículo nuevo, o modificar los ya existentes.
- Categorías: Aquí podemos ir agregando o eliminando categorías.
- Contacto: podemos añadir, eliminar o modificar los datos de contacto.
- Enlaces: podemos añadir, eliminar o modificar enlaces.
- Formas de pago: podemos añadir, eliminar o modificar las formas de pago.
- Menú administración: podemos añadir, eliminar o modificar los datos que se nos muestran en el menú de la parte de administración.

- Menú general: podemos añadir, eliminar o modificar los datos que se nos muestran en el menú de la parte pública (parte que ve el usuario).
- Menú cliente: podemos añadir, eliminar o modificar los datos que se nos muestran en el menú del cliente (menú que aparece con las funcionalidades nuevas que puede ver un cliente cuando está registrado).
- Ofertas: podemos añadir, eliminar o modificar las ofertas.
- Pedidos: podemos añadir, eliminar o modificar los pedidos.
- Productos: podemos añadir, eliminar o modificar los productos.
- Transporte: podemos añadir, eliminar o modificar el tipo de transporte.

Juegos de Pruebas

En este apartado voy a enumerar los llamados juegos de pruebas que he realizado para comprobar que toda nuestra aplicación funciona correctamente. Estos juegos de pruebas los he realizado en la parte de administrador.

9.1 Login Usuario

En este juego de pruebas voy a comprobar que funciona el login del usuario. Es decir, vamos a comprobar que al introducir una contraseña correcta, podemos entrar en la parte de administración, pero si introducimos una contraseña incorrecta.

Como he indicado anteriormente (en el manual de usuario) el login y la contraseña para entrar como administrador es: login -> admin, contraseña -> 1234.

El primer paso es entrar en la página <http://tiendavirtual.zierzoware.es/administrador> que nos redirecciona automáticamente a la siguiente página <http://tiendavirtual.zierzoware.es/administrador/login.php>

En esta página nos saldrá una pantalla como la siguiente:

LOGIN

Usuario:	<input type="text" value="admin"/>
Contraseña:	<input type="password" value="...."/>
<input type="button" value="entrar"/>	

Figura 50: Pantalla para acceder como administrador

Podemos apreciar que por defecto ya nos vienen los campos de usuario y de contraseña debidamente rellenos. Si modificamos los campos de usuario y/o contraseña introduciendo datos incorrectos, podremos observar que cuando pulsamos el botón entrar no nos lleva a ninguna página. En cambio si introducimos los datos correctos y pulsamos el botón de entrar, accederemos a todo el menú de administración que es el que nos permite hacer todas las modificaciones de nuestra tienda virtual.

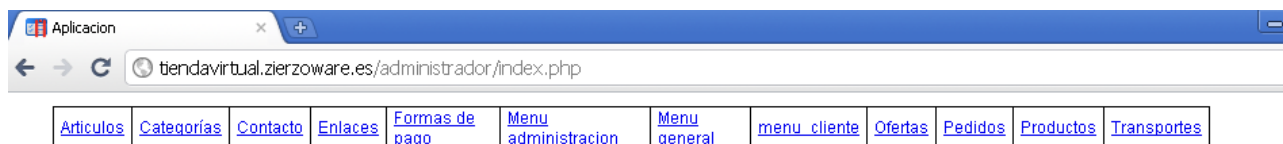


Figura 51: Menú del administrador

Una vez estamos aquí dentro ya podemos modificar todo lo correspondiente a la tienda virtual. A continuación, en los siguientes juegos de pruebas, voy a ir enumerando las pruebas que he ido realizando para comprobar que todas las funcionalidades de la tienda funcionan correctamente.

9.2 Gestión de categorías

En este juego de pruebas vamos a realizar todas las pruebas necesarias para poder constatar el correcto funcionamiento de las categorías. Para poder realizar las pruebas tenemos que situarnos sobre la opción de categorías del menú de administración.

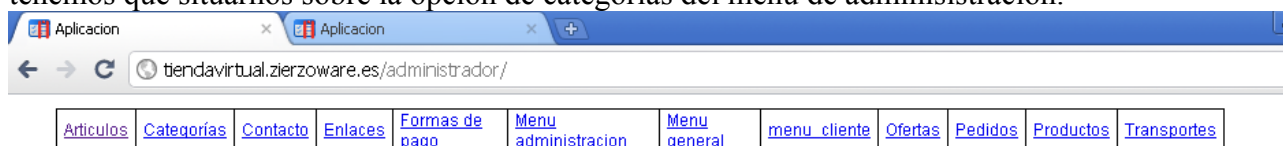


Figura 52: Menu del administrador

Una vez estemos dentro de categorías nos aparecerá un listado con todas las categorías que tenemos. El identificador de cada categoría, el nombre y la descripción.



ID	Nombre	Descripcion	
8	Diademas	diademas	
9	DIADEMAS INFANTILES	Diademas infantiles	
11	pasadores	pasadores	
10	PRENDIDOS INFANTILES	Prendidos infantiles	
12	tocados	tocados	

Figura 53: Listado de categorías en la parte de administración

Lo primero que vamos a hacer va a ser crear una nueva categoría. Posteriormente, comprobaremos que esta categoría se muestra, la modificaremos y la eliminaremos. Creamos una nueva categoría:

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



ID	Nombre	Descripcion		
13	categoria de prueba	categoria de prueba		
8	Diademas	diademas		
9	DIADEMAS INFANTILES	Diademas infantiles		
11	pasadores	pasadores		
10	PRENDIDOS INFANTILES	Prendidos infantiles		
12	tocados	tocados		

Figura 54: Listado de categorías en la parte de administración con la categoría nueva creada

Vemos que ahora en la lista nos aparece una nueva categoría llamada categoría de prueba con el identificador nº 13. Esta es la categoría que acabamos de crear. Ahora vamos a la parte pública y comprobamos que la categoría se ha añadido correctamente.

INICIO	CLIENTE NUEVO	DONDE ESTAMOS	ENLACES	CONTACTO
------------------------	-------------------------------	-------------------------------	-------------------------	--------------------------

Usuario: Contraseña:

Flores de pelo

[CATEGORIA DE PRUEBA](#)
[DIADEMAS](#)
[DIADEMAS INFANTILES](#)
[PASADORES](#)
[PRENDIDOS INFANTILES](#)
[TOCADOS](#)

Figura 55: Figura con las categorías en la parte pública

Como podemos apreciar, la categoría nueva aparece en el menu de las categorías por lo que la insercción se ha realizado correctamente. Ahora vamos a modificar los datos de la categoría (en la parte de administración).

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



Modificar los datos de la categoría

Nombre:

Descripcion:

Figura 56: Modificación de una categoría

Modificamos el nombre de la categoría por prueba. Ahora nos tendremos que ir a la parte pública para comprobar que la categoría se ha modificado.



Figura 57: Categorías en la parte pública con la modificación hecha

Como vemos, el nombre se ha cambiado satisfactoriamente, ya que ahora en lugar de aparecernos categoría de prueba nos sale solo prueba. Finalmente para comprobar toda la funcionalidad de la categoría vamos a eliminarla para ver si funciona correctamente.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



Datos categorías

Nombre: prueba

Descripcion: categoria de prueba

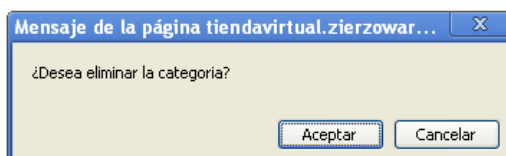


Figura 58: Eliminación de una categoría.

Aceptamos, para eliminar la categoría, y vamos a la parte pública para comprobar

que ya no nos aparece la categoría, es decir, se ha eliminado la categoría.



Figura 59: Categorías en la parte pública con categoría eliminada

Como podemos apreciar, ya no nos sale la categoría que queríamos eliminar, por lo tanto podemos afirmar que la gestión de categorías funciona correctamente.

9.3 Contacto

En este juego de pruebas vamos a comprobar que funciona el contacto. Si entramos en la parte de administración y clickamos sobre contacto podremos ver que está vacío, eso significa que ningún cliente o usuario se ha puesto en contacto con nosotros.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------

Nombre	Fecha	ID
No se han encontrado contactos para mostrar.		

Figura 60: Listado de los contactos en la parte de administración

Ahora iremos a la parte pública de la tienda y rellenaremos los campos de contacto y los enviaremos a la tienda. Si todo funciona correctamente, cuando volvamos a la parte de administración tendremos que visualizar la petición añadida.



DIADEMAS	DIADEMAS INFANTILES	PASADORES	PRENDIDOS INFANTILE
			 Enviar  Cancelar
<p>Nombre: <input type="text" value="sandra"/></p> <p>Email: <input type="text" value="sandra.gazulla@gmail.com"/></p> <p>Telefono: <input type="text" value="690175816"/></p> <p>Consulta: <input type="text" value="Prueba para ver si funciona el contacto"/></p>			

Figura 61: Formulario para rellenar para realizar un contacto

Ahora pulsamos en enviar y vamos a la parte de administración para comprobar si la gestión de contactos funciona correctamente.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------


Nombre	Fecha	ID	
sandra	2011-06-02 15:15:48	5	

Figura 62: Listado de contactos en la parte de administración con el contacto añadido

Como podemos apreciar, se ha añadido el nombre de la persona que ha realizado la consulta o sugerencia, la fecha y la hora en la que lo ha realizado y el identificador. Ahora si pulsamos en el icono que tenemos al lado del identificador nos saldrá la información del usuario y la información de la petición o consulta que ha realizado.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



[Eliminar](#)



[Listado](#)

Datos de contacto

Nombre: sandra
 Email: sandra.gazulla@gmail.com
 Telefono: 690175816
 Fecha: 02/06/11
 Consulta: prueba para ver si funciona el contacto

Figura 63: Datos del contacto en la parte de administración

En esta pantalla podemos ver toda la información ha introducido el usuario. Ahora podemos volver al listado de todos los contatos, pulsando el icono de listado, o eliminar el contacto. Así pues, vamos a eliminar el contacto para asegurarnos que funciona. Por lo tanto, vamos a pulsar el botón eliminar.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



[Eliminar](#)



[Listado](#)

Datos de contacto

Nombre: sandra
 Email: sandra.gazulla@gmail.com
 Telefono: 690175816
 Fecha: 02/06/11
 Consulta: prueba para ver si funciona el contacto

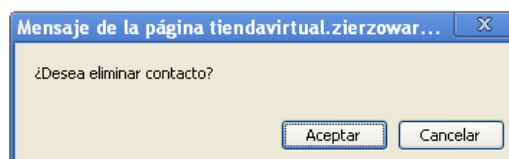


Figura 64: Eliminación de un contacto

Nos aparece un mensaje para confirmar si queremos eliminar. Pulsamos aceptar. Una vez eliminado este contacto, si todo funciona correctamente nos volverá a aparecer el listado vacío ya que solo teniamos una petición de contacto y la hemos eliminado. Así pues vamos a proceder a realizar la acción para asegurarnos de que funciona bien la aplicación.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------

Nombre	Fecha	ID
No se han encontrado contactos para mostrar.		

Figura 65: Listado de contacto con el contacto eliminado

Podemos apreciar, que ahora nos sale una tabla vacía y en la parte inferior nos aparece un mensaje diciendo que no hay contactos para mostrar. Así pues, de esta manera queda probado que el entorno de contacto funciona correctamente.

9.4 Enlaces

En este juego de pruebas voy a comprobar el correcto funcionamiento de los enlaces. Los enlaces los administraremos desde la parte de administración, y cuando queramos abrir uno en la parte pública se nos abrirán en una nueva pestaña. Así pues, ahora voy a ir enumerando los pasos que he realizado para comprobar el correcto funcionamiento de los enlaces.

Como en todos los casos empezaré por el menú de administración. Primero me coloco en el menu de administración general.

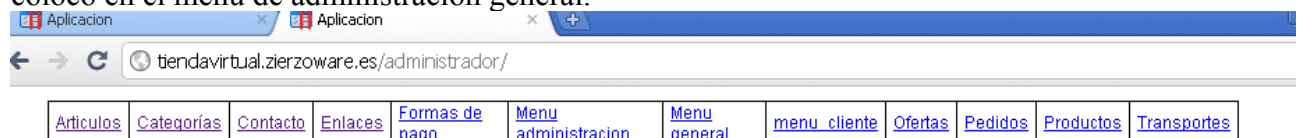


Figura 66: Menú en la parte de administración

Ahora vamos a entrar en enlaces, y empezaremos con toda la funcionalidad de éstos. Así pues, pulsamos sobre enlaces.

Registrar un enlace nuevo

Titulo:

Enlace:

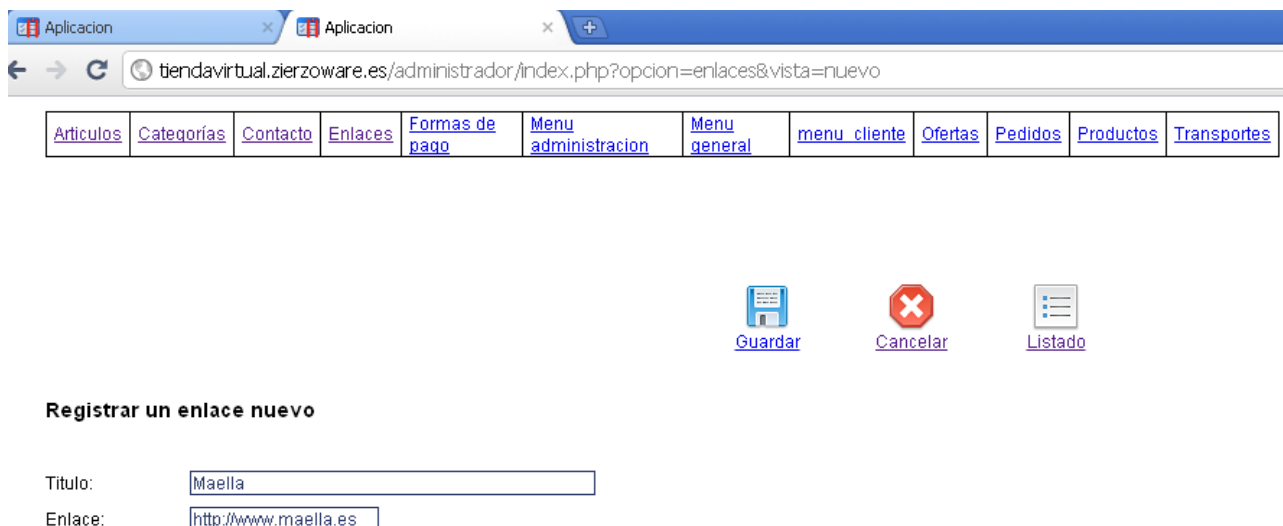
Figura 67: formulario para registrar un enlace nuevo en la parte de administración

Una vez pulsamos sobre enlace podemos registrar un enlace nuevo o ver el listado. Primero voy a mostrar el listado para que se puedan ver los enlaces que tengo, y así, comprobar luego mediante el listado otra vez, que se ha añadido el enlace nuevo. Así pues, pulso sobre listado.

ID	Título	Enlace	
5	Facebook	http://facebook.com	
4	google	http://www.google.es	

Figura 68: listado con los enlaces que tenemos en la parte de administración

Se puede apreciar que tenemos dos enlaces en nuestra página. Ahora vamos a añadir otro enlace con el botón nuevo.



Registrar un enlace nuevo

Titulo:

Enlace:

Guardar Cancelar Listado

Figura 69: Registro de un enlace nuevo en la parte de administración

Pulsamos guardar y ahora ya tendremos el enlace añadido en nuestra tienda. Ahora nos aparece una pantalla con un listado de todos los enlaces que tenemos, y podemos apreciar que aparece el que acabamos de crear.



ID	Titulo	Enlace	
5	Facebook	http://facebook.com	
4	google	http://www.google.es	
6	Maella	http://www.maella.es	

Figura 70: listado de los enlaces en la parte de administración

Ahora para comprobar que los enlaces funcionan bien, voy a ir a la parte pública. Allí clickaré sobre enlaces y me tiene que aparecer el enlace que acabo de crear.

INICIO	CLIENTE NUEVO	DONDE ESTAMOS	ENLACES								
<div><input type="text"/></div> <div>Buscar</div>											
<h1>Flores de pelo</h1>											
DIADEMAS	DIADEMAS INFANTILES	PASADORES	PRE								
<table><tr><td>Titulo</td><td></td></tr><tr><td>Facebook</td><td></td></tr><tr><td>google</td><td></td></tr><tr><td>Maella</td><td></td></tr></table>				Titulo		Facebook		google		Maella	
Titulo											
Facebook											
google											
Maella											

Figura 71: Listado de los enlaces en la parte pública

Como podemos apreciar, ahora el usuario ya puede ver el enlace en la parte pública. Ahora pulsaremos sobre el enlace y si todo funciona correctamente, se nos abrirá la página asociada al enlace en una nueva pestaña.

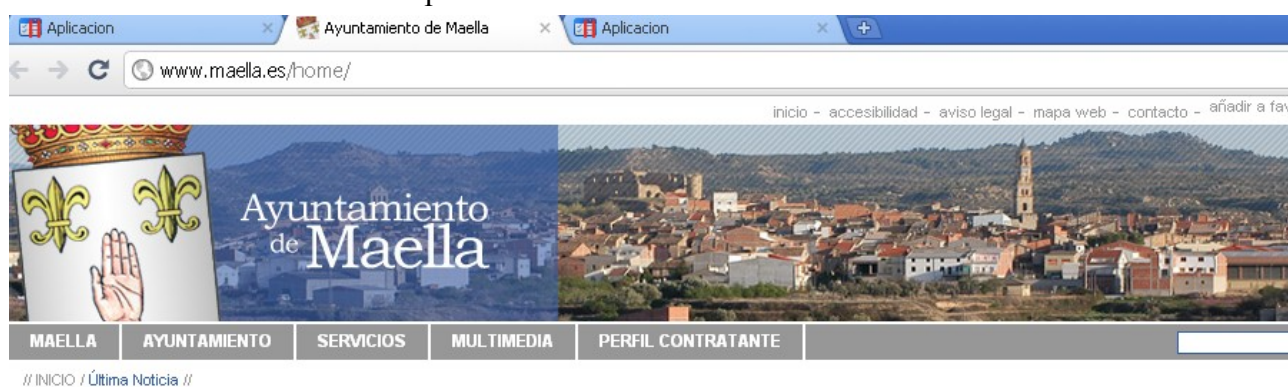


Figura 72: Página de uno de los enlaces, para comprobar que funciona correctamente

Como podemos ver, la página a la que estaba asociada el enlace se ha abierto en una nueva pestaña. Por lo tanto podemos afirmar que los enlaces funcionan correctamente.

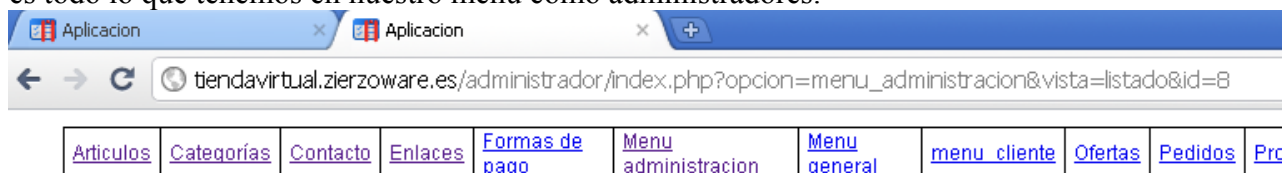
9.5 Menú Administración

El menú de administración es el menú que tenemos en la parte de administración de la base de datos. Para comprobar el correcto funcionamiento de este menú de administración se pueden añadir más secciones. Así pues, primero de todo me sitúo en el menú de la administración, es decir, en la página principal del administrador.



Figura 73: Menú del administrador

A continuación entramos en el menu de administración. Podremos ver que se nos muestra un listado con todo lo que tenemos en el menú de administración que en este caso es todo lo que tenemos en nuestro menu como administradores.



ID	Título	Opcion	Vista	Orden	
18	Articulos	articulos	listado	0	
15	Categorías	categorias	listado	0	
13	Contacto	contacto	listado	3	
19	Enlaces	enlaces	nuevo	2	
7	Formas de pago	formas_pago	listado	0	
8	Menu administracion	menu_administracion	listado	0	
14	Menu general	menu_general	listado	1	
17	menu_cliente	menu_cliente	listado	1	
10	Ofertas	ofertas	listado	0	
21	Pedidos	pedidos	listado_pendientes	0	
16	Productos	productos	listado	0	
9	Transportes	transporte	listado	1	

Figura 74: Listado de todo lo que tenemos en el menú del administrador

Aquí podemos ver que cada elemento tiene un identificador, una opción, una vista y un orden. En el caso de que queramos mostrar más cosas en nuestro menú de administración tendríamos que añadir un nuevo elemento, con su título correspondiente, su opción y con la vista queremos que se muestre. Es decir, si queremos que nos muestre un listado, normalmente la vista a mostrar será listado, pero si en cambio queremos añadir un elemento nuevo (como es el caso de enlaces) la vista será nuevo.

9.6 Menú General

Este juego de pruebas consiste en comprobar el funcionamiento del menú general. Este menú es el menú que ven los usuarios que visitan la tienda virtual sin necesidad de estar registrados. A continuación voy a poner una imagen del listado de los elementos que tengo introducidos en el menú general para comentarlos un poco.

Articulos	Categorías	Contacto	Datos empresa	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



ID	Título	Opcion	Vista	Orden	id_reg	
6	cliente nuevo	clientes	nuevo	1	0	
7	Contacto	contacto	nuevo	3	0	
10	donde estamos	donde_estamos	ver	2	0	
13	enlaces	enlaces	listado	2	0	
15	Inicio	inicio	inicio	0	0	

Figura 75: listado de lo que tenemos en el menú general

En esta imagen podemos apreciar que tenemos introducido en el menú los elementos que queremos que aparezcan con su opción correspondiente y su vista, como en el caso del menú de administración.

De todas formas, voy a añadir una imagen de la parte pública de la tienda virtual para poder comprobar que todos los campos que aquí se citan aparecen realmente.



Figura 76: Menú general en la parte pública

Como podemos apreciar en la imagen, todos los campos que se listan en la parte de administración aparecen en la parte pública.

El menú general funciona correctamente ya que si pulsas sobre cualquiera de los campos (inicio, cliente nuevo, donde estamos, enlaces, contacto y articulos) cumplen con la función por la que han sido diseñados.

9.7 Menú Clientes

Este menú es el que va a aparecer una vez el cliente se registre como nuevo cliente satisfactoriamente o inicie sesión satisfactoriamente. A continuación voy a poner también un listado (desde el menú de administración) de los campos que tiene el menú de clientes.



ID	Titulo	Opcion	Vista	Orden
3	datos cliente	clientes	ver	3
4	pedidos	pedidos	listado	1

Figura 77: listado de lo que tenemos en el menú de los clientes

En esta imagen, podemos apreciar (desde la parte de administración) el listado de los campos que tendrá este menú de clientes. Ahora, como en el caso anterior, voy a poner una imagen de la parte pública de la tienda, una vez se ha identificado un cliente, para comprobar que este menú funciona correctamente.



Figura 78: Menú de los clientes en la parte pública

El menú cliente funciona correctamente ya que si pulsas sobre cualquiera de los campos (datos cliente, pedidos) cumplen con la función por la que han sido diseñados. Por poner un ejemplo, si ahora pulso sobre datos cliente me saldrá la siguiente información:

Datos de Clientes

Nombre: Sandra
Apellidos: Gazulla Gascón
D.N.I.: 73086845K
Fecha nacimiento: 02/12/86
Telefono: 690175816
Email: sandra.gazulla@gmail.com
Usuario: sandra
Password: sandra
Dirección de envío:
c/Baviera nº 13 Maella Zaragoza 50710
Dirección de facturación:

Figura 79: Datos de un cliente

Nos muestra los datos del cliente que tenemos en estos momentos con la sesión iniciada (en este caso son nuestros datos). Por lo tanto podemos afirmar que funciona correctamente.

9.8 Formas de Pago

En este juego de prueba vamos a comprobar que funciona correctamente desde la parte del administrador el añadir, modificar y eliminar una forma de pago. El correcto funcionamiento desde la parte pública lo realizaremos cuando veamos el juego de pruebas de la realización de un pedido.

Así pues, voy a situarme sobre el menu de administrador y pulsar en el campo formas de pago.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



[Nuevo](#)

Nombre	Descripcion	Comision		
Transferencia	pago mediante transferencia bancaria	2		

Figura 80: listado de las formas de pago en la parte de administración

Como podemos apreciar en la imagen nos aparece un listado con todas las formas de pago que hay en la tienda. En este caso solo está el pago mediante transferencia bancaria. Ahora vamos a añadir una nueva forma de pago, que será el pago contra-rembolso. Así pues pulsamos el botón nuevo y rellenamos los campos.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



[Guardar](#)



[Cancelar](#)



[Listado](#)

Registrar una forma de pago nueva

Nombre:

Descripcion:

Comision:

Figura 81: insercción de una nueva forma de pago

Ahora guardamos la nueva forma de pago. Si todo ha funcionado correctamente tenemos que ver en la pantalla un listado con los dos tipos de formas de pago.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



Nombre	Descripcion	Comision		
contra-rembolso	pago mediante contra-rembolso	4		
Transferencia	pago mediante transferencia bancaria	2		

Figura 82: listado de las formas de pago

Como podemos apreciar, la inserción de la nueva forma de pago se ha realizado correctamente. Ahora vamos a hacer la comprobación de que funciona correctamente la opción de modificar una forma de pago. Así pues, voy a modificar la descripción de pago contra-rembolso para ver si funciona correctamente.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



Modificar los datos de formas de pago

Nombre:

Descripcion:

Comision:

Figura 83: Modificación de una forma de pago

Una vez hemos cambiado los campos que queremos modificar, pulsamos el botón de guardar y nos tiene que aparecer otra vez el listado pero con los datos de la forma de pago modificada nuevos.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



[Nuevo](#)

Nombre	Descripcion	Comision	
contra-rembolso	El pago se realiza en el momento que llega el paquete al domicilio	4	
Transferencia	pago mediante transferencia bancaria	2	

Figura 84: Listado de las formas de pago

Como podemos observar la modificación se ha hecho correctamente. A continuación voy a insertar una forma de pago errónea con el fin de poder eliminarla posteriormente y así comprobar que la opción de eliminar funciona correctamente.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



[Nuevo](#)

Nombre	Descripcion	Comision	
contra-rembolso	El pago se realiza en el momento que llega el paquete al domicilio	4	
forma de pago de prueba	forma de pago para comprobar si la opción de eliminar funciona correctamente.	0	
Transferencia	pago mediante transferencia bancaria	2	

Figura 85: listado de las formas de pago

Así pues, se puede apreciar que he añadido una nueva forma de pago que se llama forma de pago de prueba, con su correspondiente descripción y comisión. Ahora voy a eliminar esta forma de pago y así comprobar que la acción de borrar funciona correctamente. El primer paso a realizar es pulsar el botón de edición que tenemos al lado de la comisión.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



Datos formas de pago

Nombre: forma de pago de prueba
 Descripción: forma de pago para comprobar si la correctamente.
 Comisión: 0

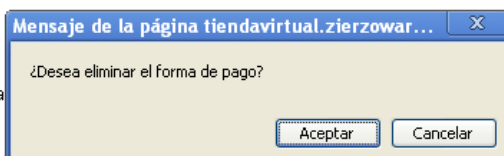


Figura 86: eliminación de una forma de pago

Una vez hemos pulsado el botón del que os he hablado anteriormente, pulsamos el botón de eliminar. Nos saldrá una ventana con un mensaje de si queremos eliminar la forma de pago. Clickamos en aceptar y nos saldrá otra vez una pantalla con el listado de todas las formas de pago que tenemos.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



Nombre	Descripción	Comisión		
contra-rembolso	El pago se realiza en el momento que llega el paquete al domicilio	4		
Transferencia	pago mediante transferencia bancaria	2		

Figura 87: listado de las formas de pago

Como podemos apreciar, la forma de pago de prueba ya no se muestra, por lo tanto podemos afirmar que la opción de eliminar una forma de pago funciona correctamente.

Como he dicho anteriormente, el apartado de comprobación del correcto funcionamiento en la parte pública, lo realizaré en un juego de pruebas a parte, que será el juego de pruebas del proceso de compra. En este juego de pruebas, si las formas de pago funcionan correctamente, nos tendrá que salir en un momento del proceso de compra, un listado con las posibles formas de pago y elegir la deseada.

9.9 Formas de Transporte

En este juego de prueba vamos a comprobar que funciona correctamente desde la parte del administrador el añadir, modificar y eliminar una forma de transporte. El correcto funcionamiento desde la parte pública lo realizaremos cuando veamos el juego de pruebas de la realización de un pedido.

Así pues, voy a situarme sobre el menu de administrador y pulsar en el campo transportes.

Articulos	Categorias	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



[Nuevo](#)

Nombre	Precio	Días de envío	ID
DHL	9	3	1

Figura 88: listado de las formas de transporte en el menú de administración

Como podemos apreciar en la imagen nos aparece un listado con todas las formas de transporte que hay en la tienda. En este caso solo está el transporte mediante DHL. Ahora vamos a añadir una nueva forma de transporte, que será el envío por correos. Así pues pulsamos el botón nuevo y rellenamos los campos.

Articulos	Categorias	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



[Guardar](#)



[Cancelar](#)



[Listado](#)

Registrar transporte nuevo

Nombre:

Precio:

Días de envío:

Figura 89: registro de una nueva forma de transporte

Ahora guardamos la nueva forma de transporte. Si todo ha funcionado correctamente hemos de ver una pantalla con el listado de todas las formas de transporte que tenemos en nuestra tienda.

Articulos	Categorías	Contacto	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



Nombre	Precio	Días de envío	ID	
Correos	5	7	2	
DHL	9	3	1	

Figura 90: listado de las formas de transporte

Como podemos apreciar, la inserción de la nueva forma de de transporte se ha realizado correctamente. Ahora voy a insertar una nueva forma de transporte, y posteriormente la modificaré para hacer la comprobación de que funciona correctamente la opción de modificar una forma de transporte.

Articulos	Categorías	Contacto	Datos empresa	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



Nombre	Precio	Días de envío	ID	
Correos	5	7	2	
DHL	9	3	1	
prueba	1000	1000	3	

Figura 91: listado de las formas de transporte

He insertado una nueva forma de transporte llamada prueba. Ahora la voy a modificar para y así comprobaremos que la forma nueva de transporte se modifica y sus datos se guardan.

Articulos	Categorías	Contacto	Datos empresa	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



Nombre	Precio	Días de envío	ID	
Correos	5	7	2	
DHL	9	3	1	
prueba	1	1	3	

Figura 92: listado de las formas de transporte

Como podemos comprobar, la forma de transporte se ha modificado. Ahora para acabar de comprobar que las formas de transporte funcionan correctamente, voy a eliminar la forma de transporte llamada prueba. Así pues, pulsaremos sobre el botón de edición de prueba y pulsamos eliminar.

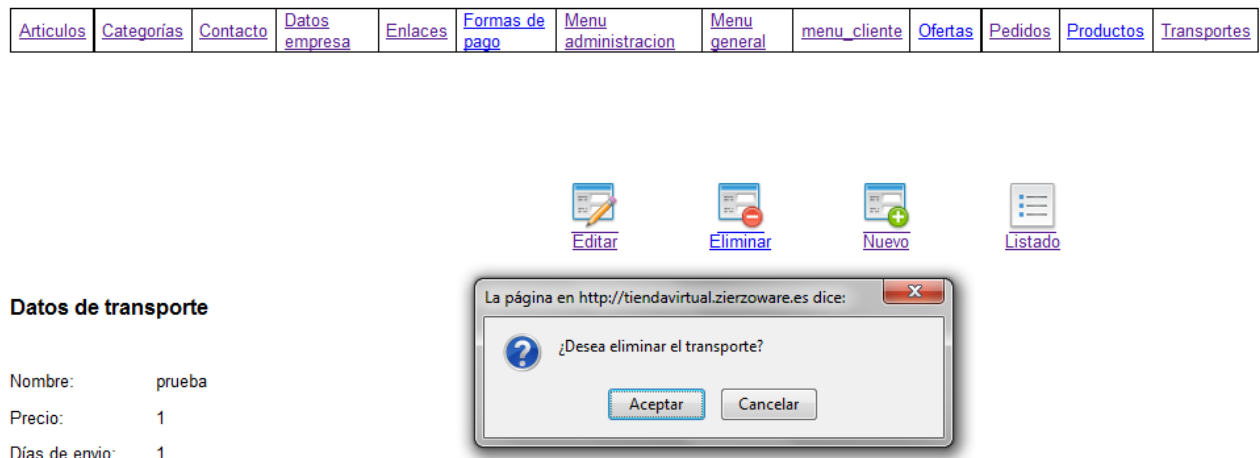


Figura 93: eliminación de una forma de transporte

Nos aparece una ventana de confirmación para eliminar la forma de transporte. Pulsamos aceptar. Si la acción de eliminar funciona correctamente nos aparecerá otra vez el listado de las formas de transporte, pero esta vez, ya sin la forma de transporte eliminada.

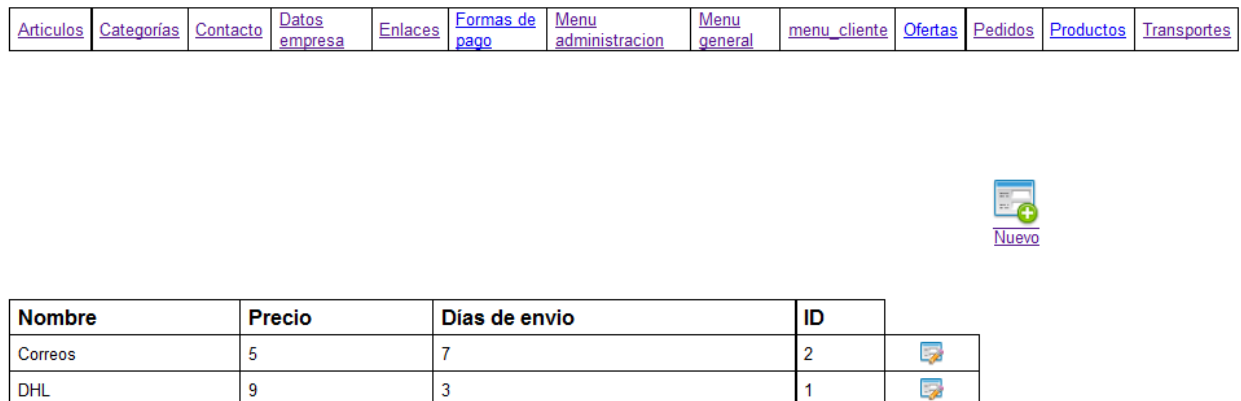


Figura 94: listado de las formas de transporte

Así pues, de esta forma, queda demostrado que funciona correctamente todo lo correspondiente a las formas de transporte.

Como he dicho anteriormente, el apartado de comprobación del correcto funcionamiento en la parte pública, lo realizaré en un juego de pruebas a parte, que será el juego de pruebas del proceso de compra. En este juego de pruebas, si las formas de transporte funcionan correctamente, nos tendrá que salir en un momento del proceso de compra, un listado con las posibles formas de transporte y elegir la deseada.

9.10 Productos

En este juego de pruebas, vamos a comprobar que la gestión de productos funciona correctamente. Este juego de pruebas va a consistir en la secuencia de varios pasos a realizar. Lo primero que tenemos que hacer, es irnos a la parte de administrador, y clicar sobre productos.

Articulos	Categorías	Contacto	Datos empresa	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos
---------------------------	----------------------------	--------------------------	-------------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------



[Nuevo](#)

ID	id_categoria	nombre	direccion_URL	descripcion	precio	stock	
32	9	Diadema atardecer	imagenes/productos/diademasinfantiles/foto11.jpg	Diadema de raso cereza, con flores de miga de pan color granate y lazo blanco roto	10	5	
13	8	Diadema butterfly	imagenes/productos/diademasinfantiles/foto9.jpg	Diadema de raso rojo, con pluma negra y brillantitos negros y cristales negros en forma de rombo	12	3	
24	9	Diadema canastera	imagenes/productos/diademasinfantiles/foto3.jpg	Diadema de raso marron con rafia y flor de pelo natural en tonos rubios	10	5	
22	9	diadema candy	imagenes/productos/diademasinfantiles/foto1.jpg	Diadema de raso rojo, con flor de pelo rojo, lazo blanco y brillante en el centro	10	2	
43	9	Diadema cebra rosa	imagenes/productos/diademasinfantiles/foto23.jpg	Diadema forrada con raso blanco roto y cinta fucsia con topes blancos	6	7	

Figura 95: listado de los productos que tenemos en la tienda en la parte de administración

Como podemos apreciar, si pulsamos en productos nos aparecerá una lista con todos los productos que tenemos guardados. También podemos apreciar, que hay una serie de id_categoria. Este identificador de categoría se lo asignamos dependiendo de la categoría a la que queramos que pertenezca el producto. Actualmente, tenemos las siguientes categorías:

- id_categoria 8: diademas
- id_categoria 9: diademas infantiles
- id_categoria 10: prendidos infantiles
- id_categoria 11: pasadores
- id_categoria 12: tocados

A continuación, vamos a crear un producto nuevo, y lo añadiremos a la categoría de tocados. Así pues, vamos a pulsar el botón de nuevo y rellenamos los campos necesarios.

Articulos	Categorias	Contacto	Datos empresa	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------



Registrar un producto nuevo

Id_categoria:

Nombre:

Direccion URL:

Descripcion:

Precio:

Stock:

Figura 96: registro de un producto nuevo

A continuación pulsamos guardar. Para comprobar si nos ha funcionado correctamente iremos a la parte pública y comprobaremos si nos aparece el tocado paradise, que es el tocado que acabamos de crear.



Figura 97: comprobación de la inserción del producto en la parte pública

Aquí podemos apreciar que aparece el tocado paradise. Ahora vamos a pulsar sobre este tocado, y nos aparecerán los datos y la foto en grande.

Ficha del producto

tocado paradise



Descripción:

Tocado de base rosa, con flores de sinamay rosas, lazo negro y plumas negras

Stock: 3 und.



Precio:

20€

Cantidad:

AÑADIR AL CARRITO 

Como podemos apreciar, el tocado se a añadido correctamente. Así pues, de la misma forma, las opciones de modificar y eliminar funcionan correctamente. Por lo tanto, podemos decir, que el juego de pruebas de los productos va a la perfección.

9.11 Carrito / Proceso de Compra

Ahora vamos a comprobar la funcionalidad del carrito de la compra. Tenemos dos opciones, que el usuario esté logueado o que el usuario no esté logueado. Primero vamos a ver la actuación del carrito en el caso de que el usuario no esté logueado. Primero, vamos a escoger un producto y añadirlo al carrito.

Ficha del producto

diadema candy



Descripción:

Diadema de raso rojo, con flor de pelo rojo, lazo blanco y brillante en el centro

Stock: 2 und.



Precio:

10€

Cantidad:

[AÑADIR AL CARRITO](#)

Figura 99: Ficha de un producto

Pulsamos añadir al carrito y esta página nos redirecciona a la página del carrito de la compra. En esta página tendremos tres opciones: vaciar carrito, seguir comprando o confirmar carrito.

VACIAR CARRITO SEGUIR COMPRANDO			
Producto	Precio	Cantidad	Total
DIADEMA CANDY	10€	- 1 +	10€ X
TOTAL			10 €
CONFIRMAR CARRITO			

Figura 100: Carrito de la compra

Nosotros pulsaremos añadir al carrito. Si todo funciona correctamente ahora nos pedirá que el cliente introduzca su usuario y su contraseña con el fin de poder tramitar la compra. Así pues, si no estamos registrados o logueados no podemos realizar una compra.

Usuario:

Contraseña

Enviar

Figura 101: login

Ahora vamos a comprobar el proceso de compra cuando un usuario está logueado. Voy a introducir los datos de un cliente y voy a añadir el mismo producto al carrito. De la misma forma, cuando tenga ese producto añadido pulsare a confirmar carrito. Si todo funciona correctamente, saldrá una pantalla con la dirección de envío y la dirección de facturación (esta última es opcional).

Dirección de envío:

Dirección:

Localidad:

Provincia:

Código Postal:

Dirección de facturación (opcional):

Dirección:

Localidad:

Provincia:

Código Postal:

siguiente

Figura 102: introducción de las direcciones

Como podemos ver hasta aquí funciona correctamente. Ahora vamos a pulsar siguiente y nos aparecen los diferentes tipos de transporte.

	Nombre	Precio	Días de envío
<input type="radio"/>	Correos	5	7
<input type="radio"/>	DHL	9	3

siguiente

Figura 103: Selección del tipo de transporte

Así pues, también podemos verificar, que las formas de transporte en la parte pública funcionan correctamente. A continuación seleccionamos la opción de correos y pulsamos siguiente. Nos deberá aparecer las diferentes formas de pago.

Nombre	Descripcion	Comision
<input checked="" type="radio"/> contra-rembolso	El pago se realiza en el momento que llega el paquete al domicilio	4
<input type="radio"/> Transferencia	pago mediante transferencia bancaria	2

[siguiente](#)

Figura 104: Selección de la forma de pago

De la misma manera, también se puede apreciar que las formas de pago en la parte pública funcionan correctamente. Seleccionamos la opción de contra-rembolso y pulsamos siguiente.

Resumen de la compra

Dirección de envío:

Dirección: **c/Baviera nº 13**

Localidad: **Maella**

Provincia: **Zaragoza**

Codigo Postal: **50710**

Dirección de facturación (opcional):

Dirección:

Localidad:

Provincia:

Codigo Postal:

Producto	Cantidad	Precio:	Total:
diadema candy	2	10	20

Transporte: Correos 5 Euros

Forma de pago: contra-rembolso Comision: 4

Total a pagar: 29

[Confirmar](#)

Figura 105: resumen de la compra

Ahora nos aparecé un resumen de la compra, con la dirección de envío, los productos, el tipo de transporte, la forma de pago, y el importe total. Así pues, pulsamos confirmar, para confirmar la compra.

El pedido se ha realizado correctamente, en breves lo recibira en su domicilio.

Figura 106: Compra finalizada

La compra queda confirmada y finaliza el proceso de compra.

9.12 Stock

En este juego de pruebas vamos a comprobar que se decrementa correctamente el stock de un producto al realizarse una compra. En el juego de pruebas anterior, hemos realizado la compra de una diadema infantil. Concretamente hemos realizado la compra de la diadema Candy. Si vemos la figura del detalle de la diadema en el juego de pruebas anterior, podremos observar que el stock de esa diadema era 2. Ahora, como hemos comprado una diadema, si el juego de pruebas funciona correctamente podremos observar que el stock de esa diadema se ha decrementado. Así pues, ahora el stock de la diadema tendrá que ser 1.

Ficha del producto

diadema candy



Descripción:

Diadema de raso rojo, con flor de pelo rojo, lazo blanco y brillante en el centro

Stock: 1 und.



Precio:

10€

Cantidad:

AÑADIR AL CARRITO 

Figura 107: ficha del producto

Así pues, queda comprobado que el stock se decrementa correctamente.

9.13 PEDIDOS

En este juego de pruebas vamos a comprobar que se guarda correctamente el historial de pedidos de cada usuario y de la misma manera los productos de cada pedido. Como tenemos la sesión iniciada de Sandra, pulsaremos sobre los la opción pedidos.



Figura 108: vista general de un cliente registrado

En este caso, este usuario sólo ha realizado un pedido, por lo que nos tendría que aparecer única y exclusivamente el pedido realizado.

INICIO	CLIENTE NUEVO	DONDE ESTAMOS	ENLACES	CONTACTO
DATOS CLIENTE	PEDIDOS			
<input type="text"/>	Buscar			
Flores de pelo				
DIADEMAS	DIADEMAS INFANTILES	PASADORES	PRENDIDOS INFANTILES	TOCADOS
fecha	env_direccion	estado		
2011-06-04	c/Baviera nº 13		Productos del pedido	
				Carrito
				ID AL CARRITO

Figura 109: historial de pedidos de un cliente

Como podemos apreciar nos aparece el pedido que hemos realizado, con la fecha y la dirección de envío. También nos aparece un campo estado vacío. Este campo estado lo tenemos que cambiar desde la parte de administración, y es el que indicará al cliente el estado del pedido (pendiente, en proceso, enviado, recibido..). Así pues vamos a irnos a la parte de administración a introducir ese campo.

Dentro del menú de administración pulsaremos la opción de pedido. Nos aparecerá un listado con todos los pedidos que tenemos.

Articulos	Categorías	Contacto	Datos empresa	Enlaces	Formas de pago	Menu administracion	Menu general	menu_cliente	Ofertas	Pedidos	Productos	Transportes
---------------------------	----------------------------	--------------------------	-------------------------------	-------------------------	--------------------------------	-------------------------------------	------------------------------	------------------------------	-------------------------	-------------------------	---------------------------	-----------------------------

id	id_carrito	estado	
61	47		
17	0	completado	
46	25	rechazado pedido	

Figura 110: listado de pedidos en la parte de administración

Editaremos el pedido con el identificador de carrito 47, y en el campo estado pondremos pendiente.



Modificar los datos del pedido

estado:

Figura 111: modificación del estado de un pedido

Guardamos el pedido, y volvemos a la parte pública de la tienda para comprobar que este cambio se ha realizado con éxito.

Flores de pelo			
DIADEMAS		DIADEMAS INFANTILES	
PASADORES		PRENDIDOS INFANTILES	
fecha	env_direccion	estado	
2011-06-04	c/Baviera nº 13	pendiente	Productos del pedido

Figura 112: historial de pedidos de un cliente

Podemos apreciar que el cambio se ha realizado satisfactoriamente ya que ahora en el campo de estado aparece pendiente. Como podemos ver, allado del campo estado, hay un link llamado productos del pedido. Si pulsamos en ese link, la web nos mostrará todos los productos comprados en ese producto. Se tiene que recordar que en ese pedido solo se compró la diadema Candy. Así pues, vamos a pulsar sobre el link para comprobar que este paso funciona correctamente.


DIADEMAS		DIADEMAS INFANTILES	
PASADORES		PRENDIDOS INFANTILE:	
nombre		descripcion	precio
diadema candy		Diadema de raso rojo, con flor de pelo rojo, lazo blanco y brillante en el centro	10€

Figura 113: Productos de un pedido de un cliente

Como podemos ver, nos muestra el producto del pedido correctamente, por lo que el proceso de los pedidos podemos afirmar que funciona correctamente.

BIBLIOGRAFÍA

La bibliografía que he utilizado para realizar este proyecto ha sido muy diversa. Me resultaría imposible nombrar todas las páginas y manuales online consultados, pero sí que puedo enumerar los manuales que más he utilizado:

- Manual de PHP: <http://php.net/manual/es/index.php>
- Manual de SQL: <http://www.desarrolloweb.com/manuales/9/>
- Manual de JavaScript: <http://www.manualdejavascript.com/>
- Manual de HTML: <http://www.uv.es/jac/guia/>
- También me resultó muy útil esta página para comprender el patrón del modelo vista controlador: <http://craftyman.net/mvc-en-php/>

CONCLUSIONES

Este proyecto ha sido un gran reto para mí, ya que anteriormente no había trabajado con herramientas de tipo Web pero era una cosa que me despertaba un gran interés. Tengo la impresión de que una sola persona puede hacer grandes cosas en el tema web y durante el proceso de creación se me han ido ocurriendo ideas para poder modificar y mejorar la página web, que antes de empezar ni siquiera se me habían ocurrido. También he descubierto el patrón modelo-vista-controlador, que antes de comenzar con este proyecto ni siquiera sabía que existía. Me parece muy interesante el hecho de que puedas cambiar toda la apariencia de una aplicación sin tocar nada interno, es decir, modificando solo las vistas.

TRABAJO PENDIENTE

Sin embargo, este proyecto no lo he podido finalizar como a mí me hubiera gustado debido a la falta de experiencia en este campo y al tiempo.

Me hubiera gustado mucho haber realizado la tienda virtual de manera internacional, haber añadido varios idiomas y conversores de monedas.

También me hubiera gustado poder añadir el pago mediante tarjeta de crédito y poder desarrollar la pasarela de pago virtual (TPV). Esto último lo voy a realizar una vez finalizada la carrera para poder acabar la tienda virtual en condiciones y sentirme realizada conmigo misma.

También me hubiera gustado poder realizar la parte de la administración con una interfaz más elaborada pero no he podido por falta de tiempo. Asimismo, prefería que todo funcionara correctamente antes de hacer una interficie más elaborada y que algunas de las funcionalidades no hagan la acción esperada.