



TIENDA ONLINE DE PITA PINTA

Admin. Web y Comercio electrónico

Autor: Sergio Martínez Montes

Consultor:

Francisco Javier Noguera Otero

Tutor:

Marcos Viñuela García

Octubre de 2013

Licencia GFDL

Se concede permiso para copiar, distribuir y / o modificar este documento bajo los términos de la licencia de documentación libre GNU, versión 1.2 o cualquier otra versión posterior publicada por la Free Software Foundation, sin secciones invariantes ni textos de cubierta delantera. Una copia de la licencia está incluida en la sección titulada GNU Free Documentation License.

RESUMEN

El proyecto consiste en la creación de un portal web para la venta de gallina de raza autóctona asturiana, pita pinta. El cliente, criador de esta raza, encarga esta web para dar a conocer su gallinero, ofrecer material relacionado con la cría y productos de alimentación, todo ello para su encargo y compra por internet, con dos dominios registrados: <http://pitaasturiana.com/> y <http://pintapinta.net>

Además de la web donde se publican los productos: animales, material y alimentación para su venta, y la parte donde se muestra el gallinero, otra parte importante del proyecto es el desarrollo del gestor para administrar todo ello, tanto los productos publicados como las ventas.

El desarrollo del gestor de administración (aplicación de gestión de contenidos) es personalizado a las necesidades del cliente y desde una fácil gestión el administrador del sitio puede cargar las fotografías utilizadas para todos los productos, añadir nuevos productos y gestionar las ventas, dando de alta nuevas comprar o editar las que provengan de usuarios del portal.

La tecnología utilizada para el desarrollo de este porta de comercio electrónicos es:

Servidor Apache en Linux, servidor de bases de datos MySQL, lenguaje de programación PHP, y tecnologías de cliente HTML5, CSS3 y jQuery (javascript).

Índice de contenidos

1	Introducción.....	5
2	Estudio de viabilidad.....	6
2.1	Alcance del sistema.....	6
2.1.1	Funcionales.....	6
2.1.2	Rendimiento.....	6
2.1.3	Seguridad.....	6
2.1.4	Disponibilidad.....	6
2.2	Estudio de la situación actual.....	6
2.3	Definición de los requisitos del sistema.....	8
2.3.1	Económicos.....	8
2.3.2	Técnicos.....	8
2.3.3	Legales.....	8
2.4	Estudio de las alternativas de la solución.....	9
2.4.1	Toldo.....	9
2.4.2	Prestashop.....	9
2.4.3	Aplicación a medida.....	9
2.5	Valoración de las alternativas.....	9
2.5.1	Toldo.....	9
2.5.2	Prestashop.....	9
2.5.3	Desarrollo a medida.....	10
2.6	Selección de la solución.....	10
3	Análisis.....	11
3.1	Definición del sistema.....	11
3.2	Establecimiento de requisitos.....	11
3.3	Definición de interfaces de usuario.....	11
3.3.1	Mapa de navegación.....	11
3.3.2	Wireframing.....	13
3.3.3	Diseño de layout.....	14
	23
4	Diseño.....	23
4.1	Diseño de la base de datos.....	23
5	Desarrollo.....	30
5.1	Planificación de las actividades de desarrollo.....	30
5.2	Desarrollo.....	31
5.3	Documentación.....	34
6	Implementación y mantenimiento.....	35
6.1	Niveles de servicios.....	35
6.2	Aceptación del sistema.....	35
6.3	Mantenimiento.....	35
7	Conclusiones.....	36
8	Bibliografía.....	38

1 Introducción

Los objetivos son varios y de varios tipos. El principal es lanzar a la red, al mundo digital una práctica del mundo rural con todo lo que ello conlleva. Proyectar y unir una actividad tradicional del sector primario con las nuevas tecnologías.

De esto nace el objetivo principal del desarrollo del portal, el cliente, criador de la raza de gallina autóctona tiene como objetivo, darla a conocer y con ello ayudar a su conservación. A la par que también poder generar un negocio entorno a ello, con la creación de un par de puestos de trabajo. Uno a nivel de gestión de web y otro dentro de las instalaciones al aumentar el criadero.

A nivel técnico se desarrollará un gestor a medida para que el mantenimiento del sistema sea lo más sencillo y ágil posible a la vez que atractivo. En lo que respecta a la web y teniendo en cuenta el posible público objetivo de las ventas su diseño y navegación será también fácil y muy usable.

Finalmente se establece un objetivo secundario, el posicionamiento del portal en buscadores. Como consecuencia del buen marcado el SEO de la web mejorará de forma natural. No se aplicará nada particular o se desarrollara o programará nada adicional para que la indexación en buscadores sea mejor, pero sí que se cuidarán los aspectos como el marcado semántico y la conexión con redes sociales para potenciar que todo el desarrollo del proyecto posicione lo mejor posible.

2 Estudio de viabilidad.

2.1 Alcance del sistema

El alcance del proyecto se centra y enfoca principalmente en la muestra de los productos, agrupados en tres tipos: Animales, Alimentación y Materiales, y en su venta por internet. Las necesidades del cliente se concretan en un sistema sencillo de utilizar tanto para ellos mismos para su administración como por el perfil (tecnológicamente medio-bajo) de sus potenciales usuarios. Se trabajará en buscar una solución donde el mantenimiento del portal tenga un coste bajo de aprendizaje si que por ello pierda usabilidad la tienda *online*.

Se tiene en cuenta también que dentro de este alcance hay que incluir la imagen de marca. El diseño de un logotipo para la web, *packaging* (empaquetado) y un futuro *merchandising* (comercialización).

Los requisitos del cliente:

2.1.1 Funcionales

- Un sistema fácil de mantener y gestionar. La herramienta de gestión de ventas y productos debe poder ser manejada por una persona con poca experiencia en Internet.
- Web comercial adaptada al tipo de usuario.
- Buen posicionamiento en buscadores. SEO

2.1.2 Rendimiento

- Se estiman de unas 50 a 100 ventas iniciales al mes. El tráfico no será elevado. No se hará ninguna campaña publicitaria en la red para promocionar el portal si el SEO funciona como se espera. Por ello en una primera instalación el hosting puede ser incluso hasta un servidor compartido.

2.1.3 Seguridad

- Se restringirá por IP el acceso al gestor de productos y ventas. La URL de acceso al gestor, sólo será alcanzable desde un número limitado de direcciones IP, como podrán ser la oficina y la casa del cliente.
- La seguridad del pago se implementará con la pasarela ofrecida por la entidad bancaria.

2.1.4 Disponibilidad

- La disponibilidad tiene que ser 24x7 como cualquier plataforma de comercio *online*. Al tratarse de una meramente nacional, los mantenimientos, modificaciones y actualizaciones del sistema pueden ser por las noches en horario de España.

2.2 Estudio de la situación actual

La situación tecnológica previa al desarrollo del portal *online* de Pita Pinta era muy precaria. Con una web totalmente obsoleta sobre información del criadero, realizaban todas las ventas desde la propia instalación a vecinos y amigos.

La ampliación de la nave del criadero y una apuesta por total por la raza autóctona han impulsado la idea de la venta por Internet. Desde el nuevo portal se mostraran los productos a la venta, tanto

animales, como alimentación o productos destinados a la cría de la gallina.

Se diseñará una nueva web que sustituirá a la actual que era totalmente estática y desfasada. Por ello los antecedentes no condicionarán este nuevo desarrollo. únicamente se tendrá que tener en cuenta los nuevos requisitos del sistema para alojar la nueva web comercial de venta *online*.

La poca experiencia previa en tecnologías de la información plantea el reto de la realización de un sistema de gestión de ventas y pedidos fácil de usar sencillo e intuitivo.

Sobre el estado del arte en la venta *online* hasta la fecha no existía ningún sitio web de gallinas, aunque se ha lanzado una recientemente (tugallinaonline.es – comentada a continuación) durante el desarrollo de esta documentación.

Existen muchos sitios web relacionado con la venta de productos de animales y su alimentación, pero muy pocos venden animales propiamente dichos. Dos de los principales están relacionados con animales exóticos:

La primera se llama: <http://www.almacenanimal.com/>, es una web que lleva al menos 5 años en funcionamiento, tienen una amplia variedad de animales de todo tipo, menos animales comunes como perros, gatos o como es nuestro caso gallinas. Su diseño es algo anticuado, pero a lo largo de estos años ha logrado un buen posicionamiento con por ejemplo “venta de animales *online*”

La segunda, <http://www.exzootica.com/>, es especialista en, arañas, anfibios, ofídios... tiene un diseño más actual pero su accesibilidad (debido al poco contraste entre menús y color de fondo) podría ser mejorada. También tiene un buen posicionamiento con la búsqueda anterior.

Sólo hemos encontrado un portal destinado a venta de aves:

<http://www.avicoladeseleccion.com/>

A su diseño puede pecar de falta de accesibilidad ya que la tipografía resulta algo pequeña en algunos apartados pero funcionalmente es muy completa. Dispone de una gran variedad de aves, gallinas, patos, ocas, perdices...

Estos portales o sitios web ya existían antes de la realización de este proyecto, pero hace muy poco tiempo (en las últimas semanas, lo demuestran los sólo 108 seguidores en facebook. El portal de pita pinta ya tiene más de 200 y aún no ha publicado la web) se ha lanzado un portal específico de venta de gallinas. No hemos encontrado un portal como el nuestro de venta de una raza de gallina pero sí de aves como el anterior o en este caso uno de venta *online* de gallinas.

<http://www.tugallinaonline.es/>

La estructura de este portal es muy similar al planteado en nuestro proyecto. Una web clara y sencilla y muy accesible para cualquier usuario, aunque en este caso la navegabilidad se podría mejorar porque se confunden las categorías (razas de las gallinas) y el menú desplegable con todas ellas se pierde con el scroll vertical.

Casualmente, los únicos portales que comercializan con gallinas son de Granada, esto puede favorecer las ventas en la parte norte de la península ya que uno de los costes añadidos que tiene para los clientes son los elevados gastos de envío para el transporte de animales.



Cabecera de la página de fans de Facebook

2.3 Definición de los requisitos del sistema

Teniendo esto como premisa se establecen los siguientes requisitos agrupados en sus correspondientes categorías:

2.3.1 Económicos

No existe requisitos económicos específicos. El sistema de venta *online* inicial es sencillo y además del coste inicial de diseño y desarrollo el cliente tiene que contar con el coste de alojamiento de la aplicación.

No se invertirá en un servidor dedicado al principio. Si las visitas a la web superan las esperadas se necesitará migrar la aplicación a un servidor más potente y contar con un mayor coste económico.

2.3.2 Técnicos

Técnicamente se tendrán en cuenta los siguientes puntos:

- Arquitectura de la información, modelo de navegación y experiencia de usuario.
- Diseño, imagen de marca y web.
- Maquetación HTML5, CSS3, Javascript.
- Programación PHP: pasarela de pago y gestor de productos y pedidos.
- Servidor Apache y base de datos MySQL

2.3.3 Legales

No existirá registro de usuarios en la plataforma (al menos en el sistema inicial planteado). Cada vez que un usuario realice una compra tendrá que introducir sus datos y la dirección de envío. Aún así el sistema almacenará datos personales en la información de las ventas y para poder generar

la factura si el cliente la demanda. Por ello se necesitará aplicar la LOPD, informar a los usuarios de sus derechos y registrar en la Agencia de Protección de Datos el fichero de ventas con la información personal de los clientes. También habrá que redactar un aviso legal y unas condiciones de uso.

2.4 Estudio de las alternativas de la solución

Se van a valorar 3 opciones técnicas para la realización del proyecto, una tienda virtual.

1. La web: <http://tol.do/>
2. La solución de código abierto: <http://www.prestashop.com/es/>
3. Desarrollo de una aplicación a medida

2.4.1 Toldo

Según su propia web Toldo es: “Toldo es la manera más fácil de crear una web. En Toldo no necesitas tener conocimientos de programación, puedes diseñar una tienda bonita y fácil de usar a golpe de clic. También podrás gestionar el stock y controlar las ventas que hagas. En las tiendas Toldo es fácil vender y también es fácil comprar. Nosotros te damos el alojamiento en internet y las herramientas necesarias para tener tu tienda *online* en orden”

2.4.2 Prestashop

Prestashop es un producto de código abierto, lanzado en 2007, traducido a 56 idiomas y usado por más de 150.000 tiendas en todo el mundo. Su modelo de negocio se basa en la venta de módulos y ampliaciones a la instalación básica. Es una de las herramientas más usadas para desarrollar una tienda *online*, con una multitud de funcionalidades y muchas formas de pago diferentes ya implementadas.

El software ofrece más de 310 funciones integradas, incluyendo la gestión ilimitada inventario de productos, la navegación por capas, plantillas totalmente personalizables, análisis y presentación de informes, reglas de la promoción, gestión multi-tienda, monedas ilimitadas y gestión del retorno...

2.4.3 Aplicación a medida

Se plantea la posibilidad de realizar un proyecto a medida. Desarrollar una aplicación web de tienda virtual desde cero. El objetivo será la mayor calidad y control del mercado final, con el SEO como premisa y que la parte de administración de la herramienta sea lo más sencilla posible y que se adapte a las necesidades.

2.5 Valoración de las alternativas

2.5.1 Toldo

Es una herramienta muy cerrada. Las ventajas es la rapidez con la que puedes poner en marcha la tienda, pero como los plazos ajustados no son un requisito, existe tiempo para poder optar por un desarrollo. Los inconvenientes son estar atados a una herramienta tan cerrada, con un pago continuo (muy habitual en los modelos de negocio actuales, pagar por el servicio consumido) con unas limitaciones marcadas por las decisiones de la propia plataforma y poco margen para otras muchas secciones del portal web que se pudieran ir incorporando como un canal de gastronomía.

2.5.2 Prestashop

La principal ventaja de aprovechar el desarrollo de un producto existente es que el cliente, llegado el caso, en el futuro podría contar con otro proveedor de desarrollo que pudiera continuar y evolucionar el portal *online* si se veía atado, en cierta manera, a un desarrollo a medida del

proveedor inicial.

Prestashop es una herramienta a la que se le pueden añadir módulos, muchos de ellos de pago, y temas, plantillas de diseño web.

Módulos de pago

El lugar para adquirirlos es: <http://addons.prestashop.com/es/> y se organizan

Algunos de ellos son:

- *Módulo TPV Banco Sabadell*: Este módulo permite configurar y utilizar de manera muy sencilla un TPV basado en el sistema del Banco Sabadell.

<http://addons.prestashop.com/es/pagos-prestashop-modulos/6971-tpv-virtual-banco-sabadell.html>

- *Módulo abonos de factura*: Con este módulo puedes crear abonos de factura para aquellos pedidos que luego son cancelados y que ya están facturados.

<http://addons.prestashop.com/es/facturacion-contabilidad-modulos-prestashop/6414-abonos-de-factura.html>

- *Módulo Facebook PS Shop Tab*: Este módulo instalará una pestaña “Shop” en la página de fans de Facebook. Aparecerá una minitienda que sugerirá los productos directamente desde la red social.

<http://addons.prestashop.com/es/redes-sociales-facebook-modulos-prestashop/1048-facebook-ps-shop-tab.html>

Módulos gratuitos

La mayoría son desarrollados por personas independientes, pero desde la propia web se pueden descargar algunos desarrollados por entidades asociadas. Se pueden descargar desde aquí:

<http://www.prestashop.com/es/modulo-gratis-de-asociados> y algunos de ellos son:

- *Twenga*: motor de búsqueda de compras.
- *Hipay*: Módulo para el desembolso de transacciones.
- *Stripe*: Módulo para facilitar la aceptación de todo tipo de tarjetas de crédito.

2.5.3 Desarrollo a medida

Continuando con el punto anterior, al realizarse en PHP, de los lenguajes de servidor más utilizados sería fácil encontrar tanto programadores como otros proveedores con los que el cliente pudiera evolucionar su producto.

2.6 Selección de la solución

Finalmente se opta por una solución a medida. Los aspectos que se ha tenido en cuenta para ellos han sido.

- Especificaciones claras y sencillas para poder acometer un desarrollo a medida donde las diferencias del coste en adaptar un producto existente no marca una diferencia sustancial.
- Posibilidad de generar de manera propia las plantillas HTML para optimizar el marcado y prepararlo para SEO.
- Desarrollar un gestor para los productos y las ventas usable, sencillo y adaptado a las necesidades.
- La total versatilidad de evolución al tener el control de todo el desarrollo.

3 Análisis

3.1 Definición del sistema

Selección de servidor Apache, base de datos MySQL y lenguaje de programación PHP.

El proyecto web contará con dos partes bien diferenciadas por un lado la web de acceso público y por otro la aplicación web de gestión.

3.2 Establecimiento de requisitos

La web pública (portal web) de venta a su vez tendrá dos partes diferenciadas, por una parte contará con una descripción del criador y gallinero (una descripción más corporativa) y por otro lado la tienda propiamente dicha. La tienda tendrá una navegación sencilla, desde la página de inicio se accederá a un resumen de la mayoría de las categorías, desde el enlace a tienda virtual también de primer nivel como el índice se listarán todas ellas. Ya desde cada categorías se podrá ver un listado de productos con un enlace en cada uno de ellos a su detalle. En la siguiente sección se muestra un detalle del mapa de navegación de la tienda.

Por otro lado la web de administración que tendrá 3 apartados: productos, imágenes y ventas. De esta forma y con una estructura también sencilla desde el gestor se podrá manejar que productos que con información aparecen en la web y el estado de las ventas así como si se considera oportuno dar de alta nuevas o modificar y añadir productos a las que los usuarios hayan hecho por internet o incluso por otra vía como mail o teléfono.

3.3 Definición de interfaces de usuario

3.3.1 Mapa de navegación

El proyecto se divide en dos partes. Portal o sitio web y el gestor de contenidos y ventas.

El sitio web cuenta con 9 plantillas principales o nodos de navegación. Cinco de ellos en el primer nivel: Inicio, quienes somos, nuestro criadero, tienda online y contacto. Por debajo se situaría la plantilla de categoría de producto (accesible desde el primer nivel en tienda online), esta da acceso al detalle aunque desde el primer nivel, la página de inicio también se puede acceder al detalle. Al igual que la cesta de la compra accesible desde el primer nivel o desde cualquiera. La finalización de la compra sólo es accesible desde la cesta de la compra.

Las nueve plantillas principales del sitio son:

1. *Inicio*

Portada de la web con un resumen de los productos en oferta y las novedades y los más destacados de las categorías principales. También se podrá acceder a otras partes del portal, como redes sociales o la futura parte de gastronomía.

2. *Quiénes somos*

Imágenes del criador y explicación de su experiencia y currículum

3. *Nuestro criadero*

Textos explicativos sobre: orígenes de la raza, características de la pita pinta, incubación, instalaciones e imágenes del gallinero.

4. *Tienda online*

Portada de la tienda, listado de todas las categorías.

5. *Contacto*

Formulario de contacto.

6. *Categoría de producto*

Listado de los productos de una categoría seleccionada.

7. *Detalle de producto*

Plantilla para la información de un producto. Descripción, imagen, precio, selección de número de unidades y botón de compra.

8. *Cesta de la compra*

Listado de la compra con la opción de modificar las unidades de cada producto, eliminarlos o finalizar la compra.

9. *Datos del comprador*

Plantilla donde el usuario finalizará la compra introduciendo sus datos y eligiendo un método de pago.

Gestor de contenidos y ventas

El gestor de contenidos servirá para mantener los productos de la tienda y gestionar los pedidos. Para ello se han desarrollado las siguientes plantillas o pantallas:

1. *Listado de imágenes.*

Existirá un repositorio donde se puedan ver todas las imágenes que posteriormente se puedan vincular o asociar a los productos. Desde este listado se podrán eliminar o enlazar para su edición.

2. *Nueva imagen / Modificar imagen*

Plantilla para subir una nueva imagen al gestor o editar una existente.

3. *Listado de productos.*

Relación de todos los productos dados de alta en el gestor, sólo los que tengan el estado publicado aparecerán en el sitio web.

4. *Nuevo producto / modificar producto*

Pantalla en el gestor para añadir un nuevo producto, la misma plantilla servirá para editar uno.

5. *Listado de ventas (pedidos)*

Todas las ventas o pedidos (ventas aún no pagadas por elegir transferencia como método de pago) que se hagan desde el portal web tendrán su reflejo en este listado.

6. *Edición de ventas*

Cada venta del listado podrá ser editada desde esta plantilla. En casi todos los casos será para cambiar el estado a pagada, devuelta, enviada... Se podrá ver los datos del comprador así como también modificar las ventas, añadiendo nuevos productos, eliminando o cambiando los productos.

Esta misma plantilla se puede utilizar para generar una nueva venta, por ejemplo si se recibe un pedido por teléfono o en las mismas instalaciones, de esta manera todo estará centralizado en la herramienta de gestión.

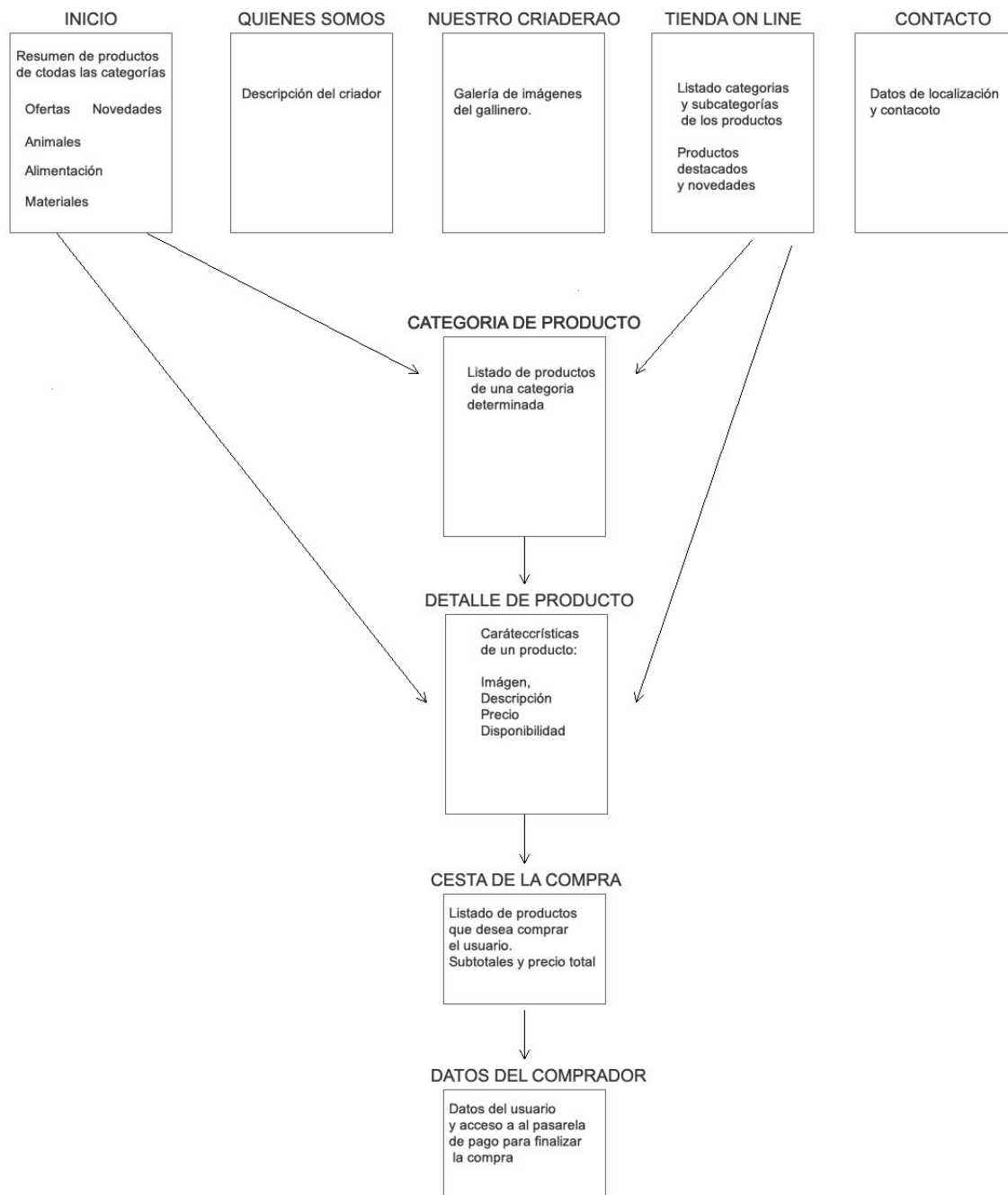
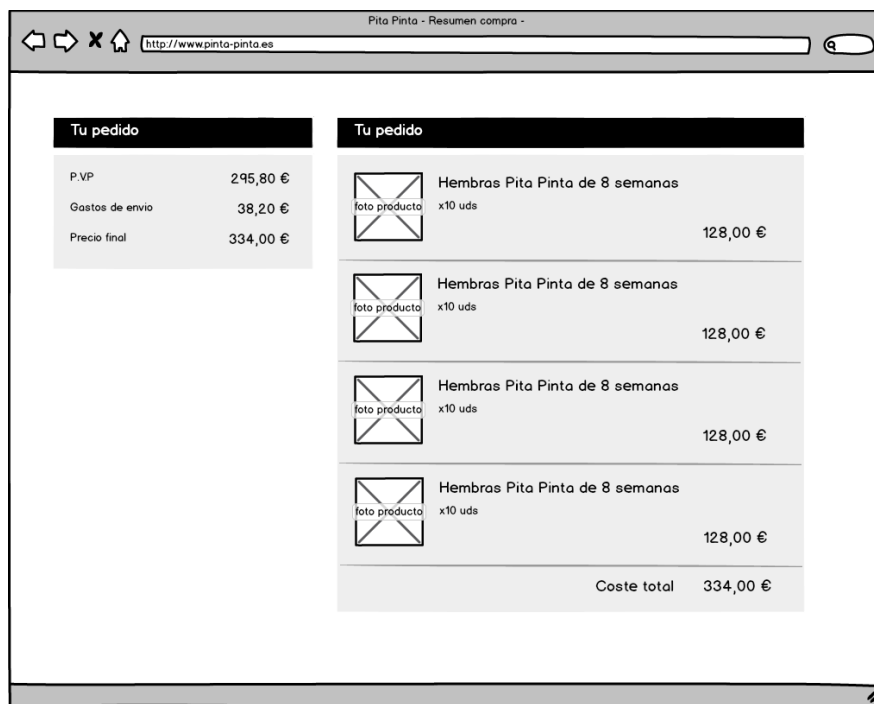


Diagrama del mapa de navegación del portal web

3.3.2 Wireframing

Diseño de la disposición de la información de cada plantilla.

Para cada plantilla se marcan los bloques con la información que mostrarán. Este es un paso previo al diseño, es el análisis para poder ordenar la información en un interfaz de usuario. A continuación se muestra un ejemplo del wireframe de una pantalla de interfaz dada:



Ejemplo de wireframing: Cesta de la compra

3.3.3 Diseño de layout

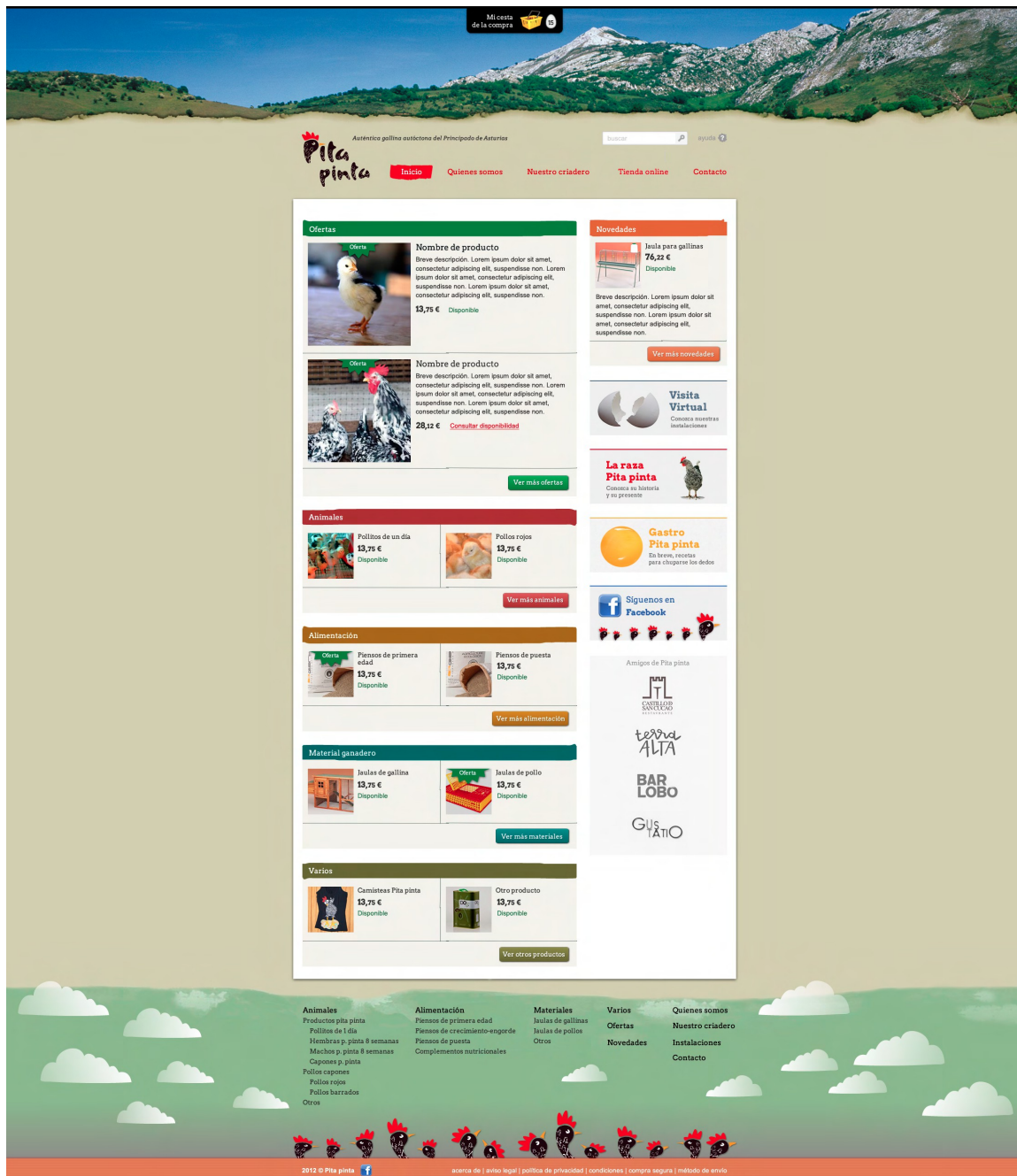
En la siguiente imagen se muestra el diseño de la página principal. En la cabecera podemos encontrar el enlace a la cesta de la compra, siempre será accesible desde todas las páginas. Por debajo de ella, se encuentra el logotipo y los enlaces a las secciones de primer nivel: Inicio, Quienes somos, Nuestro criadero, Tienda y Contacto. Por debajo el contenido principal dividido en dos columnas. La parte de la derecha con el producto novedad, los banners, con acceso a socios o amigos, redes sociales o el futuro canal gastronómico. Y la parte de la izquierda con los productos en oferta y más destacados. Por debajo el pie que a su vez hace de mapa web. En resumen y como ya se ha mencionado, un diseño claro y muy usable.

Sobre el **gestor** de contenidos se ha puesto especial empeño en que el diseño sea muy claro y como se está reiterando en esta documentación lo más sencillo de usar. Tres botones de menú principal para cada una de las partes de gestión: productos, imágenes y ventas. En cada uno de ellos un listado y la opción de ir o a una pantalla de edición seleccionando un elemento o pinchando en el botón, crear un nuevo.

El **listado de productos** del gestor, como el resto de listados lo encabeza un formulario para poder filtrar por algunos campos

La **pantalla de una venta** es algo diferente. Si se edita una se puede ver como en la parte superior aparecerán los datos del cliente y la parte inferior se divide en dos: a la derecha un buscador de productos para si se desea poder añadir más productos a la derecha donde aparecerán los que realmente hay en esa venta, con la cantidad de cada uno de ellos y el precio total.

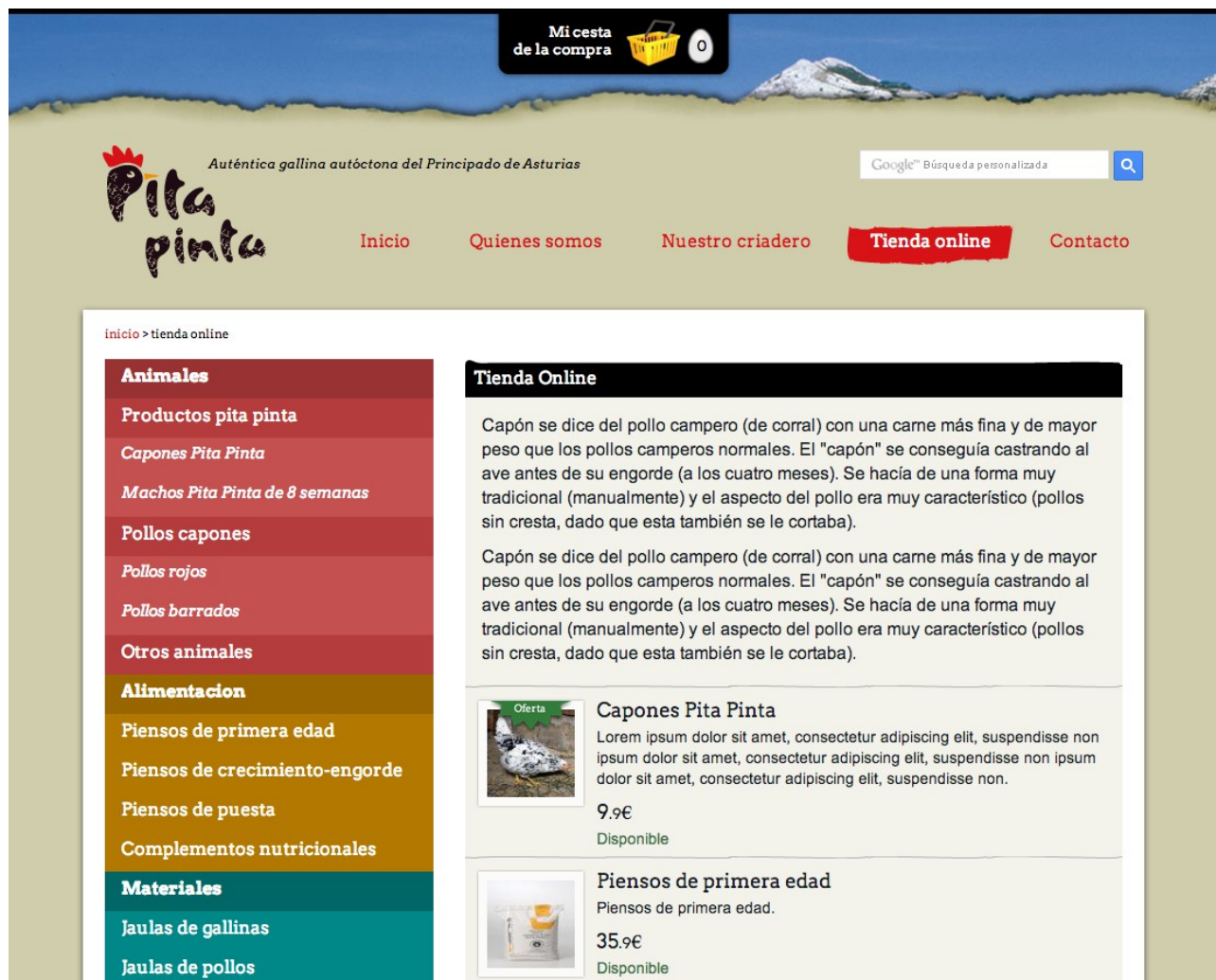
A continuación se muestran unas pantallas tanto del diseño del portal como del gestor de contenidos.



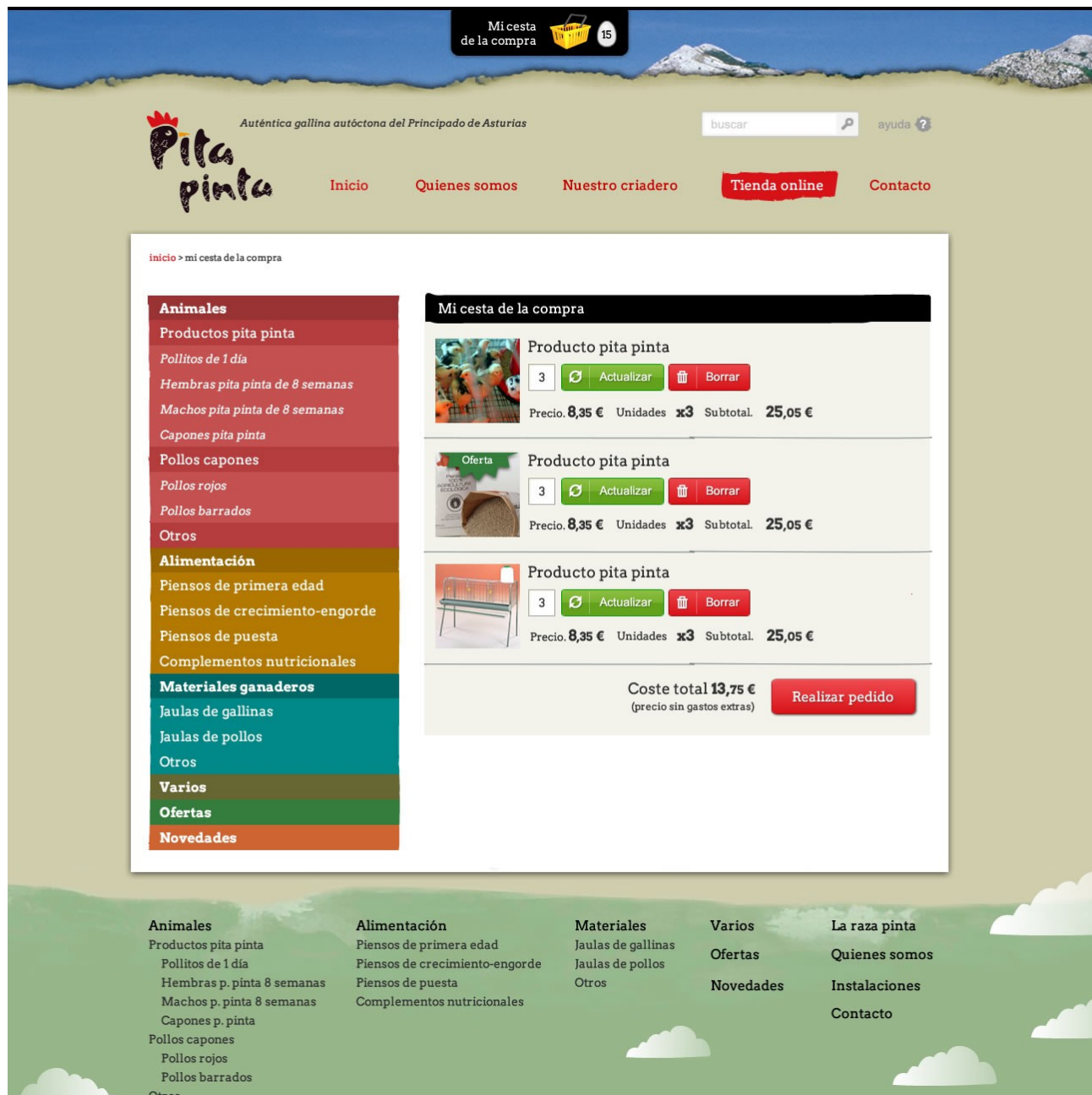
Diseño de la portada del portal




Las páginas de la sección de *Tienda online* mantienen un menú a la izquierda con las categorías y subcategorías. Como en un principio en número de productos no será muy elevado también se visualiza en el menú los productos de la categoría de Animales. De una manera muy visual se puede distinguir cada categoría en el menú.



Diseño de la página tienda online




Diseño de la cesta de la compra



Productos

Listado de ventas

Listado de imágenes



Administrador
 Gestión Pita Pinta
 Cerrar sesión

Listado productos

Nuevo producto

Nombre

Descripción

Categoría
 Seleccione ...

Subcategoría
 Seleccione categoría

☐ Disponible ☐ No disponible

☐ Portada ☐ Ofertas ☐ Novedades

Buscar

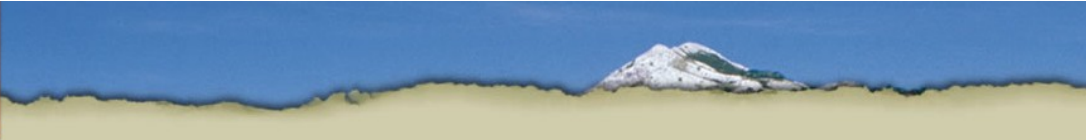

Nombre	Disponible	Oferta	Novedad	Categoría	Subcategoría	Precio	Eliminar
Capones Pita Pinta	✓	✓	✗	Animales	Productos pita pinta	9.9€	<input type="checkbox"/>
Pollitos de 1 día	✗	✗	✓	Animales	Productos pita pinta	8.5€	<input type="checkbox"/>
Machos Pita Pinta de 8 semanas	✓	✗	✗	Animales	Productos pita pinta	15.5€	<input type="checkbox"/>
Pollos rojos	✓	✗	✗	Animales	Pollos capones	7.35€	<input type="checkbox"/>
Pollos barrados	✓	✗	✗	Animales	Pollos capones	15.5€	<input type="checkbox"/>
Pensos de primera edad	✓	✗	✗	Alimentacion	No Aplicable	35.9€	<input type="checkbox"/>
Pensos de crecimiento-engorde	✓	✗	✗	Alimentacion	No Aplicable	24.85€	<input type="checkbox"/>
Pensos de puesta.	✓	✗	✗	Alimentacion	No Aplicable	12.3€	<input type="checkbox"/>
Complementos Nutricionales.	✓	✗	✗	Alimentacion	No Aplicable	12.8€	<input type="checkbox"/>
borrame	✗	✗	✗	Alimentacion	Complementos nutricionales	10€	<input type="checkbox"/>

Número de registros: 13


páginas 1 de 2
 siguiente »

 Ir a página

Diseño del listado de productos del gestor



[Productos](#)[Listado de ventas](#)[Listado de imágenes](#)

**Administrador**
Gestión Pita Pinta
[Cerrar sesión](#)

Pollitos de 1 día [Modificar](#) [Regresar al listado](#)

Datos del producto
Nombre

Descripción

Lorem ipsum dolor sit amet, consectetur adipiscing elit, suspendisse non ipsum dolor sit amet, consectetur adipiscing elit, suspendisse non ipsum dolor sit amet, consectetur adipiscing elit, suspendisse non.

Precio





☐ Disponible ☐ En oferta ☒ Novedad ☒ Portada
Categoría

Subcategoría


Imágenes
Descripción

Descriptor



[Buscar](#)

Nombre	Descriptor	Imagen	Asociar
JAULAAVESEXÓTICAS.jpg	material, aves exóticas		+
Jaula exposición.JPG	material, exposición		+
JAULACONEJOS.jpg	material, conejos		+
JAULA CODORNIZ.jpg	material, codorniz		+



Número de registros: 22 páginas 1 de 6 [siguiente »](#)

Imagen asociada al producto


Diseño de la plantilla de creación/edición de productos



[Productos](#)[Listado de ventas](#)[Listado de imágenes](#)


**Administrador**
Gestión Pita Pinta
[Cerrar sesión](#) 

Listado de ventas

[Nueva venta](#)


Fecha desde

Nombre

Estado de la venta
Seleccione ... 

Fecha hasta

Apellidos






Forma de envío
Seleccione ... 

Nº de venta

D.N.I.

Estado del envío
☐ Solo no enviados ☐ Solo enviados

Buscar

Nº Venta	Fecha	Nombre	Apellidos	Estado Pago	Forma Pago	Importe	Eliminar
51	04-10-2012	Sergio	sadfsadf		TPV - Tarjeta	105.45€	<input type="checkbox"/>
50	11-09-2012	adsfa	adsf		TPV - Tarjeta	175.1€	<input type="checkbox"/>
46	19-04-2012	Oscar	González González		TPV - Tarjeta	3€	<input type="checkbox"/>
47	19-04-2012	Marcos	Morilla		PayPal	164.9€	<input type="checkbox"/>
49	19-04-2012	Luis	Garca Campos		Transferencia	102.3€	<input type="checkbox"/>

Número de registros: 5

páginas 1 de 1

Ir a página

[Ir](#)

Diseño de la plantilla del listado de ventas

Diseño de la pantalla de detalle de una venta

4 Diseño

4.1 Diseño de la base de datos

El proyecto al ser un desarrollo a medida, no se ha basado en ningún software previo (drupal, wordpress, joomla...) que provea la estructura de la base de datos. Por lo tanto una parte más del proyecto es la definición de las tablas y sus relaciones.

Los diferentes productos, la mayoría diferentes clases de pollos y gallinas se almacenan en la tabla 01. La variedad de tipos de productos, en un principio no serán muchas (animales, alimentación, materiales y varios) se almacenarán en la tabla 02. Las diferentes variedades de estos 4 tipos: pollos, piensos... estará reflejados en la tabla 03. Los datos de cada venta con la información del usuario comprador se guardará en la tabla 04 pero la lista la compra estará en la tabla 05 donde se relaciona la 04 con la 01 con el número de unidades que se ha comprado de un determinado producto.

Se destinará una tabla para guardar las imágenes de los productos (07) y otras para completar la información necesario para una venta, provincias (06) estado de la venta (041) formas de envío (042) formas de pago (043)

La estructura de la base de datos tendrá las siguientes tablas:

- 01_productos

Esta tabla almacenará todos los productos que se den de alta en el gestor. Cuando se especifique en precio este producto se visualizará en la web, mientras el precio sea NULL o cuando se fije de nuevo a null el producto no aparecerá en la web. Existen varios campos para manejar su posición en la web. Portada (si se desea que aparezca en el index), novedad, oferta. El resto de los campos son los siguientes:

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id01_productos	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
02_categorias_id02_categorias	INT(11)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
03_subcategorias_id03_subcategorias	VARCHAR(4)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
07_imagenes_id07_imagenes	INT(11)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
nombre	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
descripcion	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
disponible	TINYINT(4)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
oferta	TINYINT(4)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
novedad	TINYINT(4)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
portada	TINYINT(2)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
precio	FLOAT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

- 02_categorias

Cada producto estará asociado a una de las categorías que se fijen en esta tabla, en principio cuatro: Animales, Alimentación, Materiales y otros.

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id02_categorias	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
nombre	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

- 03_subcategorias

Cada una de las anteriores categorías se podrán dividir en subcategorías. En el lanzamiento del portal la subcategorías a las que pertenezcan los primeros productos de la tienda serán:

Animales: Pita pinta y pollos capones.

Alimentación: Piensos de primera edad, piensos de crecimiento, piensos de puesta, complementos nutricionales.

Materiales: Jaulas de gallinas, Jaulas de pollos, otros.

Los campos para esta tabla son:

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id03_subcategorias	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
02_categorias_id02_categorias	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
nombre	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

- 04_ventas

En el lanzamiento del proyecto no se contempla el registro de usuarios, quizá sea una posible evolución pero al principio cada usuario que realice una compra tendrá que introducir sus datos por eso la tabla de ventas tiene los campos necesarios para poder identificar a la persona que ha realizado la compra. Es cierto que los mismos datos de la persona se pueden repetir se realiza muchas compras.

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id04_ventas	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
id_generado	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
06_provincias_id06_provincias	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
041_estadoVenta_id041_estadoVenta	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
042_formasEnvio_id042_formasEnvio	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
nombre_cliente	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
apellidos_cliente	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
dni_cliente	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
mail_cliente	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
direccion_cliente	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
telefono_cliente	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
forma_de_pago	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
codigo_postal	VARCHAR(5)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
codigo_transferencia	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
precio_total	FLOAT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
iva	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
fecha	TIMESTAMP	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

- 05_ventas_productos

Esta tabla es la que relaciona los productos con las ventas, donde se indica cuales se han seleccionado para la compra, que precio tenían en el momento de comprarse y cuantas unidades de cada uno se han elegido.

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id05_ventas_productos	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
precio	FLOAT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
cantidad	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
04_ventas_id04_ventas	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
01_productos_id01_productos	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

- 06_provincias

Tabla con las provincias, sólo se venderá a España.

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id06_provincias	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
nombre	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

- 07_imagenes

Tabla donde se referencia todas las imágenes que se suban al gestor. El campo nombre especifica el nombre del fichero y el campo ruta será la secuencia de directorios donde se guardará la imagen en el servidor. El fichero se subirá a una carpeta que indique el año y otra interior el mes, como por ejemplo 2013/12. El resto de campos para esta tabla son:

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id07_imagenes	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
id_generado	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
nombre	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
ruta	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
descripcion	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
descripcion_larga	VARCHAR(250)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
descriptor	VARCHAR(150)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
fecha_creacion	TIMESTAMP	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
fecha_modificacion	TIMESTAMP	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

- 041_estadoVenta

Una venta, podrá estar en diferentes estados: pagada, si se hace directamente desde el tpv, pendiente si se ha elegido transferencia bancaria o devuelta si finalmente son se realiza.

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id041_estadoVenta	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
nombre	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

- 042_formasEnvio

Se buscará una empresa para realizar los envíos y en principio las formas serán dos mediante la compañía de mensajería o recogida en las instalaciones para los clientes de la zona del criadero que quieran acercarse a por su pedido.

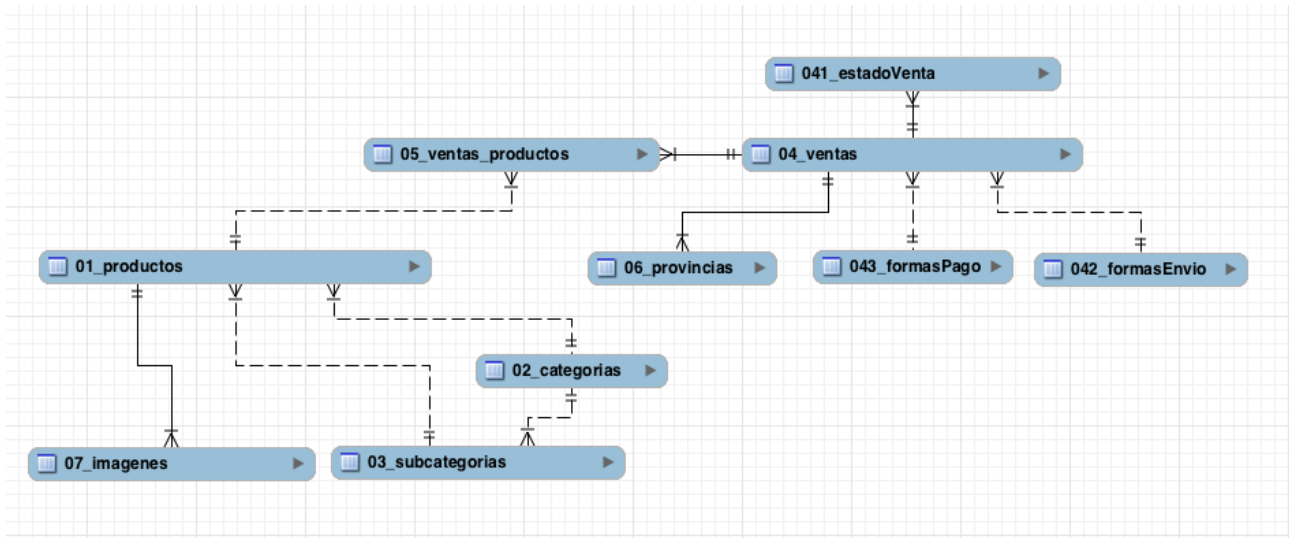
Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id042_formasEnvio	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
nombre	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
precio	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

- 043_formasPago

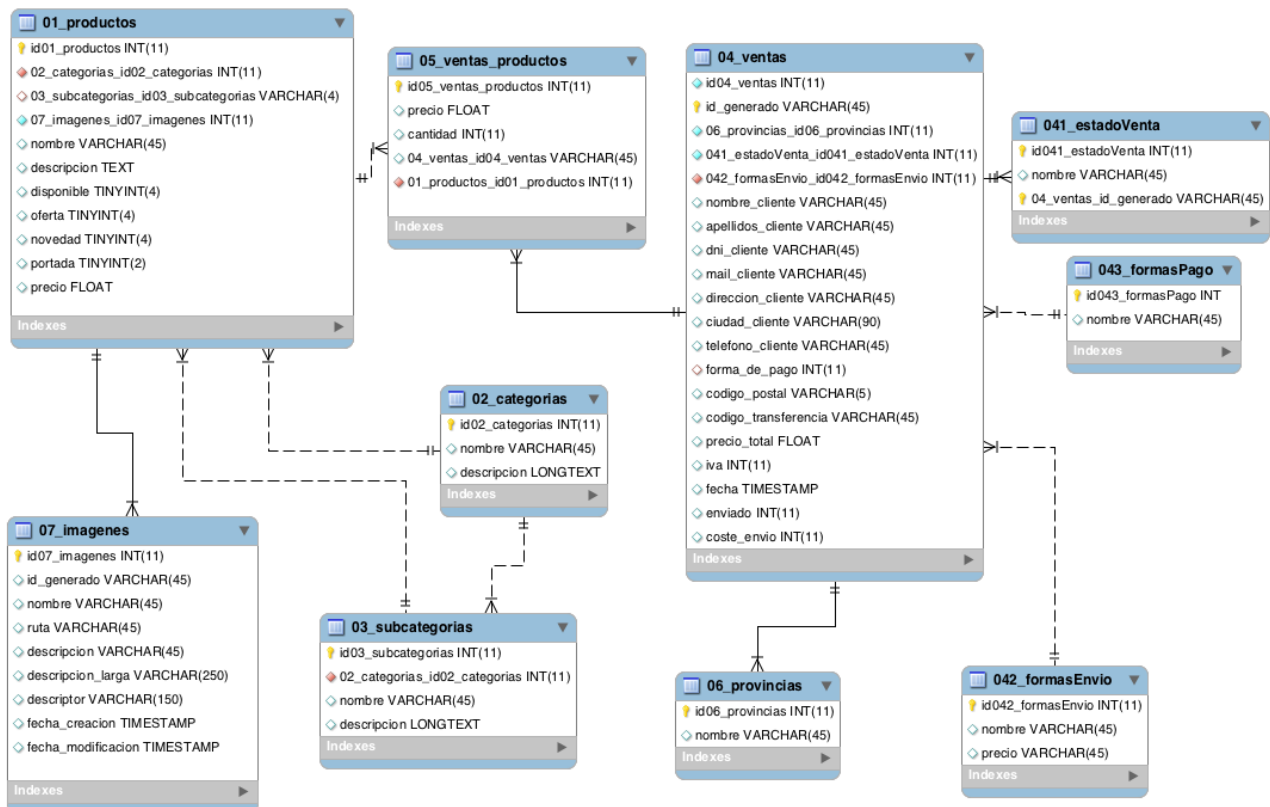
También serán dos, TPV y transferencia, estas dos se fijarán en esta tabla:

Column	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id043_formasPago	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
nombre	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

La relaciones de todas estas tablas se puede ver en el siguiente diagrama:



A continuación, con todos sus atributos, se muestra el diagrama completo entidad-relación de la base de datos:



Relaciones de entre las tablas de la base de datos del portal

A continuación se lista el código SQL necesario para la creación de la base de datos de la aplicación.

```

-- phpMyAdmin SQL Dump
-- version 3.5.0
-- http://www.phpmyadmin.net
--
-- Servidor: localhost
-- Tiempo de generación: 07-12-2013 a las 13:44:47

```

```

-- Versión del servidor: 5.5.20
-- Versión de PHP: 5.3.13

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";

--
-- Base de datos: `pitapinta_db`
--
--
--
-- Estructura de tabla para la tabla `01_productos`
--
CREATE TABLE IF NOT EXISTS `01_productos` (
  `id01_productos` int(11) NOT NULL AUTO_INCREMENT,
  `02_categorias_id02_categorias` int(11) NOT NULL,
  `03_subcategorias_id03_subcategorias` varchar(4) DEFAULT NULL,
  `07_imagenes_id07_imagenes` int(11) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL,
  `descripcion` text,
  `disponible` tinyint(4) DEFAULT NULL,
  `oferta` tinyint(4) DEFAULT NULL,
  `novedad` tinyint(4) DEFAULT NULL,
  `portada` tinyint(2) DEFAULT NULL,
  `precio` float DEFAULT NULL,
  PRIMARY KEY (`id01_productos`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=28 ;

--
-- Estructura de tabla para la tabla `02_categorias`
--
CREATE TABLE IF NOT EXISTS `02_categorias` (
  `id02_categorias` int(11) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL,
  `descripcion` longtext,
  PRIMARY KEY (`id02_categorias`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Estructura de tabla para la tabla `03_subcategorias`
--
CREATE TABLE IF NOT EXISTS `03_subcategorias` (
  `id03_subcategorias` int(11) NOT NULL,
  `02_categorias_id02_categorias` int(11) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL,
  `descripcion` longtext,
  PRIMARY KEY (`id03_subcategorias`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Estructura de tabla para la tabla `04_ventas`
--
CREATE TABLE IF NOT EXISTS `04_ventas` (
  `id04_ventas` int(11) NOT NULL AUTO_INCREMENT,
  `id_generado` varchar(45) NOT NULL DEFAULT '',
  `06_provincias_id06_provincias` int(11) NOT NULL,
  `041_estadoVenta_id041_estadoVenta` int(11) NOT NULL,
  `042_formasEnvio_id042_formasEnvio` int(11) NOT NULL,
  `nombre_cliente` varchar(45) DEFAULT NULL,
  `apellidos_cliente` varchar(45) DEFAULT NULL,
  `dni_cliente` varchar(45) DEFAULT NULL,
  `mail_cliente` varchar(45) DEFAULT NULL,
  `direccion_cliente` varchar(45) DEFAULT NULL,
  `ciudad_cliente` varchar(90) DEFAULT NULL,

```

```

`telefono_cliente` varchar(45) DEFAULT NULL,
`forma_de_pago` int(11) DEFAULT NULL,
`codigo_postal` varchar(5) DEFAULT NULL,
`codigo_transferencia` varchar(45) DEFAULT NULL,
`precio_total` float DEFAULT NULL,
`iva` int(11) DEFAULT NULL,
`fecha` timestamp NULL DEFAULT NULL,
`enviado` int(11) DEFAULT NULL,
`coste_envio` int(11) DEFAULT NULL,
PRIMARY KEY (`id_generado`),
UNIQUE KEY `id04_ventas` (`id04_ventas`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=52 ;

-----

--
-- Estructura de tabla para la tabla `05_ventas_productos`
--

CREATE TABLE IF NOT EXISTS `05_ventas_productos` (
  `id05_ventas_productos` int(11) NOT NULL AUTO_INCREMENT,
  `precio` float DEFAULT NULL,
  `cantidad` int(11) DEFAULT NULL,
  `04_ventas_id04_ventas` varchar(45) DEFAULT NULL,
  `01_productos_id01_productos` int(11) NOT NULL,
  PRIMARY KEY (`id05_ventas_productos`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=732 ;

-----

--
-- Estructura de tabla para la tabla `06_provincias`
--

CREATE TABLE IF NOT EXISTS `06_provincias` (
  `id06_provincias` int(11) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`id06_provincias`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-----

--
-- Estructura de tabla para la tabla `07_imagenes`
--

CREATE TABLE IF NOT EXISTS `07_imagenes` (
  `id07_imagenes` int(11) NOT NULL AUTO_INCREMENT,
  `id_generado` varchar(45) DEFAULT NULL,
  `nombre` varchar(45) DEFAULT NULL,
  `ruta` varchar(45) DEFAULT NULL,
  `descripcion` varchar(45) DEFAULT NULL,
  `descripcion_larga` varchar(250) DEFAULT NULL,
  `descriptor` varchar(150) DEFAULT NULL,
  `fecha_creacion` timestamp NULL DEFAULT NULL,
  `fecha_modificacion` timestamp NULL DEFAULT NULL,
  PRIMARY KEY (`id07_imagenes`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=34 ;

-----

--
-- Estructura de tabla para la tabla `041_estadoVenta`
--

CREATE TABLE IF NOT EXISTS `041_estadoVenta` (
  `id041_estadoVenta` int(11) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`id041_estadoVenta`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-----

--
-- Estructura de tabla para la tabla `042_formasEnvio`
--

```

```
CREATE TABLE IF NOT EXISTS `042_formasEnvio` (  
  `id042_formasEnvio` int(11) NOT NULL,  
  `nombre` varchar(45) DEFAULT NULL,  
  `precio` varchar(45) DEFAULT NULL,  
  PRIMARY KEY (`id042_formasEnvio`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5 Desarrollo

5.1 Planificación de las actividades de desarrollo

La planificación inicial al comenzar el proyecto fueron estas:

1. Requisitos del cliente

Tras unas entrevistas con el cliente se elaboraron los requisitos necesarios para el correcto desarrollo del proyecto. Se recogieron las especificaciones del producto y se escribieron unos informes detallando las tareas y características del producto final que fue aprobado por el cliente.

2. Wireframing y prototipado

A partir de las especificaciones de las entrevistas se realizó la arquitectura de la información del sitio así como su navegación entre las diversas páginas y cierta experiencia de usuario mediante un wireframe o prototipo. Este wireframe ha sido aprobado por el cliente.

3. Diseño

Teniendo en cuenta el prototipo anterior, conociendo el número de plantillas que tiene la web se dió comienzo al diseño del sitio. Se primó la usabilidad y claridad aportando un diseño limpio y moderno.

4. Maqueta

Cuando la mayoría de las plantillas fueron diseñadas se pudo empezar con la maquetación de las primeras. El marcado será en HTML5 aportando además de esta manera más semántica importante para el SEO. También se utilizó CSS3 e interactividad para la mejora de la experiencia de usuario (UX) con jQuery.

5. Desarrollo del gestor

Paralelamente a las tareas anteriores, una vez recogidos los requisitos se empezó con la programación del gestor que estará formado por tres bloques:

1. Módulo de imágenes

Desde el gestor se podrán subir imágenes que después se podrán asociar a los productos. Existirá una administración de imágenes que se podrán buscar, editar y borrar.

2. Módulo de productos

Se podrán dar de alta nuevos productos, modificar su precio, categoría, disponibilidad.

3. Módulo de ventas

Desde este módulo, se podrán chequear todas las ventas, buscar por estado (pagada, pendiente, enviada, cancelada...) editar (ver los datos del comprador, modificar los productos, o su estado) y se podrán generar ventas nuevas como puede ser los pedidos que puedan llevar por mail o teléfono para disponer de todas las ventas en el mismo sistema.

6. Integración

Un vez que la maqueta se ha finalizado y se han terminado los dos primeros, módulos del gestor, imágenes y productos, se integró la maqueta estática con la información almacenada en base de datos desde el gestor.

7. Pasarela de pago

El cliente contratará un TPV virtual con una entidad bancaria y esta se integrará en la web para además de realizar las compras mediante transferencia bancaria también se puedan pagar con tarjeta de crédito o débito.

8. Pruebas

Al finalizar todo el desarrollo se establecerán unos días de pruebas para testear el sistema antes de su puesta en producción y la corrección de los posibles bugs y fallos.

La planificación establecida inicialmente se fijo en un plazo de 8-9 semanas para el desarrollo del proyecto. Con 3 hitos de entrega principales. El primero era el wireframing. Posteriormente se entregará del gestor y se dará una explicación de su funcionamiento para el testeo por parte del cliente. Y por último la entrega final del proyecto. Esta planificación se puede ver en el siguiente diagrama de Gantt.

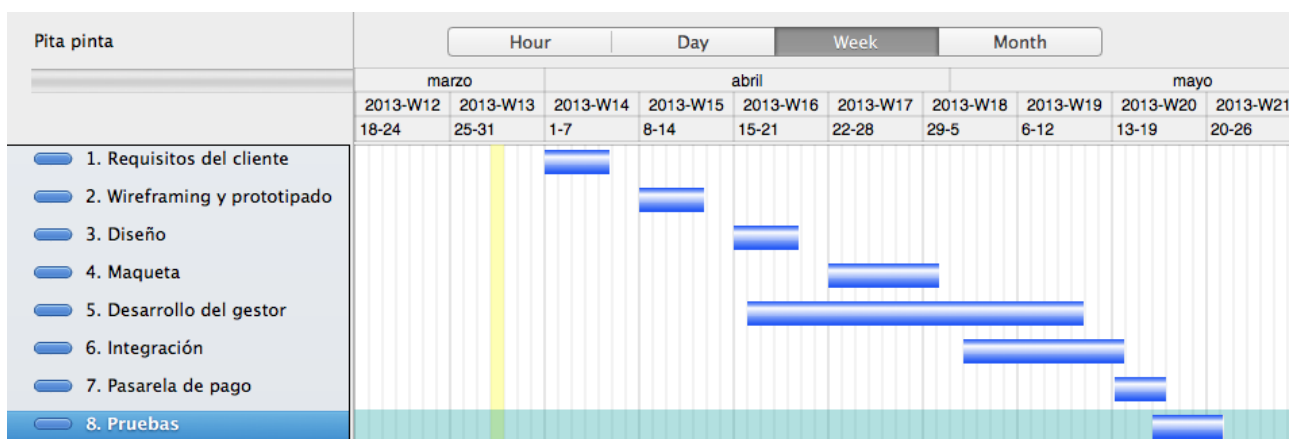


Diagrama de Gantt de la planificación inicial ante de empezar el proyecto

5.2 Desarrollo

La estructura del desarrollo en php del proyecto se compone de una serie de clases que prácticamente coinciden con las tablas de la base de datos y un fichero para cada clase con los métodos para acceder a ella. En esta caso las clases que se van a crear se lista en la siguiente imagen:

Nombre	Fecha de modificación
class.Categorias.php	18/05/2013 14:38
class.CommonPitaPinta.php	18/05/2013 14:38
class.EstadoVenta.php	18/05/2013 14:38
class.FormasEnvio.php	18/05/2013 14:38
class.Imagenes.php	18/05/2013 14:38
class.Productos.php	18/05/2013 14:38
class.Provincias.php	18/05/2013 14:38
class.Subcategorias.php	18/05/2013 14:38
class.Ventas.php	18/05/2013 14:38
class.VentasProductos.php	18/05/2013 14:38

A todas ellas hay que sumarse la clase PropertyObject que será la padre de todas las anteriores que heredarán de esta los métodos set y get.

Un ejemplo de la estructura de una clase base sería así, donde hay un método constructor que

genera los atributos y los relaciona con los de la base de datos, un método de borrado y un método destructor que cuando termina el script que ha llamada al constructor decide si el objeto se crea de nuevo porque no existía o se actualiza porque se estaba editando.

```
<?php
class Productos extends PropertyObject {

    public function __construct ($ProductosID) {

        if (isset($ProductosID)) {
            $arData = DataManagerProductos::getProductosData($ProductosID);
            parent::__construct ($arData);
        }

        $this->propertyTable ['id_producto'] = 'id01_productos';
        $this->propertyTable ['id_categoria'] = '02_categorias_id02_categorias';
        $this->propertyTable ['id_subcategoria'] =
'03_subcategorias_id03_subcategorias';
        $this->propertyTable ['id_imagenes'] = '07_imagenes_id07_imagenes';
        $this->propertyTable ['nombre'] = 'nombre';
        $this->propertyTable ['descripcion'] = 'descripcion';
        $this->propertyTable ['disponible'] = 'disponible';
        $this->propertyTable ['oferta'] = 'oferta';
        $this->propertyTable ['novedad'] = 'novedad';
        $this->propertyTable ['portada'] = 'portada';
        $this->propertyTable ['precio'] = 'precio';
    }

    public function del () {
        DataManagerProductos:: deleteProductos ($this);
    }

    public function __destruct () {











        if (sizeof($this->changedProperties)){
            if ($this->data ['id01_productos']) {
                DataManagerProductos::updateProductos ($this);
            }
            else {
                echo "en el destructor de Productos";
                DataManagerProductos::createProductos ($this);
            }
        }
    }
}

?>
```

Fichero class.productos.php

En ella se puede observar que cuenta con un método constructor y otro destructor. El primero, define para los nuevos elementos las propiedades y su relación con los campos de la base de datos y un método para obtener la información de ella si ya existe. El último distinguirá si se está trabajando con un objeto nuevo o la edición de uno existente para llamar a la inserción o a la actualización de la base de datos.

Para cada clase existe otra que es la de acceso a los datos, en este caso *DataManagerProductos* que contará con un método para obtener la información en diferentes puntos, tanto del portal como del gestor.

Nombre	Fecha de modificación
 class.DataManager.php	18/05/2013 14:38
 class.DataManagerCategorias.php	18/05/2013 14:38
 class.DataManagerEstadoVenta.php	18/05/2013 14:38
 class.DataManagerFormasEnvio.php	18/05/2013 14:38
 class.DataManagerImágenes.php	18/05/2013 14:38
 class.DataManagerProductos.php	18/05/2013 14:38
 class.DataManagerProvincias.php	18/05/2013 14:38
 class.DataManagerSubcategorias.php	18/05/2013 14:38
 class.DataManagerVentaProductos.php	18/05/2013 14:38
 class.DataManagerVentas.php	18/05/2013 14:38

El fichero class.DataManager.php se incluye en todas las plantillas php del proyecto, a su vez este fichero incluye el resto de ficheros DataManager. Todos los ficheros DataManager tiene en común 4 métodos (xxx es el nombre del fichero que identifica la clase)

getxxxData: Obtiene los datos de un elemento dado (id)

updatexxx: Actualiza los datos de un objeto previamente creado.

Createxxx: Después de llamar al constructor del objeto (new) si el objeto no existe se llama a este método para crearlo, si ya existe se llama al anterior para editarlo

deletexxx: Borra el objeto.

```
class DataManagerProductos {

    static $hDB;
    static $abiertas;
    static $cerradas;

    public static function getProductosData ($id) {

        $sql = "SELECT * FROM 01_productos WHERE id01_productos = ".$id;
        $res = mysql_query ($sql);

        if (! ($res && mysql_num_rows ($res))) {
            die ("Fallo en la obtencion de datos en productos");
        }
        return mysql_fetch_assoc ($res);
    }

    public static function updateProductos ($objeto) {

        $properties = $objeto->__getProperty ();
        $data = $objeto->__getData ();
        $changed = $objeto->__getChanged();

        $sql= "UPDATE 01_productos SET ";
        $setStatements = array ();
        foreach ($changed as $prop) {
            $setStatements[] = "$properties[$prop] =
{$data[$properties[$prop]]}";
        }
        $sql .= join (' , ', $setStatements);
        $sql .= " WHERE id01_productos = ".$data['id01_productos'];
        $res = mysql_query ($sql);
    }

    public static function createProductos ($objeto) {

        $data = $objeto->__getData ();
        $sql = " INSERT INTO 01_productos (". join (' , ', array_keys ($data)). "
values";

        $sql .= " ( " . join (' , ', array_values($data)) . " )";
        //echo $sql;
        $res = mysql_query ($sql);
    }
}
```

```
}  
  
public static function deleteProductos ($objeto) {  
  
    $sql = "DELETE FROM 01_productos WHERE id01_productos = ".$objeto->__get('id_producto');  
    echo "delete ".$sql;  
    $res = mysql_query ($sql);  
  
}
```

Además de estos cuatro métodos cada clase en concreto tiene los suyos como puede por ejemplo un método para listados en las clases de imágenes, productos y ventas.

Se ha comentado la base de como es la estructura de los scripts php que forman tanto el gestor como el sitio web. No se ha utilizado ningún framework, únicamente la capacidad de las últimas versiones de PHP de poder trabajar con clases y objetos.

5.3 Documentación

A fecha de hoy (Enero 2014) se está elaborando un manual de usuario destinado al cliente. En este manual se indicará como gestionar los productos y publicarlos en cada parte del portal. Como gestionar las imágenes y que herramientas son las idóneas para editar una imagen antes de subirlas. También se explica la manera de buscar una determinada venta, cambiar su estado, añadir nuevos productos y crear una nueva venta desde cero para poder usar la herramienta para ventas por teléfono o en el propio gallinero y tenerlas todas centralizadas.

No se desarrollará un manual del programador. No se valoró (presupuestó) desde un principio.

6 Implementación y mantenimiento

Antes de poder trabajar con el sitio web el cliente ha tenido que conocer como utilizar el gestor para administrar el contenido. Después de una reunión, una charla sobre el funcionamiento de los listados, los filtros de búsqueda, los campos obligatorios y su significado, ya pudo empezar a realizar las primeras pruebas y cargar los primeros contenidos.

El servidor de desarrollo (Fedora 19) se ha abierto al cliente en la última fase del desarrollo. Es el propio servidor de desarrollo el que ha servido para hacer primeras pruebas. En él el cliente ha cargado los primeros datos y tenido el primer contacto con el portal.

Para la implantación en producción no será necesario un servidor dedicado y puede que para las visitas que se esperan puede ser suficiente con un servidor compartido. Como para otras mejoras esto es algo que se analizará continuamente para ver el comportamiento del sistema y buscar una ampliación si fuera necesario.

6.1 Niveles de servicios

Para la próxima publicación del portal se establecerá un nivel de servicio básico. Atención en horario de oficina tanto a usuarios del portal por parte del cliente por correo electrónico y teléfono como al propio cliente por parte de la empresa desarrolladora. En principio esta última nunca atenderá directamente usuarios del portal.

6.2 Aceptación del sistema

Después de las pruebas del cliente, el grado de satisfacción ha sido grande. A la espera de la definición de la pasarela de pago que contrate con un banco se estaría en condiciones de publicar el sitio.

6.3 Mantenimiento

Los dominios y el hosting es contratado directamente por el cliente a una empresa externa. Una vez entregado el proyecto, la propia garantía del producto cubrirá los errores que pudieran surgir. Desde la empresa se ha llegado al compromiso de modificar las partes estáticas siempre que lo requiera el cliente ya que se prevé que no se produzcan muchos cambios.

7 Conclusiones

Después de cumplir con los plazos de ejecución del proyecto y con los objetivos inicialmente marcados por el cliente, el portal web a principios de 2014 aún no se ha publicado por decisión del mismo. El proyecto está a la espera de la selección de una pasarela de pago. Y este es el desarrollo que queda pendiente, la conexión con el TPV, aún así el proyecto sería totalmente operativo, si por ejemplo, el cliente decide publicarlo realizando ventas con transferencia bancaria.

Aún así el portal cuenta ya con más de 200 seguidores en Facebook y dos dominios que redirigen a la página donde se avisa de la próxima publicación:

<http://pitaasturiana.es/>

<http://pitapinta.net/>

El resultado final se tiene que evaluar sin la explotación en producción del portal, pero se pueden sacar conclusiones del, las apreciaciones (grado de satisfacción del cliente), resultado final, y el coste de desarrollo así como la evolución futura del producto.

Grado de satisfacción del cliente y resultado final.

El desarrollo se ha realizado según las necesidades del cliente y en función de los usuarios que podrá tener el portal. El público objetivo será en su mayoría un público no relacionado con las nuevas tecnologías. Por ello uno de los objetivos principales era que el portal fuese claro y sencillo de usar y ese fin creo que se ha conseguido, el cliente también está de acuerdo con ello. Aún así, el ciclo de mejora continua hará que se realice un análisis de la evolución del uso de la herramienta por parte de los clientes/usuarios para poder corregir y mejorar.

La formación y explicación del uso del gestor de productos y ventas ha sido rápida y el cliente ya ha hecho pruebas con productos, imágenes y ventas. Parte de esa formación y pruebas, ha sido que el propio cliente cargase toda la información que va a aparecer en la web. Le ha parecido un sistema sencillo, cómo y práctico como había requerido al principio del proyecto por lo que ha quedado bastante satisfecho con ello.

Coste de desarrollo

Es cierto que la decisión de realizar un desarrollo a medida puede ser algo comprometida. Creo que en un sistema con unas características muy determinadas y concretas debe resolverse de esta manera ya que el coste de una adaptación en muchos casos puede hasta superar el desarrollo a medida. Incluso el coste de dominar el producto para realizar una tarea muy específica en ocasiones es muy elevado.

Seguramente el caso de una tienda *online* no sea el mejor ejemplo para defender este argumento ya que como se explico en apartados anteriores existe herramientas que permiten crear una de forma fácil. Pero para desarrollos de portales, como es el caso de este proyecto, relativamente sencillos el coste de desarrollo frente al de adaptación puede que se den la mano. La ventaja de los productos empaquetados, como Drupal, Joomla, Wordpress, Open cms, Concrete5 o Liferay es que tienen la posibilidad de incorporar muchos módulos que faciliten las tareas más comunes como gestión de usuario, conexión a redes sociales, encuestas a usuarios y una infinita variedad de funcionalidades así como incluso la posibilidad de que faciliten un tema, un tema para también adaptar la parte visual del portal en cuestión. Aunque en muchos de ellos la curva de aprendizaje es elevada.

Si el portal no va a necesitar muchas evoluciones ni funcionalidades y se necesita un control completo del marcado HTML para cumplir uno de los objetivos como es el posicionamiento en buscadores y sobre todo su herramienta de gestión tiene que ser

sencilla, la mejor solución es un desarrollo a medida.

Uno de los mayores inconvenientes con que nos encontramos con productos como los gestores de contenidos generalistas que antes he mencionado es, en primer lugar, la dificultad con la que usuarios no expertos en informática encuentran a la hora de tener que administrar su contenido. La mayoría cuentan con muchas características que el usuario gestor del sitio nunca utiliza y lo único que hace es complicarle su trabajo a la hora de mantener el portal.

Evolución, mejoras.

En una segunda fase se podrían establecer las siguientes mejoras del portal, aunque no se espera que el portal evolucione mucho más allá que estos aspectos:

- Módulo generador de facturas. Generar un pdf para poder enviar al cliente. También se puede, desde el portal web, dar la posibilidad que los clientes que lo deseen marquen una casilla para recibir el pdf en su buzón de correo electrónico. Desde el gestor se tendrá acceso a todas las facturas y las ventas se podrán incluso filtrar por las que se ha generado su factura o no.
- Gestor de recetas para poder mantener un canal – gastro pita – de recetas relacionadas con huevos y carne de pollo.
- Gestor de recetas para poder mantener un canal – gastro pita – de recetas relacionadas con huevos y carne de pollo.
- Gestor de usuarios registrados. De esta manera los clientes más habituales no tendrán que repetir sus datos cada vez que quieran realizar una compra.
- Comentarios de los productos con cuentas de Facebook. Aportará confianza a los futuros compradores poder leer los comentarios de personas, totalmente identificables sobre los productos de la tienda. Además de ofrecer por su puesto más información con la opinión personal de otros compradores.

Finalmente, para mí ha sido un proyecto muy atractivo de abordar. Al ser un tipo de tienda que no abundan en Internet hace este portal pueda ser un referente. Los plazos aunque ajustados eran factibles y el desarrollo ha marcado todos sus hitos sin atascarse en ningún punto.

El producto final puede ser la base para cualquier tipo de tienda. El desarrollo del gestor de productos, imágenes y ventas podría servir para generar a partir de él otro tipo de portal o gestor para poder administrar otro tipo de productos simplemente con alguna modificación en los campos de la base de datos y poder reutilizarse para otra tienda *online* de características similares.

8 Bibliografía

MySQL Avanzado. ANAYA MULTIMEDIA. Jeremy D. Zawodny y Derek J. Balling

Desarrollo web con PHP6, Apache y MySQL. ANAYA MULTIMEDIA. Timothy Boronczyk