

**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA**

*GRADUADO EN INGENIERIA DEL SOFTWARE*

**IMPLEMENTACION DE UNA APLICACIÓN WEB USANDO EL SISTEMA GESTOR  
DE CONTENIDOS DRUPAL**

*IMPLEMENTATION OF A WEB APPLICATION USING THE CONTENT  
MANAGEMENT SYSTEM DRUPAL*

Realizado por

**Antonio Molina Olivares**

Tutorizado por

**Antonio Jesús Nebro Urbaneja**

Departamento

**Lenguajes y Ciencias de la Computación**

**UNIVERSIDAD DE MÁLAGA**

**Málaga, Julio 2015**

Fecha de defensa:

El secretario del tribunal:

**Resumen:** El proceso de desarrollo de software define una secuencia de actividades que se aplican en la creación de un producto o aplicación de tipo software. Entre las actividades que se pueden realizar dentro del ciclo de vida del desarrollo software destacamos algunas como pueden ser la captura de requisitos, el análisis, el diseño, la implementación, las pruebas, la documentación, el despliegue o el mantenimiento.

En este trabajo fin de grado se propone desarrollar una aplicación web desde sus primeras etapas hasta las últimas, indicando como aplica cada una de ellas al ejemplo de un proyecto real.

Para nuestro caso, el desarrollo de software consistirá en la creación de una aplicación web para una clínica podológica, en la que vamos a tener un cliente, en este caso el propietario de la clínica, que demanda funcionalidades y que necesita que el software cumpla con sus necesidades.

Dicha página Web servirá para promocionar los servicios de la clínica, mostrar una galería de fotos, tendrá un formulario de contacto, gran cantidad de páginas de información, un mapa de geo-localización para mostrar la ubicación, menús de navegación, mapa web, buscador y otras funcionalidades más típicas de cualquier página Web. Además, la aplicación deberá cumplir ciertos requisitos de usabilidad así como ser navegable en dispositivos móviles, responsivo.

La creación de la aplicación se hará con el sistema gestor de contenidos Drupal, una herramienta muy utilizada actualmente para crear y gestionar aplicaciones Web y con la que podremos implementar todas las funcionalidades demandadas por nuestro cliente.

**Palabras clave:** Drupal, Software, Ciclo, Vida, Ingeniería, Sistema, Gestión, Contenidos, Requisitos, Análisis, Diseño, Implementación, Documentación, Pruebas, Implantación, Despliegue, Calidad.

**Abstract:** The software development process defines a sequence of activities that apply to the creation of a product or application software. Among the activities that can be performed within the life cycle of software development we highlight some as requirements capture, analysis, design, implementation , testing, documentation, deployment and maintenance.

This final degree project it intends to develop a Web application from its early stages to the last , indicating how each of them applies to a real project.

In our case, the software development will be the creation of a Web application for a foot clinic, which will have a customer , in this case the owner of the clinic, who demands functionality and needs that the software resolves their needs.

This website will serve to promote the services of the clinic, show a photo gallery, have a contact form , a lot of pages of information, a geo-location map to show the location, navigation menus, site map, search and more typical functions os this kind of Web page. Besides this, the application must meet certain requirements of usability and mobile mobility, responsive design.

The creation of the application will be made with the content management system Drupal, currently used to create and manage Web applications and with which we can implement all the requirements demanded by our client.

**Keywords:** Drupal Software, Cycle, Life, Engineering, System, Management, Contents, Requirements, Analysis, Design, Implementation, Documentation, Tests, Implantation, Deployment, Quality

# Indice General

1. Introducción
  - 1.1 Objetivos
  - 1.2 Fases del Proyecto
  - 1.3 Estructura de la Memoria
  
2. Conceptos Básicos
  - 2.1 Apache
  - 2.2 MySQL
  - 2.3 XAMPP
  - 2.4 HTML
  - 2.5 CSS
  - 2.6 PHP
  - 2.7 DRUPAL CMS
  
3. Ciclo de Vida del Desarrollo Software
  - 3.1 Captura de Requisitos
  - 3.2 Análisis de la Solución
  - 3.3 Diseño de la Aplicación
  - 3.4 Implementación
  - 3.5 Pruebas
  - 3.6 Documentación
  - 3.7 Implantación y Despliegue
  - 3.8 Mantenimiento
  
4. Conclusiones y Trabajos Futuros
  
5. Referencias
  
6. Apéndice
  - 6.1 Contenido del CD
  - 6.2 Manual de Instalación y Configuración
  - 6.3 Manual de Usuario

# 1. Introducción

El término ciclo de vida del software describe el desarrollo de software desde la fase inicial hasta la fase final. El propósito de este proyecto es definir cada una de estas etapas en detalle y así poder validar el desarrollo de una aplicación Web.

Todo ingeniero software debe conocer perfectamente estas tareas y debe ser capaz de aplicarlas en el desarrollo de productos y aplicaciones software. Es importante conocer cuáles son estas actividades en el ciclo de vida de desarrollo pero mas aun lo es el saber como aplica cada una de ellas a la etapa del desarrollo en la que nos encontremos.

En este caso, se va a desarrollar una aplicación Web que servirá para promocionar los servicios de una clínica podológica, mostrar una galería de fotos, tendrá un formulario de contacto, gran cantidad de páginas de información, un mapa de geo-localización para mostrar la ubicación, menús de navegación, mapa Web, buscador y otras funcionalidades más típicas de un portal Web estándar.

## 1.1 Objetivos

Los objetivos del trabajo fin de grado son principalmente los siguientes:

1. Desarrollo de una aplicación Web paso a paso, desde las primeras etapas del desarrollo del ciclo de vida del software hasta las últimas.
2. Como proyecto desarrollado con Drupal, explicar como esta herramienta que nos aporta las soluciones que necesitamos para poder implementar el producto final.

### 1. Desarrollo de una aplicación Web

En este apartado se detallará paso por paso como iremos creando nuestra aplicación Web, desde las fases iniciales de captura de requisitos y análisis, pasando por la fase de configuración de la herramienta, la implementación, las pruebas o el despliegue.

Podemos diferenciar varios apartados en este punto:

- Definición de requisitos
  - Reuniones con el cliente
- Análisis de la solución
  - Viabilidad de la solución software
- Diseño de la aplicación
  - Creación de mockups de presentación que se ajusten a la interfaz gráfica deseada
  - Diseño de la estructura y regiones para el tema gráfico (cabecera, menú, cuerpo, pie, menús...)
- Implementación
  - Instalación y configuración de módulos necesarios
  - Creación de contextos para las páginas (unificar comportamientos comunes)

- Creación de bloques con funcionalidades (regiones particulares de código)
- Creación de listados para mostrar contenidos (uso de vistas)
- Creación de páginas de información (elemento básico de contenido)
- Creación de formularios (mediante componentes)
- Responsivo
- Pruebas
  - Tests en Drupal
- Documentación
  - Manual de usuario/administrador
- Despliegue
  - Instrucciones de instalación y despliegue
- Mantenimiento

Con el desarrollo de este trabajo fin de grado la idea es que un usuario sea capaz de crear software aplicando satisfactoriamente todas las etapas existentes en el desarrollo (que para nuestro caso se verá en la implementación de la aplicación Web de una clínica podológica) así como dar una visión cercana sobre cuáles son los conceptos mas importantes de un sistema de gestión de contenidos como Drupal y como aplicarlos para la gestión de un sitio Web.

## *2. Principales componentes de Drupal*

Se explicarán cuáles son los conceptos principales que maneja la herramienta y que son necesarios entender para la construcción de nuestra aplicación Web.

Sobre todo haremos hincapié en las principales componentes de esta herramienta como son las vistas, contextos, temas, páginas o bloques y la manera en la que aplican en nuestro proyecto para darnos la solución que necesitamos.

### **1.2 Fases del Proyecto**

La elaboración del trabajo fin de grado se puede dividir en las siguientes fases:

- Captura de requisitos de la aplicación Web y análisis de la solución
- Estudio y análisis general del sistema de gestión de contenidos Drupal
- Creación del diseño gráfico (hojas de estilo, imágenes, templates)
- Instalación y configuración de un servidor Apache con Mysql
- Instalación de Drupal
- Instalación y configuración de módulos adicionales
- Implementación de componentes y funcionalidades
- Personalización mediante la creación de contextos, bloques, vistas...
- Pruebas
- Documentación y elaboración de memoria del trabajo fin de grado
- Despliegue y mantenimiento

### 1.3 Estructura de la Memoria

Los capítulos de esta memoria son los siguientes:

- **Capítulo 1 Introducción.** Descripción general de los objetivos y fases del proyecto.
- **Capítulo 2 Conceptos Básicos.** Introducción a las tecnologías utilizadas en el proyecto. Descripción de conceptos relacionados.
- **Capítulo 3 Ciclo de Vida del Desarrollo Software.** Descripción de las principales etapas del desarrollo software y su aplicación a la Web.
- **Capítulo 4 Conclusión y Trabajos Futuros.** Comentarios sobre la aplicación.
- **Capítulo 5 Referencias.**
- **Capítulo 6 Apéndice.** Descripción del CD adjunto.

## 2. Conceptos Básicos

En esta sección se describen algunas de las componentes más importantes que se utilizan en el proyecto.

### 2.1 Apache

Apache es un servidor Web HTTP de código abierto para plataformas Unix, Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y el concepto de sitio virtual.

El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation (Fundación creada para dar soporte a los proyectos Software bajo la denominación Apache) y es usado principalmente para enviar páginas Web estáticas y dinámicas en la World Wide Web. Muchas aplicaciones Web están diseñadas asumiendo como ambiente de implantación a Apache, o que utilizarán características propias de este servidor.

Apache es el servidor de aplicaciones que se utiliza en la popular plataforma de aplicaciones LAMPP, junto a MySQL y los lenguajes de programación PHP, Perl, Python, o Ruby.

Entre las principales características destacamos las siguientes:

- **Modular** → Sus componentes se dividen en módulos del core y externos. Esto permite una mejor estructuración del código.
- **Código abierto** → Cualquier persona puede descargar y utilizar gratuitamente este servidor. Es software de libre distribución.
- **Multiplataforma** → Válido para múltiples entornos Linux, Macintosh o Windows.
- **Extensible** → Al ser modular permite la implementación de extensiones fuera del core, como por ejemplo la mod\_php.
- **Popular** → Fácil conseguir ayuda y soporte técnico



Para obtener mas información sobre Apache, se puede consultar el sitio:

<http://httpd.apache.org/>

## 2.2 MySQL

Es un sistema de gestión de bases de datos relacional, multihilo y multiusuario. MySQL se desarrolla como software libre licenciamiento dual. Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible de esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en el estándar ANSI C.

Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y los derechos de autor del código están en poder del autor individual, MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios.

Para hacerse una idea de la popularización que tiene MySQL hoy en día, basta con citar algunos de los proyectos que lo utilizan: Wikipedia, Google, Facebook, Twitter, Flickr y Youtube.

Entre sus principales características destacamos:

- **Multihilo** → Permite la separación de las tareas en múltiples hilos (threads) que son ejecutados por separado para obtener los resultados. Esto permite aumentar la velocidad en las operaciones lo que lo convierte en uno de los gestores relacionales más rápidos y con mejor rendimiento.
- **Multiusuario** → Permite procesar operaciones de múltiples usuarios simultáneamente.
- **Software libre** → Podemos utilizar licencias con fines no lucrativos de forma gratuita.
- **Multiplataforma** → Se puede ejecutar en muchos sistemas operativos (Linux, Mac, Windows....)
- **Popular** → Al ser muy utilizado, se puede encontrar información sin dificultad.



Para obtener más información sobre Apache, se puede consultar el sitio:

<https://www.mysql.com/>

## 2.3 XAMPP

Cuando queremos realizar proyectos Web se hace necesario la instalación de una serie de herramientas y aplicaciones para tal fin como por ejemplo compiladores de lenguajes de programación, servidores de aplicaciones para poder probar los

programas en modo stand-alone (sin uso de internet) o sistemas gestores de bases de datos para manejar la información de forma eficiente. Por todo lo anterior nace el paquete XAMPP para dar una solución a la integración de todas estas herramientas.

XAMPP es una distribución de Apache completamente gratuita y fácil de instalar que contiene MySQL, PHP y Perl. El paquete de instalación de XAMPP ha sido diseñado para ser increíblemente fácil de instalar y usar. Tiene licencia GNU que está orientada a proteger la libre distribución, modificación y su uso. El software cubierto por esta licencia es libre.

Con este paquete tendremos la posibilidad de probar nuestro entorno de trabajo sin la necesidad de tener acceso a internet.

Entre sus características destacamos:

- **Multiplataforma** → Se puede instalar tanto en Linux, como Windows como Mac.
- **Facilidad de uso** → Está preparado para que su instalación y configuración sea rápida y eficaz. La idea es que un usuario no avanzado sea capaz de usarlo sin problemas.
- **Integración de múltiples servicios** → La principal característica que proporciona XAMPP es la capacidad de tener en un solo paquete una distribución de PHP, un servidor de aplicaciones Apache y un gestor relacional de bases de datos MySQL. Pero además de estos tres servicios, también proporciona algunos más, por ejemplo SMTP para envío de correos o FTP para el intercambio de ficheros con el servidor a través de este protocolo.
- **Interfaz amigable para administración de MySQL** → XAMPP incorpora una herramienta llamada phpMyAdmin que permite administrar MySQL a través de una interfaz Web amigable. Desde phpMyAdmin podremos fácilmente crear, eliminar y alterar tablas así como borrar, editar y añadir campos a las mismas, ejecutar sentencias SQL, administrar claves en campos, administrar privilegios o exportar datos en distintos formatos (CSV, SQL, ...)



Para obtener mas información sobre Apache, se puede consultar el sitio:

<https://www.apachefriends.org/es/index.html>

## 2.4 HTML

**HTML (HyperText Markup Language)** es un lenguaje muy sencillo que permite describir hipertexto, es decir, texto presentado de forma estructurada y agradable, con

enlaces (hyperlinks) que conducen a otros documentos o fuentes de información relacionadas, y con *inserciones* multimedia (gráficos, sonido...). La descripción se basa en especificar en el texto la estructura lógica del contenido (títulos, párrafos de texto normal, enumeraciones, definiciones, citas, etc) así como los diferentes efectos que se quieren dar (especificar los lugares del documento donde se debe poner cursiva, negrita, o un gráfico determinado) y dejar que luego la presentación final de dicho hipertexto se realice por un programa especializado (como por ejemplo exploradores Web como Internet Explorer, Firefox, Chrome, etc...).

La mayoría de los efectos se especifican de la misma forma: rodeando el texto que se quiere marcar entre dos etiquetas o directivas (tags, en inglés), que definen el efecto o unidad lógica que se desea. Las etiquetas están formadas por determinados códigos metidos entre los signos < y >, y con la barra / cuando se trata de la segunda etiqueta de un efecto (la de cierre). Por ejemplo: <efecto> para abrir y </efecto> para cerrar. Ciertas directivas sólo se ponen una vez en el lugar del texto donde queramos que aparezca el efecto concreto.

A veces es necesario ofrecer datos adicionales en una directiva. Por ejemplo, cuando se define un hiperenlace hay que especificar su destino. Para ello se incluyen parámetros en la directiva inicial (la de apertura), de la siguiente forma: <efecto parametro1 parametro2 ...>. La directiva de cierre, caso de ser necesaria, queda como antes: </efecto>.

```
<html>
<head>
<title>Hola Mundo!</title>
</head>
<body>Eso es mi primer documento HTML!</body>
</html>
```

Para obtener mas información sobre HTML y todas las posibilidades que ofrece este lenguaje, se puede consultar el sitio Web: <http://www.w3schools.com/html/>

## 2.5 CSS

**CSS (Cascading Style Sheets)** es un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

Las hojas de estilo proporcionan un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos. La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación.

Por ejemplo, el elemento de HTML `<h1>` indica que un bloque de texto es un encabezamiento y que es más importante que un bloque etiquetado como `<h2>`. Versiones más antiguas de HTML permitían atributos extra dentro de la etiqueta abierta para darle formato (como el color o el tamaño de fuente). No obstante, cada etiqueta `<h1>` debía disponer de la información si se deseaba un diseño consistente para una página y, además, una persona que lea esa página con un navegador pierde totalmente el control sobre la visualización del texto.

Cuando se utiliza CSS, la etiqueta `<h1>` no debería proporcionar información sobre como va a ser visualizado, solamente marca la estructura del documento. La información de estilo separada en una hoja de estilo, especifica cómo se ha de mostrar `<h1>`: color, fuente, alineación del texto, tamaño y otras características

La información de estilo puede ser adjuntada tanto como un documento separado o en el mismo documento HTML. En este último caso podrían definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo "style".



Para obtener mas información sobre CSS y todas las posibilidades que ofrece este lenguaje, se puede consultar el sitio Web: <http://www.w3schools.com/css/>

## 2.6 PHP

**PHP (Hypertext Pre-processor)** es un lenguaje de programación de uso general de código de lado del servidor originalmente diseñado para el desarrollo Web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página Web resultante.

PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes. Puede ser usado en la mayoría de los servidores web al igual que en casi todos los sistemas operativos y plataformas sin ningún costo.

Se considera uno de los lenguajes más flexibles, potentes y de alto rendimiento conocidos hasta el día de hoy, lo que ha atraído el interés de múltiples sitios con gran

demanda de tráfico, como Facebook, para optar por el mismo como tecnología de servidor.

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="UTF-8" />
    <title> Ejemplo básico PHP</title>
  </head>
  <body>
    <?php
      echo 'Hola mundo';
    ?>
  </body>
</html>
```

Para obtener mas información sobre CSS y todas las posibilidades que ofrece este lenguaje, se puede consultar el sitio Web: <http://php.net/>



## 2.7 DRUPAL CMS

Un sistema gestor de contenidos (**Content Management System**) es un programa informático que permite crear una estructura de soporte para la creación y administración de portales Webs y contenidos.

Consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio web. Este tipo de sistemas permiten manejar de forma independiente el contenido y el diseño. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio Web sin tener que darle formato al contenido nuevo, además de permitir la fácil y controlada publicación de contenidos en el sitio. Por ejemplo, un editor se puede encargar de meter el contenido en el sistema, mientras que un administrador permite que esta información sea visible a todo el público.

La creación de la nuestra aplicación Web se hará con el sistema gestor de contenidos Drupal, que es un potente sistema de gestión de contenidos que ofrece grandes posibilidades a los usuarios. Está implementado con código libre, es modular y tiene gran variedad de propósitos. Tiene licencia Gnu/Gpl y está desarrollado con lenguaje Php que se combina con Mysql para dar una solución funcional completa.

Permite un alto grado de configuración con la que el usuario puede manejar de forma eficiente todos los recursos y servicios de un sitio Web. Puede gestionar artículos, imágenes, videos y otros muchos tipos de contenidos, así como cualquier servicio añadido como pueden ser foros, encuestas, votaciones, blogs, mensajería,

administración de usuarios, permisos, servicios Webs, portales, intranets y muchos mas.

Las aplicaciones prácticas reales que tiene Drupal al mundo Web son infinitas, desde poder gestionar una comunidad en Internet hasta poder ofrecer soluciones empresariales mas complejas. La gran cantidad de módulos que lo componen hacen que tenga mucha versatilidad y que su uso pueda ser adecuado en muchas situaciones y necesidades diferentes.

Drupal posee muchas componentes importantes, entre los que destacan los conceptos de vista, bloque, tema, contexto o tipos de contenido.

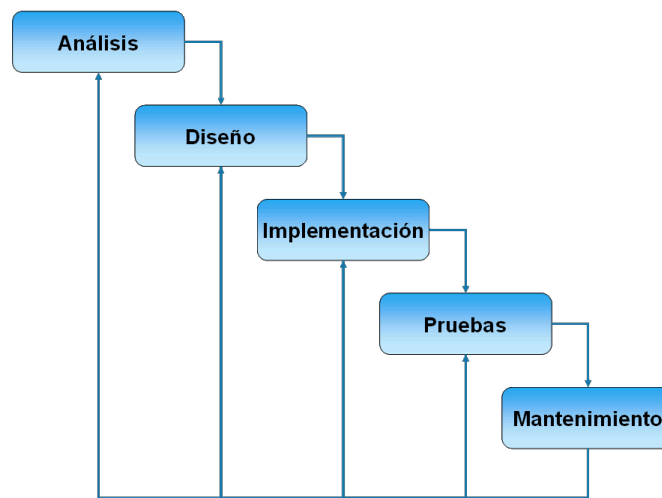


Para consultar toda la información relacionada con este sistema de gestión se puede consultar el enlace: <https://www.drupal.org/>

### 3. Ciclo de Vida del Desarrollo Software

El proceso para el desarrollo de software, también denominado ciclo de vida del desarrollo de software es una estructura aplicada al desarrollo de un producto de software. Hay varios modelos a seguir para el establecimiento de un proceso para el desarrollo de software, cada uno de los cuales describe un enfoque diferente para diferentes actividades que tienen lugar durante el proceso.

En el caso de nuestra aplicación Web, el modelo utilizado ha sido el tradicional modelo en cascada pero con algunos conceptos de desarrollo ágil.



El desarrollo tradicional en cascada establece que los distintos procesos del desarrollo software se ejecutan de manera sucesiva. El modelo más estricto de este tipo establece que sólo cuando se finaliza una fase, comienza la siguiente. Es decir, se desaconseja visitar y revisar fases que ya se han completado y como consecuencia de esta falta de flexibilidad el modelo de cascada puro es en ocasiones criticado frente a los defensores de modelos más flexibles.

Precisamente por lo anterior, el modelo en cascada utilizado para el ciclo de vida del software del proyecto no ha sido del todo puro si no que ha tenido cierta flexibilidad. En ocasiones se ha realizado una revisión de alguna fase anterior pero siempre con el objetivo de alcanzar el mejor producto al final del ciclo.

En esta sección se van a describir cuáles son las principales etapas en el desarrollo software y como se ha aplicado cada una de ellas a nuestro proyecto.

El ciclo de vida básico de software cuenta con los siguientes procedimientos:

- Definición de requisitos
- Análisis de requisitos y estudio de la viabilidad
- Diseño de la solución
- Implementación
- Pruebas

- Documentación
- Implantación
- Mantenimiento

### 3.1 Captura de Requisitos

#### Introducción

Esta etapa mas formalmente llamada ingeniería de requisitos, comprende todas las tareas relacionadas con la determinación de las necesidades y requisitos a satisfacer para cumplir con los objetivos del producto software.

En el ámbito de la ingeniería del software, podemos definir un requisito como una necesidad documentada sobre el contenido, forma o funcionalidad de un producto o servicio. Establecen que debe hacer un sistema pero no como hacerlo.

Destacamos dos tipos de requisitos:

- **Funcionales** → son descripciones documentadas de cosas que debe hacer el producto a entregar.
- **No funcionales** → pueden referirse a conceptos mas generales pero no por ello menos importantes, por ejemplo rendimiento, calidad o usabilidad.

Los buenos requisitos deben ser ponderables y sobre todo no contener ambigüedades o contradicciones.

La captura de requisitos en general es de vital importancia porque depende de ella el éxito del resto ya que es donde tenemos que asentar las especificaciones del proyecto y tener absolutamente claro que tenemos que hacer. De hecho, los requisitos son directamente los datos de entrada que recibe la etapa justamente posterior en el ciclo de vida, la etapa de diseño.

El propósito de la ingeniería de requisitos es hacer que los requisitos alcancen un estado óptimo antes de llegar a la fase de diseño en el proyecto.

#### Aplicación

El desarrollo de la aplicación Web comenzó con una primera reunión con el cliente, en nuestro caso una podóloga que acababa de abrir su clínica y quería promocionarse.

En ella el cliente expresó que quería crear un portal Web con los siguientes requisitos:

1. Determinar la tecnología y herramientas a utilizar
2. Promocionar los servicios destacados
3. Incorporar distintas secciones de información
4. Mostrar banners representativos
5. Poder realizar ofertas y promociones

6. Formulario de contacto
7. Mapa de geo-localización para ubicar la clínica fácilmente
8. Buscador de información de contenido
9. Galería de fotos
10. Adaptada a dispositivos móviles
11. Zona donde poder dar consejos a los usuarios
12. Enlaces destacados
13. Facilidad de gestión de la información
14. Temática de acuerdo a los colores de la clínica
15. Fácil navegación
16. Buen rendimiento
17. Coste económico no elevado
18. Estabilidad y seguridad del software

La mayoría de los requisitos como se puede apreciar son no funcionales, es decir, son requisitos sobre conceptos o ideas que se necesitan implementar pero que no se encuentran muy detallados. Los requisitos no funcionales son mas habituales cuando el cliente no es del ámbito tecnológico o no tiene normalmente experiencia en requerir este tipo de productos o servicios ya que cuando es así normalmente los requisitos se especifican en detalle y quedan perfectamente documentados.

En principio podemos decir que las necesidades estaban claras y las funcionalidades requeridas eran bastante comunes. El cliente no detalló en exceso cuál quería que fuese la estructura de la Web, tenía claro las conceptos que necesitaba pero lo dejaba en mis manos para analizar la solución y poder darle una respuesta sobre si era o no viable hacer el proyecto con lo que estaba pidiendo.



### **3.2 Análisis de la Solución**

#### *Introducción*

En esta etapa del ciclo de vida del desarrollo software el ingeniero determina si es o no viable la realización del producto de acuerdo a las necesidades del cliente.

Hay que analizar con especial detalle todo los requisitos que en principio puedan ser mas complicados porque de su ejecución podrá depender que la implementación del producto final se lleve a cabo o no. Incluso aunque se de el caso de que algún requisito no se puede realizar o sea incompatible con otro, se pueden proponer alternativas siempre y cuando cuenten con la aprobación final del cliente.

Los posibles problemas que se pueden producir durante el análisis de los requisitos son:

- El personal técnico y los clientes finales pueden tener vocabularios distintos y pueden llegar a creer incorrectamente que están de acuerdo cuando en realidad ambos se están refiriendo a cosas distintas.
- Los desarrolladores pueden intentar encajar el sistema en un modelo existente en vez de desarrollar un sistema nuevo y único adaptado al cliente.
- En ocasiones el análisis de requisitos lo puede hacer un ingeniero o programador que no tenga las suficientes habilidades de relación o el conocimiento apropiado para entender las necesidades de un cliente.



### Aplicación

A continuación se muestran las deducciones y conclusiones que obtuve al analizar cada uno de los requisitos presentados.

1. **Determinar la tecnología y herramientas a utilizar** → Teniendo una visión genérica de las necesidades del cliente y con el objetivo de ver si cada requisito podría ser satisfecho o no cuando lo analizara, creí conveniente primero intentar buscar una solución tecnológica que se adaptara a nuestras necesidades. Teniendo en cuenta que existían requisitos vitales para el cliente tales como el rendimiento, el coste económico y la facilidad de gestión y administración decidí que lo que tenía que usar era un sistema gestor de contenidos o más comúnmente llamado CMS (Content Management System).

Un CMS tiene la ventaja que al estar pensado ya de por sí para la gestión presenta interfaces de usuario amigables y que facilitan las tareas de mantenimiento. Teniendo en cuenta que debía abaratar costes como fuese posible para dar viabilidad al proyecto, pensé en un CMS de software libre. Por último, para asegurar el correcto funcionamiento y rendimiento me puse a investigar sobre CMS que fuesen muy populares y que tuvieran gran comunidad de usuarios detrás con el objetivo de que ante cualquier duda que pudiera surgir, poder encontrar una solución rápidamente.

Por todo lo anterior decidí inclinarme por Drupal (implementado con PHP) debido también a que vi que tenía un carácter bastante modular y eso me podría ayudar a la hora de incorporar de forma independiente las distintas funcionalidades que el cliente demandaba.

2. **Promocionar los servicios destacados** → Una clínica de podología presta servicios de distinto tipo de acuerdo a los tratamientos que ofrece. Cualquier usuario de una Web que entre en una página que servicios lo que quiere es poder encontrarlos rápido y con buenas descripciones de qué pueden ofrecer. Al ser quizás el punto de información mas importante a mostrar consideré que debía tener una sección específica para mostrarlos. Desde el punto de vista de Drupal estos servicios estarían en un contenido de tipo página.
3. **Incorporar distintas secciones de información** → Como cualquier negocio, es importante además de promocionar los servicios, otro tipo de información. Es por ello que pensé en la creación de distintas secciones también teniendo en cuenta algunos requisitos que me había percatado que serían necesarios:
  - a. Inicio, a modo de página principal donde mostrar información directa).
  - b. ¿Quiénes somos? Información mas concreta sobre la clínica y su personal).
  - c. Equipamiento, donde mostrar los utensilios y maquinaria con los que la clínica está equipada.
  - d. Servicios, mostrar los servicios que la clínica realizar.
  - e. Ofertas, donde ubicar las ofertas en promoción de la clínica.
  - f. Contacto, para incluir un formulario de contacto.
  - g. Buenas prácticas, para mostrar consejos relacionados con el cuidado del pie.
  - h. ¿Cómo llegar? Incluyendo un mapa de geo-localización de la clínica.

Todos estos contenidos también podrían considerarse como páginas dentro de la arquitectura de contenidos de Drupal. Además todas estas secciones iban a estar representadas a través de un menú transversal de navegación.

4. **Mostrar banners representativos** → Se refiere a información mas destacada y visible que debía venir acompañada como es lógico de una imagen representativa de la información que mostraba. Drupal ofrece un módulo que se llama Nivo Slider y que ofrece precisamente esta funcionalidad.
5. **Poder realizar ofertas y promociones** → Para ello tenía pensado crear una sección específica donde poder mostrar contenido sobre ofertas y promociones. Igualmente sería un contenido de tipo página de Drupal.
6. **Formulario de contacto** → Ante la necesidad del cliente de poder facilitar a los usuarios que se pusieran en contacto con él, mi idea era crear una nueva sección única y exclusivamente para poder contactar con la clínica. El módulo Webform de Drupal me daba la solución.
7. **Mapa de geo-localización para ubicar la clínica fácilmente** → Está muy de moda tener un mapa de tipo Google Maps para poder ubicar en un mapa dinámico un lugar. Pretendía justamente esto y ví que lo podía conseguir utilizando el módulo de Drupal Location Map.

8. **Buscador de información de contenido** → El objetivo de este requisito es que cualquier usuario pudiera realizar búsquedas de contenidos para poder acceder directamente a los temas de su interés. El módulo Search de Drupal proporciona esta funcionalidad.
9. **Galería de fotos** → El cliente quería tener una sección donde poder ver fotos de su clínica. Para ello pensé habilitar en el pie de página un listado de imágenes que se pudieran mostrar al pulsar sobre ellas. En Drupal esto se podía implementar con el módulo Lightbox.
10. **Adaptada a dispositivos móviles** → Era fundamental que la aplicación Web se visualizase correctamente en dispositivos móviles. Drupal proporciona gran cantidad de temas para poder aplicar a un sitio Web, de manera que utilicé uno de ellos que ya tenía implementada esa característica responsive. El tema utilizado es Rubix Responsive Theme aunque por supuesto tuve que hacer muchas modificaciones en él para adaptarlo a la temática y estilo gráfico de la clínica.
11. **Zona donde poder dar consejos a los usuarios** → Se podía considerar un tipo de contenido de Drupal de tipo página. Tendrá una ubicación destacada en el menú de la aplicación.
12. **Enlaces destacados** → Similar a las páginas de información, Drupal permite definir bloques de código para ubicarlo en distintas regiones de la Web. Los enlaces destacados serían precisamente esto.
13. **Facilidad de gestión de la información** → Drupal proporciona una interfaz amigable para que una persona con los permisos necesarios sea capaz de gestionar y mantener los contenidos de la Web.
14. **Temática de acuerdo a los colores de la clínica** → Además del concepto de responsive, la aplicación debía poseer estilos gráficos que fuesen en la misma línea de la decoración de la consulta. Como la clínica tiene tonos blancos, grises y morados, esos mismos colores son los que había que modificar en las hojas de estilos del tema para que quedase de acuerdo a las necesidades indicadas.
15. **Fácil navegación** → Uno de los aspectos mas importantes a tener en cuenta en cualquier producto o servicio, incluso independientemente del ámbito al que nos refiramos, es la facilidad para poder interactuar con él. Si no diseñamos un producto que sea intuitivo y que realmente satisfaga las necesidades de los clientes habremos fracasado. No hay que olvidar que por muy bonita que sea una aplicación Web si al final no es intuitiva lo más probable es que no tenga éxito.

Drupal tiene una estructura bastante jerarquizada de componentes con las cuáles se puede gestionar de manera eficiente los elementos visibles y facilitar la navegación al usuario. Mediante el uso de menús, bloques, contextos, paneles, categorías, etc... se permite personalizar el aspecto de nuestra Web.

16. **Buen rendimiento** → Nuestro CMS es mundialmente conocido por la gran cantidad de usuarios que lo utilizan. Existe una comunidad muy grande de

programadores que mantienen Drupal y van continuamente sacando nuevas versiones del core (núcleo) con el objetivo de ir siempre mejorando sus características. Existen algunos módulos en Drupal para mejorar el rendimiento, pero el más importante es sin duda Memcache.

**17. Coste económico no elevado** → Teniendo en cuenta que el coste del proyecto no podía ser elevado, recortar gastos en la tecnología a utilizar como consecuencia de no tener que pagar licencias era un punto a favor y por ese motivo Drupal cumple perfectamente con este requisito.

**18. Estabilidad y seguridad del producto software** → Un aspecto interesante se refería a la estabilidad del producto. Desde el punto de vista de programación, la comunidad de Drupal se encarga de estabilizar el software cada cierto tiempo liberando las últimas modificaciones que tiene testeadas y que aportan valor.

Desde el punto de vista de la gestión del código, la seguridad del mismo se puede asegurar utilizando un control de versiones para tener un control exhaustivo de los cambios. Esto va ligado directamente con la mantenibilidad. En nuestro caso hemos usado el cliente Tortoise SVN (<http://tortoisesvn.net/>)

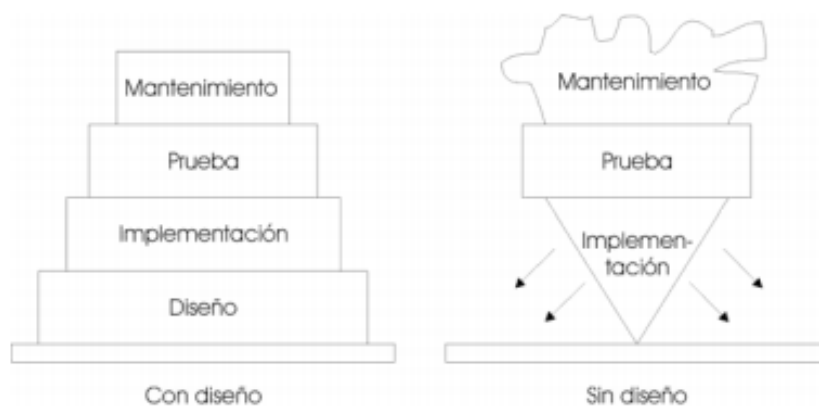
### 3.3 Diseño de la Aplicación

#### Introducción

El diseño es el primer paso de la fase de desarrollo de cualquier producto o sistema de ingeniería.

Esta etapa del ciclo de vida del desarrollo permite describir como el sistema va a satisfacer los requisitos. A menudo tiene diferentes niveles de detalle. Los niveles más altos de detalle generalmente describen los componentes o módulos que formarán el software a ser producido. Los niveles más bajos, describen, con mucho detalle, cada módulo que contendrá el sistema.

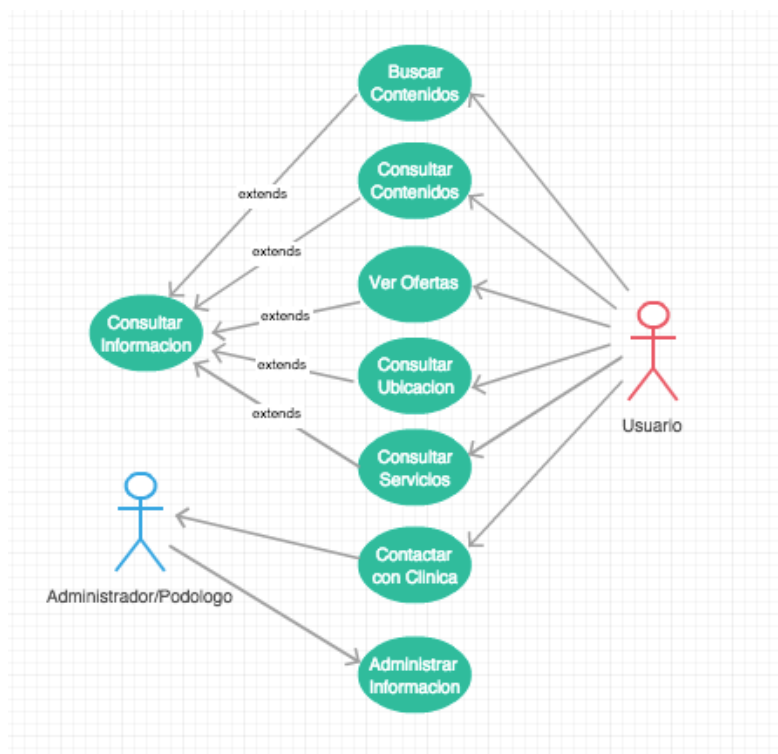
Una vez que se han establecido los requisitos del software, el diseño es la primera de las tres actividades técnicas: diseño, codificación y prueba. Cada actividad transforma la información de forma que al final se obtenga un software validado.



El diseño es la única forma mediante la que podemos traducir con precisión los requisitos del cliente en un producto o sistema acabado. Es la base de todas las partes posteriores del desarrollo y de la fase de pruebas.

### Aplicación

En nuestro proyecto el diseño de la aplicación no tenía una estructura predeterminada, es decir, el cliente nos solicitó una serie de requisitos pero no hizo tanto hincapié en el diseño de la aplicación, lo único que quería es que la temática de la misma tuviese relación con la clínica (por ejemplo en cuanto a colores) pero no indicó nada sobre cómo se iban a distribuir los contenidos en el portal Web.



Teniendo decidido a Drupal como el CMS a utilizar como solución tecnológica al proyecto, y sabiendo que la herramienta me proporcionaba soluciones para poder implementar los requisitos solicitados, lo siguiente fue empezar a proponer bocetos de diseño de cómo irían quedando las distintas partes de la aplicación.

Es decir, antes de poder empezar con la implementación, el cliente tiene que dar su aprobación al diseño para ver si se ajusta a sus necesidades, y por ello estructuré la Web en varias pantallas principales cuyo mockup fue:

#### **3.3.1 Pantalla principal**

Es la pantalla principal de acceso al portal, también comúnmente conocida como Home. En ella quería concentrar los contenidos mas importantes de la clínica de tal manera que un usuario al entrar en la Home pudiera tener información directa de cuáles son las posibilidades que ofrece la clínica.

El mockup del diseño de esta pantalla es el siguiente:



Las principales componentes que destacamos son:

- Nombre de la clínica
- Buscador
- Menú transversal (Inicio, ¿Quiénes somos?, Equipamiento, Servicios, Ofertas, Buenas prácticas, ¿Cómo llegar?)
- Banner informativo
- Bloques de información (Horario, Contacto, ¿Dónde estamos?)
- Sección información (Nombre clínica, servicios, descripción)
- Bloques con imágenes seleccionables (Equipamiento, Servicios, Atención a domicilio)
- Podología en general
- Mapa Web (enlaces)
- Galería de imágenes seleccionables

Además de las componentes anteriores, podemos destacar que vamos a tener varias regiones que son compartidas por todas las pantallas, y que son:

- Cabecera (que incluye nombre de la clínica, buscador y menú transversal)
- Pie (que incluye información de podología en general, mapa web y galería de imágenes)
- Contenido (esta región cambiará en función de la página que estemos visitando)

### 3.3.2 Pantalla detalle contenido

El mockup del diseño de esta página es:



Este tipo de página mantiene la misma cabecera y pie que en el caso de la pantalla principal, lo único que cambia es la región de contenido.

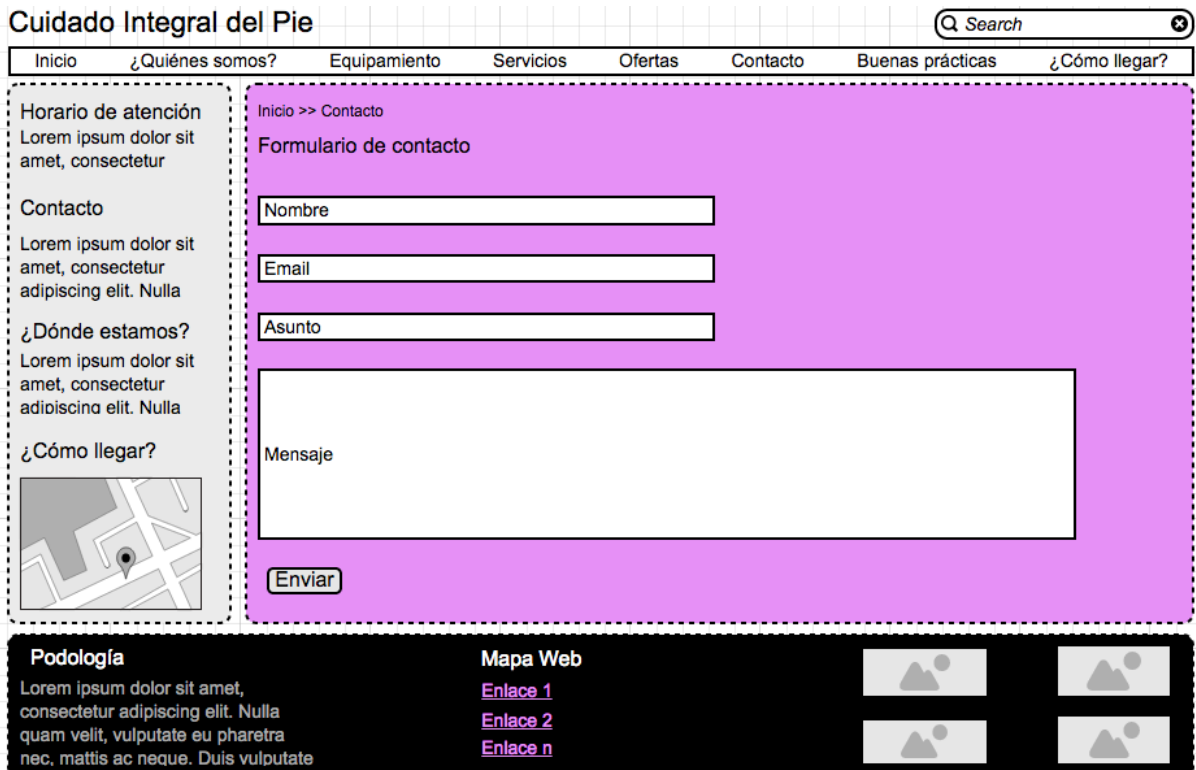
Destacar en esta pantalla que aparece una nueva componente que es la barra de información vertical y que mostrará información útil para el usuario así como algunas fotos y enlaces de interés.

Pertencen a este tipo de pantalla las secciones de ¿Quiénes somos?, Equipamiento, Servicios, Ofertas, Contacto y Buenas prácticas.

### 3.3.3 Pantalla formulario contacto

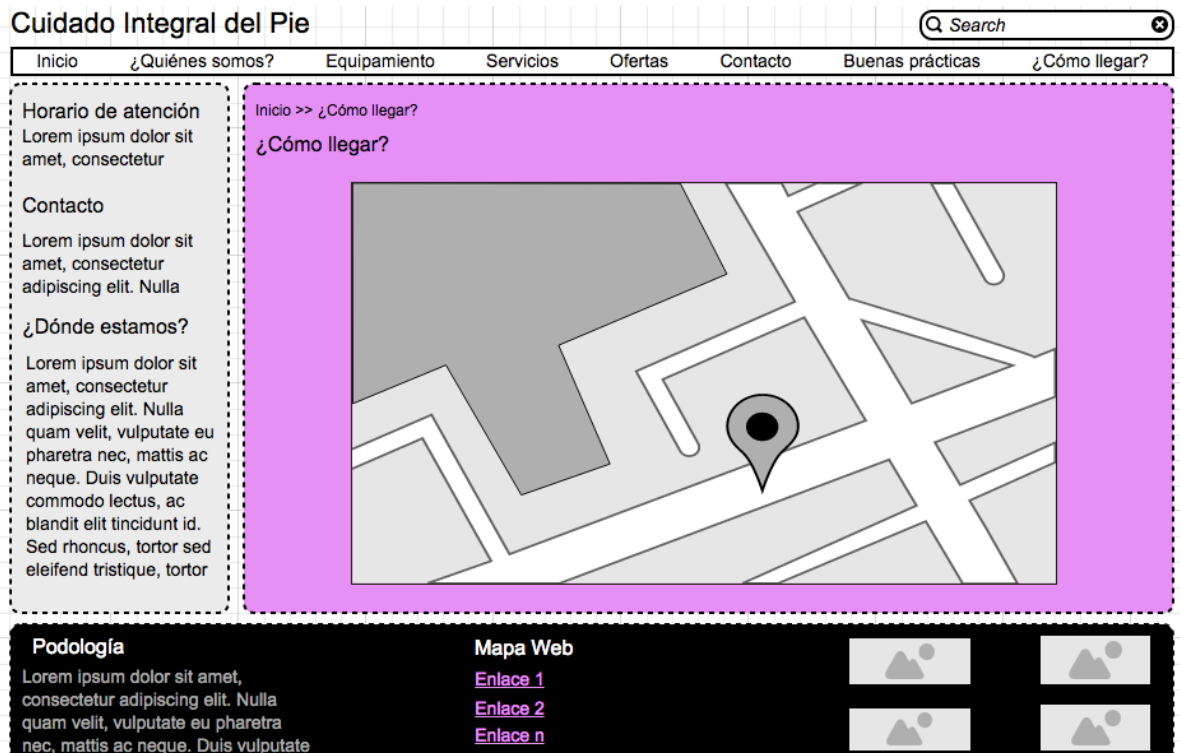
Este tipo de pantalla es similar al anterior, por supuesto comparte la cabecera y el pie común a toda la aplicación pero cambia ostensiblemente la información generada en la zona de contenido. La barra de información vertical mostrará información asociada al horario, contacto, dónde estamos o cómo llegar mientras que la descripción de la página incluirá un formulario de contacto.

El mockup de esta pantalla es:



### 3.3.4 Pantalla ¿Cómo llegar?

Su diseño es muy similar a la pantalla anterior, únicamente que se elimina el mapa de geo-localización de la barra vertical de información y se incorpora como zona principal del contenido ya que no tenía mucho sentido mantener el mapa pequeño en la barra vertical cuando podemos encontrar el mapa principal en el contenido de la pantalla.



Acabada la parte de diseño de la aplicación, el siguiente paso fue mostrar todos estos mockups al cliente con el objetivo de poder obtener el visto bueno que permitiera seguir con la siguiente etapa, la de implementación.

El cliente quedó muy satisfecho con el resultado, y le gustó mucho la sencillez e información directa y concisa que quería mostrar. Después de comentarle que utilizaría una tecnología de software libre para implementar la solución (con lo cuál los costes se reducían) que era muy popular y estable con la que se aseguraría el buen funcionamiento de la Web, y sobre todo después de dar el visto bueno al boceto de diseño que le presenté, el proyecto recibió la aprobación del cliente y podíamos pues ya empezar con la fase de implementación.

### **3.4 Implementación**

#### *Introducción*

Partiendo del análisis y diseño de la solución, en esta etapa se procede a desarrollar el correspondiente programa que solucione el problema mediante el uso de una herramienta computacional determinada. En el ámbito de las ciencias de la computación, la implementación de un producto software se refiere a la realización de sus especificaciones técnicas.

En definitiva, decimos que la implementación es la fase que da vida al producto y lo transforma de una idea a un producto real.

#### *Aplicación*

La implementación de la aplicación Web para la clínica podológica se ha realizado con el gestor de contenidos Drupal. En la fase de análisis ya se pudo comprobar que los requisitos del cliente se podían satisfacer utilizando este CMS.

En este apartado se detalla como se ha realizado paso a paso el proyecto desde el punto de vista de la implementación, por lo que iremos viendo como mediante el uso de Drupal hemos ido creando las pantallas que ya tenían la aprobación del cliente en la fase anterior de diseño.

#### **3.4.1 Preparación entorno desarrollo**

Consiste en la instalación de Drupal en un entorno local de desarrollo, en este casi mi equipo de trabajo. Previo a la instalación de Drupal instalé el paquete XAMPP con el que conseguí tener configurado un servidor Apache, un gestor de base de datos MySQL y además una herramienta de manejo del mismo muy útil, phpMyAdmin. Como editor de desarrollo he utilizado Eclipse para PHP.

La instalación de todas estas herramientas de trabajo puede realizarse fácilmente siguiendo cualquier guía por la Web o directamente entrando en las páginas oficiales de cada una de ellas.

### 3.4.2 Instalación de módulos

Drupal se caracteriza por ser un CMS completamente modular. Podemos diferenciar entre los módulos del core, los módulos contribuidos y los customizados dependiendo de si son del propio sistema, si hay que instalarlos a parte o bien si hay que crearlos de cero.

En nuestro caso vimos que la mayoría de funcionalidades del sistema al ser genéricas, podíamos satisfacerlas mediante la utilización de módulos contribuidos en la mayoría de los casos. En concreto los módulos que hemos utilizado para nuestro proyecto son:

- CKEditor → Permite utilizar en la edición de contenido texto enriquecido a modo de editor WYSIWYG (what you see is what you get)
- Lightbox 2 → Utilizado para implementar la galería de fotos
- SMTP Authentication Support → Usado para el envío de correo para el formulario de contacto
- Location Map → Incorpora mapas de Google
- Nivo Slider → Banner rotatorios
- Webform → Implementación de formularios
- Overlay → Facilita la gestión de información para el administrador
- Context → Define tipos de pantallas independientes donde insertar bloques de código
- Superfish → Para implementar fácilmente menús desplegados

Además de estos módulos por supuesto se han tenido que utilizar muchos del core de Drupal que son básicos para su funcionamiento, entre los que destacamos los siguientes:

- Search → Proporciona búsquedas sobre los contenidos del sistema
- Devel → Usado para el desarrollo, ayuda en la programación de páginas, bloques, funciones, etc...
- Exclude node title → Elimina el título por defecto que aparece en los detalles de los contenidos
- Pathauto → Simplifica urls largas y las sustituye por nombres amigables
- Views → Módulo fundamental para la creación de listados de contenidos
- Views PHP → Incorpora la posibilidad de definir un campo de tipo PHP sobre una vista para mostrar el resultado de la ejecución de cualquier código PHP

La gestión de módulos solo puede hacerle el administrador de la plataforma. La pantalla tiene una apariencia como esta:

Panel de control | Contenido | Estructura | Apariencia | Usuarios | Módulos | Configuración | Informes | Ayuda | Bienvenido, admin | Cerrar sesión

Agregar contenido | Buscar contenido | Menús | Bloques | Vistas | Contextos | Editar atajos

Descargar **módulos contribuidos** para ampliar el funcionamiento de Drupal.  
 Revise e instale regularmente **las actualizaciones disponibles** para mantener un sitio seguro y actualizado. Ejecute el **script de actualización** siempre que un módulo es actualizado.

[+ Instalar nuevo módulo](#)

**▼ NÚCLEO**

ACTIVADO	NOMBRE	VERSIÓN	DESCRIPCIÓN	OPERACIONES
<input type="checkbox"/>	<b>Aggregator</b>	7.28	Integra contenido sindicado (canales de noticias RSS, RDF y Atom).	
<input checked="" type="checkbox"/>	<b>Block</b>	7.28	Controla el montaje visual de los bloques con los que se construye una página. Los bloques son cajas de contenido que se representan en una zona o región de una página web. Necesitado por: Dashboard (activado), Nivo Slider (activado)	<a href="#">Ayuda</a> <a href="#">Permisos</a> <a href="#">Configurar</a>
<input type="checkbox"/>	<b>Blog</b>	7.28	Activa los blogs multi usuario.	
<input type="checkbox"/>	<b>Book</b>	7.28	Permite a los usuarios crear y organizar los contenidos relacionados en un índice.	
<input checked="" type="checkbox"/>	<b>Color</b>	7.28	Permite a los administradores cambiar el esquema de color de los temas compatibles con él. Necesitado por: Stylizer (desactivado)	<a href="#">Ayuda</a>
<input checked="" type="checkbox"/>	<b>Comment</b>	7.28	Permite a los usuarios comentar y discutir contenido publicado. Necesita: Text (activado), Field (activado), Field SQL storage (activado) Necesitado por: Forum (desactivado), Tracker (desactivado)	<a href="#">Ayuda</a> <a href="#">Permisos</a> <a href="#">Configurar</a>
<input type="checkbox"/>	<b>Contact</b>	7.28	Habilita el uso de formularios de contacto personal y a nivel sitio.	
<input type="checkbox"/>	<b>Content translation</b>	7.28	Permite que el contenido sea traducido a diferentes idiomas. Necesita: Locale (activado)	

Más adelante cuando hablemos de implementación de funcionalidades específicas mostraremos como configurar algunos de estos módulos.

### 3.4.3 Implementación de bloques

Drupal define el concepto de bloque como un trozo de código que muestra información. Toda la gestión de bloques se hace igualmente desde la zona de administración. De acuerdo al diseño presentado, en nuestro proyecto se ha implementado los siguientes bloques:

#### Search



#### Implementación

El buscador que se usa es el que trae el módulo del core de Drupal Search. Por tanto, para su uso lo único que hizo falta fue activar el módulo y posteriormente en su configuración asignar el bloque de búsqueda dentro de la región de cabecera.

#### Nivo Slider



## Implementación

Para resolver el requisito de los banners, opté por utilizar el módulo contribuido Nivo Slider. Su configuración permite crear imágenes de forma sencilla y aplicarles efectos de transición de muchos tipos.

Más información sobre bloques en el punto 3.6 referente al manual de administración.

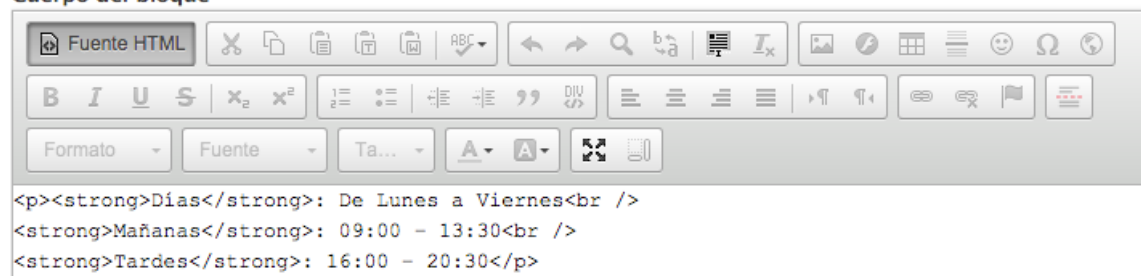
### Bloque horario atención al cliente

**Horario de atención al cliente**

**Días:** De Lunes a Viernes  
**Mañanas:** 09:00 – 13:30  
**Tardes:** 16:00 – 20:30

## Implementación

### Cuerpo del bloque \*



The screenshot shows a WYSIWYG editor interface with a toolbar and a text area. The toolbar includes options for source code, undo, redo, search, and various text and list formatting tools. The text area contains the following HTML code:

```
<p><strong>Días</strong>: De Lunes a Viernes<br />
<strong>Mañanas</strong>: 09:00 - 13:30<br />
<strong>Tardes</strong>: 16:00 - 20:30</p>
```

### Bloque contacto

**Contacto**

**Teléfono:** 692 201 219 - 951 334 586  
**Email:** [podologiadeborahgarrido@gmail.com](mailto:podologiadeborahgarrido@gmail.com)  
**Redes:** [www.facebook.com/podologiadeborahgarrido](http://www.facebook.com/podologiadeborahgarrido)

## Implementación

### Cuerpo del bloque \*



The screenshot shows a WYSIWYG editor interface with a toolbar and a text area. The toolbar includes options for source code, undo, redo, search, and various text and list formatting tools. The text area contains the following HTML code:

```
<p>692 201 219 - 951 334 586<br />
podologiadeborahgarrido@gmail.com<br />
<a href="http://www.facebook.com/podologiadeborahgarrido">Ver página Facebook</a></p>
```

## Bloque donde estamos

### ¿Dónde estamos?

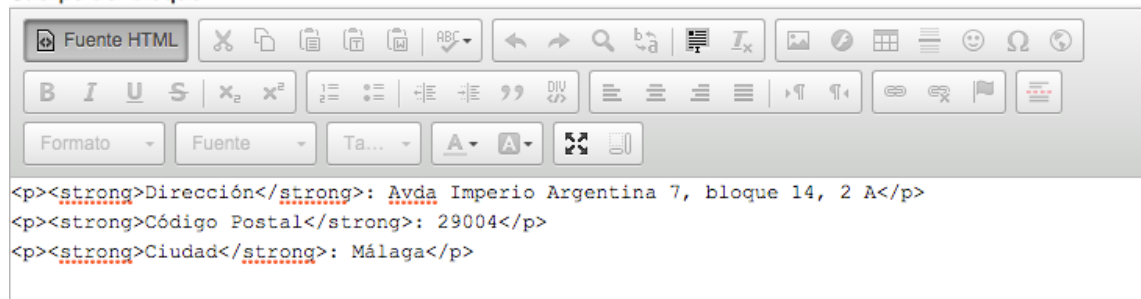
**Dirección:** Avda Imperio Argentina 7, bloque 14, 2 A

**Código Postal:** 29004

**Ciudad:** Málaga

## Implementación

### Cuerpo del bloque \*



```
<p><strong>Dirección</strong>: Avda Imperio Argentina 7, bloque 14, 2 A</p>
<p><strong>Código Postal</strong>: 29004</p>
<p><strong>Ciudad</strong>: Málaga</p>
```

## Bloque equipamiento

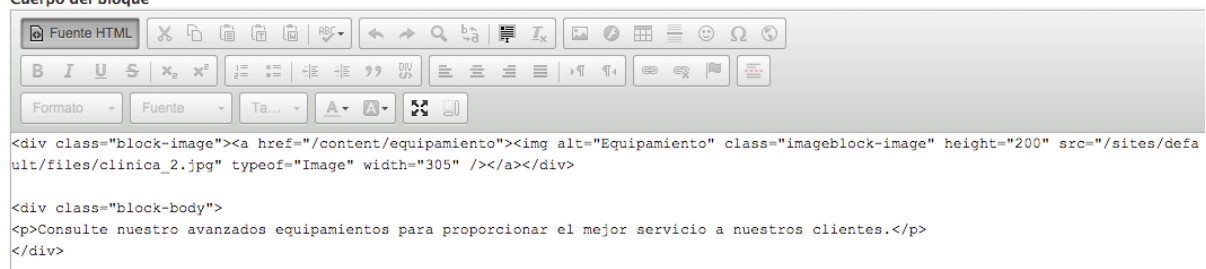
### Equipamiento



Consulte nuestro avanzados equipamientos para proporcionar el mejor servicio a nuestros clientes.

## Implementación

### Cuerpo del bloque \*



```
<div class="block-image"><a href="/content/equipamiento"></a></div>
<div class="block-body">
<p>Consulte nuestro avanzados equipamientos para proporcionar el mejor servicio a nuestros clientes.</p>
</div>
```

## Bloque servicios

**Servicios**



Proporcionamos un cuidado integral del pie. Realizamos tratamiento podológico a todos los niveles.

## Implementación

Cuerpo del bloque \*

```
<div class="block-image"><a href="/content/servicios"></a></div>
```

```
<div class="block-body">
```

```
<p>Proporcionamos un cuidado integral del pie. Realizamos tratamiento podológico a todos los niveles.</p>
```

```
</div>
```

## Bloque atención a domicilio

**Atención a domicilio**



Realizamos atención podológica a domicilio para atender a personas con movilidad reducida.

## Implementación

Cuerpo del bloque \*

```
<div class="block-image"><a href="/content/atención-domicilio"></a></div>
```

```
<div class="block-body">
```

```
<p>Realizamos atención podológica a domicilio para atender a personas con movilidad reducida.&nbsp;</p>
```

```
</div>
```

## Contenido de página principal

Inicio » Servicios


### Servicios

- DUREZAS / CALLOSIDADES
- VERRUGAS / PAPILOMAS
- ORTESIS PLANTARES
- HONGOS / MICOSIS
- UÑA ENCARNADA
- ORTESIS DE SILICONA
- PIE DIABETICO / RIESGO
- PODOLOGIA DEPORTIVA
- PODOLOGIA GERIATRICA
- PODOLOGIA INFANTIL
- VENDAJES NEUROMUSCULARES
- ATENCION A DOMICILIO

### DUREZAS / CALLOSIDADES

#### ¿Qué son?

La hiperqueratosis, vulgarmente denominada como callo o "dureza", es una zona de la piel en la que se produce una acumulación de queratina correspondiente a la compactación de células muertas inertes de la epidermis en respuesta a un estímulo que puede ser, generalmente, el roce o la fricción excesiva. Las callosidades son más comunes en la zona del pie o de la mano dependiendo la actividad o factor en el cual haya surgido el callo.



#### Tratamiento

Los callos en los pies son un signo de sobrecarga que debe ser valorado por el podólogo. El tratamiento convencional para eliminar las durezas es la quiropodia o eliminación mediante bisturí. Una vez retirada la hiperqueratosis la piel queda completamente sana, pero el factor etiológico, que es el funcionamiento anormal del pie, sigue estando presente, por lo que el problema vuelve a aparecer pasado un tiempo. Por esta razón, el tratamiento debe encaminarse no sólo a eliminar manualmente estas durezas, sino a buscar el mecanismo etiológico que las origina y compensarlo, generalmente mediante unos soportes plantares (plantillas) confeccionadas a medida por parte del podólogo.

## Implementación

Esta página de contenido varía en función de la sección que estemos consultando, por ejemplo si entramos en la Home, vemos un contenido, si entramos en ¿Quiénes somos? vemos otro y así con todas las pantallas que definan contenido en esta área.

Por ejemplo, para el caso de la página de contenidos para la sección de Servicios, tendríamos lo siguiente:

Body (Editar resumen)

```
<link href="/sites/all/themes/rubix-responsive-theme/bootstrap-combined.min.css" rel="stylesheet" /><script src="//ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script><script src="//netdna.bootstrapcdn.com/twitter-bootstrap/2.3.2/js/bootstrap.min.js"></script><script src="tabcordion.js"></script><script>$(function() {
  $('#tabcordion').tabcordion()
});
</script>
<div class="container">
<div class="row">
<div class="span12">
<div class="tabcordion tabbable">
<ul class="nav nav-tabs">
  <li class="active"><a data-target=".durezas" data-toggle="tab" href="">DUREZAS / CALLOSIDADES</a></li>
  <li><a data-target=".verrugas" data-toggle="tab" href="">VERRUGAS / PAPILOMAS</a></li>
  <li><a data-target=".plantares" data-toggle="tab" href="">ORTESIS PLANTARES</a></li>
  <li><a data-target=".hongos" data-toggle="tab" href="">HONGOS / MICOSIS</a></li>
  <li><a data-target=".encarnada" data-toggle="tab" href="">UÑA ENCARNADA</a></li>
  <li><a data-target=".silicona" data-toggle="tab" href="">ORTESIS DE SILICONA</a></li>
  <li><a data-target=".diabetico" data-toggle="tab" href="">PIE DIABETICO / RIESGO</a></li>
  <li><a data-target=".deportiva" data-toggle="tab" href="">PODOLOGIA DEPORTIVA</a></li>
</ul>
</div>
</div>
</div>
</div>
```

## Bloque lateral de consejos permanentes

### Consejos Permanentes

Además de nuestros servicios, siempre estamos dispuestos a darle cualquier tipo de consejo que pueda ayudarle en su salud.



Haga siempre caso a las siguientes recomendaciones:

- ❖ No use ningún tipo de producto que pueda dañar sus pies.
- ❖ Mantenga sus pies siempre bien aseados.
- ❖ Use un calzado cómodo.

## Implementación

### Cuerpo del bloque

```
<h3 class="color-1">Consejos Permanentes</h3>

<p>Además de nuestros servicios, siempre estamos dispuestos a darle cualquier tipo de consejo que pueda ayudarle en su salud.</p>

<p><a href="http://www.clinicapodologicazaragoza.com/consejos.html#"></a></p>

<p>Haga siempre caso a las siguientes recomendaciones:</p>

<ul>
  <li>No use ningún tipo de producto que pueda dañar sus pies.</li>
  <li>Mantenga sus pies siempre bien aseados.</li>
  <li>Use un calzado cómodo.</li>
</ul>
```

## Bloque de enlaces de interés

### Enlaces de interés

- ❖ [Enlace 1](#)
- ❖ [Enlace 2](#)
- ❖ [Enlace 3](#)
- ❖ [Enlace 4](#)
- ❖ [Enlace 5](#)
- ❖ [Enlace 6](#)
- ❖ [Enlace 7](#)

## Implementación

Cuerpo del bloque \*

```
<ul class="links-list">
<li><a href="/enlace1">Enlace 1</a></li>
<li><a href="/enlace2">Enlace 2</a></li>
<li><a href="/enlace3">Enlace 3</a></li>
<li><a href="/enlace4">Enlace 4</a></li>
<li><a href="/enlace5">Enlace 5</a></li>
<li><a href="/enlace6">Enlace 6</a></li>
<li><a href="/enlace7">Enlace 7</a></li>
</ul>
```

## Bloque location map



## Implementación

Para los mapas de geo-localización de tipo Google, se ha utilizado el módulo contribuido Location Map. Este módulo permite configurar una ubicación (mediante latitud y longitud) y automáticamente crear un punto de búsqueda sobre él. Igualmente permite añadirle información al mismo. Su pantalla de configuración es:

### LOCATION INFORMATION

#### Título

The title of the automatically generated [map page](#).

#### Address of your location \*

Enter your address separated by commas. This will be sent to Google for geocoding to determine the geographical coordinates of your location. Include any suitable from: # Street, Suburb, City, Region/State, Postcode/Zip, Country.

#### Map type

#### Map zoom level

#### Map Width

 px

#### Map Height

 px

**GEOGRAPHICAL COORDINATES**

Geographical coordinates for your location. Location map will try to obtain this information from Google using the address above. You are also able to fine-tune this by dragging the marker on the [map page](#). Under normal circumstances you would not set these coordinates manually.

**Latitude**

**Longitude**

**Marker Information**

Fuente HTML

Podología Deborah Garrido

Avda Imperio Argentina 7

692 201 219 - 951 334 586

podologiadeborahgarrido@gmail.com

[Switch to plain text editor](#)

The description that will be shown when a user clicks on the marker. If this field is empty, the address will be used.

Formato de texto Filtered HTML

## Bloque formulario contacto

**Formulario de contacto**

Nombre \*

Email \*

Asunto

Mensaje \*

## Implementación

El formulario de contacto se ha creado a partir del módulo contribuido Webform. Su configuración es muy sencilla, únicamente tenemos que crear un contenido de tipo Webform e ir asignando los campos del formulario con su correspondiente nombre y tipo. La pantalla de configuración es tal que así:

ETIQUETA	TIPO	VALUE	MANDATORY	OPERACIONES
+ Nombre	Textfield	-	<input checked="" type="checkbox"/>	Editar Clone Eliminar
+ Email	Correo electrónico	-	<input checked="" type="checkbox"/>	Editar Clone Eliminar
+ Asunto	Textfield	-	<input type="checkbox"/>	Editar Clone Eliminar
+ Mensaje	Textarea	-	<input checked="" type="checkbox"/>	Editar Clone Eliminar
+ <input type="text" value="New component name"/>	<input type="text" value="Textfield"/>		<input type="checkbox"/>	<input type="button" value="Agregar"/>

## Sitio elaborado con Drupal

Funciona con [Drupal](#)

Copyright © 2015, Podología Deborah Garrido

Theme by [Zymphonies](#)

## Implementación

Este bloque se crea en Drupal en base al tema que tengamos seleccionado. Podemos decidir si mostrarlo o no en el pie de la Web únicamente en la configuración del bloque.

## Bloque podología

**¿Qué es la Podología?**

La Podología es una rama de la medicina que tiene por objeto el estudio, diagnóstico y tratamiento de las enfermedades y alteraciones que afectan el pie. El podólogo es un especialista calificado, a través de su años de estudios y su entrenamiento, para el diagnóstico y tratamiento de diversas afecciones del pie y tobillo.

La podología engloba los siguientes aspectos:

- ❖ Promoción de la salud integral.
- ❖ Prevención de afecciones y deformidades del pie.
- ❖ Dar respuesta mediante técnicas diagnósticas y tratamientos adecuados.

## Implementación

Cuerpo del bloque \*

Formato Fuente Tamaño Fuente Color Fondo Color

```
<p>La <strong>Podología</strong>&nbsp;es una rama de la medicina que tiene por objeto el estudio, diagnóstico y tratamiento de las enfermedades y alteraciones que afectan el pie. El podólogo es un especialista calificado, a través de su años de estudios y su entrenamiento, para el diagnóstico y tratamiento de diversas afecciones del pie y tobillo.</p>
<p>La podología engloba los siguientes aspectos:</p>
<ul>
<li>Promoción de la salud integral.</li>
<li>Prevención de afecciones y deformidades del pie.</li>
<li>Dar respuesta mediante técnicas diagnósticas y tratamientos adecuados.</li>
</ul>
```

## Bloque mapa web



## Implementación

Cuerpo del bloque \*

```
<ul>
  <li><a href="/home " >Inicio</a></div></li>
  <li><a href=" /content/quienes-somos">¿Quiénes somos?</a></li>
  <li><a href=" /content/equipamiento">Equipamiento</a></li>
  <li><a href=" /content/servicios">Servicios</a></li>
  <li><a href=" /content/ofertas">Ofertas</a></li>
  <li><a href=" /content/formulario-contacto">Contacto</a></li>
  <li><a href=" /content/buenas-practicas">Buenas prácticas</a></li>
  <li><a href="/content/locationmap">¿Cómo llegar?</a></li>
</ul>
```

## Bloque galería de imágenes



## Implementación

Para la galería de imágenes, hemos creado un tipo de contenido Imagen que representará cada una de las imágenes. Posteriormente mediante el uso de las vistas de Drupal, hemos creado un listado de imágenes para poder tener un listado dinámico. Las vistas en Drupal tiene un algo grado de configuración, por tipo de contenido, campos a mostrar, ordenación específica, etc...

Mención especial también para el módulo Lightbox que hemos tenido que utilizar para la visualización de las imágenes a modo de popup y para poder iterar por ellas de forma intuitiva.

A modo de resumen, podemos decir que con la utilización de bloques la idea es tener unidades independientes de código que muestran información con el objetivo de luego poder ubicarlas en las distintas zonas que se definen en la Web, como si fuese una especie de puzzle donde vamos encajando las piezas.

Por ejemplo, en nuestra zona de cabecera pueden ir los bloques búsqueda y nivo slider, en la zona central tendremos los bloques de horario atención al cliente, contacto, donde estamos, equipamiento, servicios, atención a domicilio, contenido de página principal, location map y formulario de contacto, mientras que en el pie tenemos el bloque de podología, mapa web, galería de imágenes o sitio web creado con Drupal.

Evidentemente no todos los bloques se muestran en las mismas pantallas, por lo que hay que establecer un criterio para decir que bloques se muestran en que pantallas y para eso tenemos que hablar de los contextos de Drupal.

Las zonas (cabecera, central, vertical, pie...) se definen en la configuración del tema (apariciencia gráfica) usado. Mas adelante en la sección de adaptación de apariciencia gráfica se hablará de cómo configurar estas zonas o regiones.

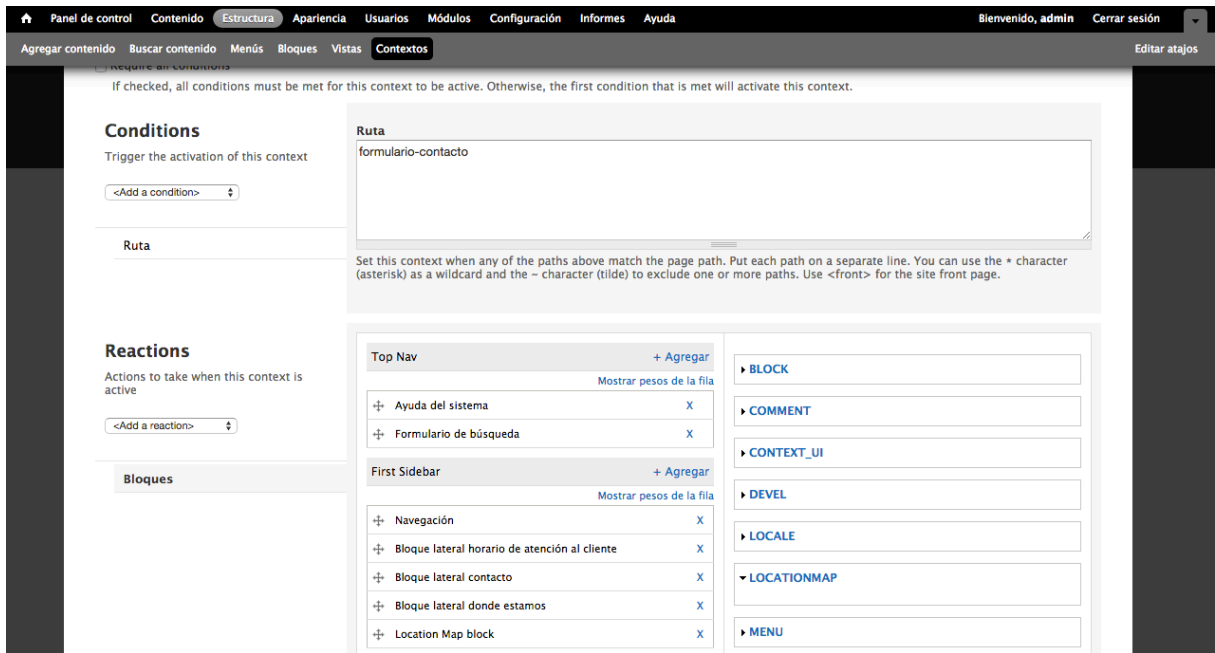
Mas información sobre vistas en el manual de administración, sección 3.6.

#### **3.4.4 Creación de contextos**

Como acabamos de ver en la fase de diseño, nuestra aplicación va a implementar cuatro pantallas tipo, una Home, una para el detalle de contenidos, otra para el formulario de contacto y una última para el mapa de geo-localización, de manera que todas las pantallas que tengamos en nuestra aplicación se ajustarán siempre a un tipo de estos cuatro.

Drupal puede definir distintos contextos mediante el uso del módulo Context que resuelve perfectamente esta necesidad. El módulo funciona de forma sencilla, primero se define la condición bajo la cuál queremos que el contexto actúe (por ejemplo la url de las páginas que queremos que se vean afectadas por el contexto) y posteriormente se definen los bloques de información que queremos mostrar en él.

La pantalla de configuración de contextos es:



### 3.4.5 Adaptación del tema gráfico

El tema gráfico era uno de los temas mas importantes en el proyecto. El cliente tenía claro que los colores debían ser parecidos a la decoración de la clínica, donde predominan los colores, blancos, grisáceos y morados.

Para hacer el diseño, me basé en un tema de Drupal que se llama Rubix Responsive Theme porque se adaptaba a lo que estaba buscando y porque las modificaciones que tenía que realizar sobre él no serían demasiado costosas.

Lo primero de todo era definir las áreas o regiones que necesitaríamos en nuestro portal Web. Para ello hay que ir al fichero "rubix\_responsive\_theme.info". Aquí se define las hojas de estilo y Javascripts que usa el tema y lo mas importante, también se definen las regiones. Estas regiones son las que se utilizan para asignar los bloques.

```

name = Rubix responsive theme
description = rubix responsive theme is a responsive theme its perfect for all sorts of corporate
core = 7.x

engine = phptemplate
stylesheets[all][] = style.css
stylesheets[all][] = media.css
scripts[] = js/custom.js

regions[top_nav] = Top Nav
regions[sidebar_first] = First Sidebar
regions[sidebar_second] = Second Sidebar
regions[slideshow] = Slideshow
regions[front_welcome] = Front Welcome
regions[top_first] = First top
regions[top_second] = Second top
regions[top_third] = Third top
regions[image_first] = First Image Block
regions[image_second] = Second Image Block
regions[image_third] = Third Image Block
regions[image_forth] = Forth Image Block
regions[image_fifth] = Fifth Image Block
regions[image_sixth] = Sixth Image Block
regions[content] = Content
regions[content_top] = Content Top
regions[footer] = Footer
regions[footer_first] = First Bottom
regions[footer_second] = Second Bottom
regions[footer_third] = Third Bottom
features[] = logo
features[] = name
features[] = slogan
features[] = node_user_picture
features[] = comment_user_picture
features[] = comment_user_verification
features[] = favicon
features[] = main_menu
features[] = secondary_menu
settings[breadcrumbs] = 1
settings[social_links] = 1
settings[twitter_username] = "shanidkv"
settings[facebook_username] = "shanidkv"

```

Los cambios que se hicieron a nivel gráfico afectaban básicamente a los colores de las hojas de estilos y a las coordenadas de posicionamiento de los estilos de algunos bloques. Las modificaciones se hicieron sobre el fichero de estilos del tema "style.css". Algunos de ellos fueron:

#### Cambio sobre los enlaces

```

a {
  color: #c92ccc;
  text-decoration:none;
}
a:hover {
  /*color: #93C90F;*/
  color: #727272;
}

```

#### Cambios en las tablas

```

table {

```

```

/*background: #ececec;*/
border: 0px solid #bbbbbb;
}
td {
/*background: #ebebeb;*/
color: #747474;
text-align: left;
padding-left: 4px;
border-right: 0px solid #c8c7c7;
border-bottom: 0px solid #c8c7c7;
}

```

### Cambios en los menús

```

#main-menu a {
display: block;
color: rgb(153, 153, 153);
text-align: center;
/*padding: 13px 23px;*/
padding: 13px 21.9px;
text-decoration: none;
font-size: 12px;
text-transform: uppercase;
font-weight: bold;
background: url(images/menu_bg_a.png);
}
#main-menu a:hover,
#main-menu li a.active{
color: #fff;
text-decoration: none;
background: #c92ccc;
}

```

### Cambios en las regiones de contenidos

```

.front .content-sidebar-wrap{
/*margin-top: 5px;*/
margin-top: -15px;
}
.front .content-sidebar-wrap article{
margin: 0;
}
.content-sidebar-wrap #content {
display: inline;
margin-bottom: 2.7%;
float: right;
width: 100%;
/*margin-right: -12px;*/
}

```

```
margin-top: 20px;
border-radius:15px;
background: #F4E8FF;
padding-top:10px;
}
```

#### Cambios en las migas de pan (path)

```
.breadcrumb {
margin:0 20px;
/*padding-top:20px;*/
padding-top:10px;
margin-bottom:10px;
border-bottom:#eee 1px solid;
}
```

#### Cambios en el formulario de búsqueda

```
#block-search-form .form-item-search-block-form input { float: left; font-size: 1em;
height: 1.143em; margin-right: 5px; width: 9em; margin-top:10px;}
#search-block-form input.form-submit,
#search-form input.form-submit {
margin-left: 0; margin-right: 0; height: 25px; width: 34px; padding: 0; cursor: pointer;
text-indent: -9999px; border-color: #e4e4e4 #d2d2d2 #b4b4b4; background:
url(images/search-button.png) no-repeat center top; overflow: hidden;
margin-top:10px;
}
#search-form .form-item-keys label { display: none; }
```

#### Cambios generales

```
#user-login-form .form-text{ width: 90%; }
.region-sidebar-first{
border: 0px solid #363636;
padding: 10px 10px;
/*margin-top: 20px;*/
margin-top: 10px;
background: #F8F8F8;
width: 192px;
border-radius: 15px 15px 15px 15px;
padding-top: 1px;
}
.content #locationmap_map{
margin-left:80px;
border-radius:15px;
}
ul.col3
{
PADDING-RIGHT: 0px;
```

```

PADDING-LEFT: 0px;
FLOAT: left;
PADDING-BOTTOM: 0px;
MARGIN: 0px 10px 10px 20px;
WIDTH: 100%;
PADDING-TOP: 0px;
LIST-STYLE-TYPE: none
}
ul.col3 li
{
DISPLAY: inline;
FLOAT: left;
WIDTH: 30%;
}
.content .block-image a img {
padding:3px;
box-shadow:0 0 3px #EEE;
background:#FFF;
border:#EEE 0px solid;
}
.content .block-image a: hover img {
background:#F4E8FF;
}
.rss-class{
margin-left:20px;
margin-top:1px;
}
.facebook-class{
margin-left:2px;
margin-top:1px;
}
.tab-pane img{
width:50%;
height:50%;
float: right;
border-radius: 15px;
padding:10px;
margin-top:15px;
}

```

### 3.4.6 Pantallas con diseño final

Una vez ya terminada la fase de implementación, creados todos los bloques, contextos, vistas, páginas... y teniendo la adaptación gráfica finalizada, la aplicación Web ya tiene un aspecto mucho mas atractivo. A continuación vemos como quedan las cuatro pantallas tipo que detallamos anteriormente, Home, Pantalla de detalle de contenido, Pantalla de formulario de contacto y Pantalla de mapa de geo-localización.


# Home

## Pantalla de portada de la aplicación:

### Podología Deborah Garrido

Cuidado Integral del pie

- INICIO
- ¿QUIÉNES SOMOS?
- EQUIPAMIENTO
- SERVICIOS
- OFERTAS
- CONTACTO
- BUENAS PRÁCTICAS
- ¿CÓMO LLEGAR?



Podología deportiva

#### Horario de atención al cliente

Días: De Lunes a Viernes  
Mañanas: 09:00 – 13:30  
Tardes: 16:00 – 20:30

#### Contacto

Teléfono: 692 201 219 - 951 334 586  
Email: [podologiadeborahgarrido@gmail.com](mailto:podologiadeborahgarrido@gmail.com)  
Redes: [www.facebook.com/podologiadeborahgarrido](http://www.facebook.com/podologiadeborahgarrido)

#### ¿Dónde estamos?

Dirección: Avda Imperio Argentina 7, bloque 14, 2 A  
Código Postal: 29004  
Ciudad: Málaga

### Podología Déborah Garrido

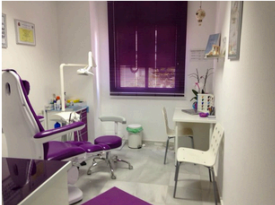
Podología Deborah Garrido proporciona el **mejor servicio a sus clientes** con el objetivo de alcanzar su **mayor grado de satisfacción**. Disponemos de los mas **avanzados equipamientos y tecnología** para detectar y tratar cualquier anomalía que esté relacionada con el pie. Poseemos los **certificados de calidad** necesarios para **garantizar el mejor servicio**.

Entre los **servicios principales que prestamos** destacan:

- DUREZAS / CALLOSIDADES
- HONGOS / MICOSIS
- PIE DIABETICO / RIESGO
- PODOLOGIA INFANTIL
- VERRUGAS / PAPILOMAS
- UÑA ENCARNADA
- PODOLOGIA DEPORTIVA
- ATENCIÓN A DOMICILIO
- ORTESIS PLANTARES
- ORTESIS DE SILICONA
- PODOLOGIA GERIATRICA


Puede obtener un mayor detalle de nuestro equipamiento, servicios y calidad dirigiéndose a las secciones correspondientes de la Web.

#### Equipamiento




Consulte nuestro avanzados equipamientos para proporcionar el mejor servicio a nuestros clientes.

#### Servicios



Proporcionamos un cuidado integral del pie. Realizamos tratamiento podológico a todos los niveles.

#### Atención a domicilio



Realizamos atención podológica a domicilio para atender a personas con movilidad reducida.

#### ¿Qué es la Podología?

La Podología es una rama de la medicina que tiene por objeto el estudio, diagnóstico y tratamiento de las enfermedades y alteraciones que afectan el pie. El podólogo es un especialista cualificado, a través de su años de estudios y su entrenamiento, para el diagnóstico y tratamiento de diversas afecciones del pie y tobillo.

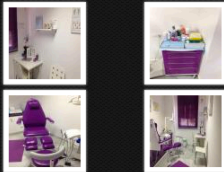
La podología engloba los siguientes aspectos:

- Promoción de la salud integral.
- Prevención de afecciones y deformidades del pie.
- Dar respuesta mediante técnicas diagnósticas y tratamientos adecuados.

#### Mapa Web

- Inicio
- ¿Quiénes somos?
- Equipamiento
- Servicios
- Ofertas
- Contacto
- Buenas prácticas
- ¿Cómo llegar?

#### Galería de Imágenes



Funciona con [Drupal](#)

Copyright © 2015, Podología Deborah Garrido

Theme by [Zymphonies](#)

## Pantalla de contenidos

Esta pantalla cambia en función del contenido que estemos mostrando, por ejemplo para el caso de la página de servicios, el resultado sería este:

# Podología Deborah Garrido

Cuidado integral del pie

- INICIO
- ¿QUIÉNES SOMOS?
- EQUIPAMIENTO
- SERVICIOS**
- OFERTAS
- CONTACTO
- BUENAS PRÁCTICAS
- ¿CÓMO LLEGAR?

### Consejos Permanentes

Además de nuestros servicios, siempre estamos dispuestos a darle cualquier tipo de consejo que pueda ayudarle en su salud.



Haga siempre caso a las siguientes recomendaciones:

- No use ningún tipo de producto que pueda dañar sus pies.
- Mantenga sus pies siempre bien aseados.
- Use un calzado cómodo.

#### Enlaces de interés

- Enlace 1
- Enlace 2
- Enlace 3
- Enlace 4
- Enlace 5
- Enlace 6
- Enlace 7

### Servicios

- DUREZAS / CALLOSIDADES**
- VERRUGAS / PAPILOMAS
- ORTESIS PLANTARES
- HONGOS / MICOSIS
- UÑA ENCARNADA
- ORTESIS DE SILICONA
- PIE DIABETICO / RIESGO
- PODOLOGIA DEPORTIVA
- PODOLOGIA GERIATRICA
- PODOLOGIA INFANTIL
- VENDAJES NEUROMUSCULARES
- ATENCION A DOMICILIO

#### DUREZAS / CALLOSIDADES

##### ¿Qué son?

La hiperqueratosis, vulgarmente denominada como callo o "dureza", es una zona de la piel en la que se produce una acumulación de queratina correspondiente a la compactación de células muertas inertes de la epidermis en respuesta a un estímulo que puede ser, generalmente, el roce o la fricción excesiva. Las callosidades son más comunes en la zona del pie o de la mano dependiendo la actividad o factor en el cual haya surgido el callo.



##### Tratamiento

Los callos en los pies son un signo de sobrecarga que debe ser valorado por el podólogo. El tratamiento convencional para eliminar las durezas es la quiropodia o eliminación mediante bisturí. Una vez retirada la hiperqueratosis la piel queda completamente sana, pero el factor etiológico, que es el funcionamiento anormal del pie, sigue estando presente, por lo que el problema vuelve a aparecer pasado un tiempo. Por esta razón, el tratamiento debe encaminarse no sólo a eliminar manualmente estas durezas, sino a buscar el mecanismo etiológico que las origina y compensarlo, generalmente mediante unos soportes plantares (plantillas) confeccionadas a medida por parte del podólogo.

### ¿Qué es la Podología?

La Podología es una rama de la medicina que tiene por objeto el estudio, diagnóstico y tratamiento de las enfermedades y alteraciones que afectan el pie. El podólogo es un especialista calificado, a través de su años de estudios y su entrenamiento, para el diagnóstico y tratamiento de diversas afecciones del pie y tobillo.

La podología engloba los siguientes aspectos:

- Promoción de la salud integral.
- Prevención de afecciones y deformidades del pie.
- Dar respuesta mediante técnicas diagnósticas y tratamientos adecuados.

### Mapa Web

- Inicio
- ¿Quiénes somos?
- Equipamiento
- Servicios
- Ofertas
- Contacto
- Buenas prácticas
- ¿Cómo llegar?

### Galería de imágenes



Funciona con [Drupal](#)  
Copyright © 2015, Podología Deborah Garrido  
Theme by [Zymphonies](#)

## Formulario de contacto

Pantalla de contacto que los clientes usan para poder contactar con la clínica:

### Podología Deborah Garrido

Cuidado integral del pie

INICIO ¿QUIÉNES SOMOS? EQUIPAMIENTO SERVICIOS OFERTAS **CONTACTO** BUENAS PRÁCTICAS ¿CÓMO LLEGAR?

#### Horario de atención al cliente

Días: De Lunes a Viernes  
Mañanas: 09:00 – 13:30  
Tardes: 16:00 – 20:30


#### Contacto

692 201 219 - 951 334 586  
[podologiadeborahgarrido@gmail.com](mailto:podologiadeborahgarrido@gmail.com)  
[Ver página Facebook](#)

#### ¿Dónde estamos?

Dirección: Avda Imperio Argentina  
7, bloque 14, 2 A  
Código Postal: 29004  
Ciudad: Málaga

#### ¿Cómo llegar?



Inicio » Formulario de contacto

#### Formulario de contacto

Nombre \*

Email \*

Asunto

Mensaje \*

Enviar

#### ¿Qué es la Podología?

La Podología es una rama de la medicina que tiene por objeto el estudio, diagnóstico y tratamiento de las enfermedades y alteraciones que afectan el pie. El podólogo es un especialista calificado, a través de su años de estudios y su entrenamiento, para el diagnóstico y tratamiento de diversas afecciones del pie y tobillo.

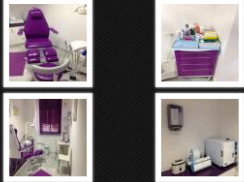
La podología engloba los siguientes aspectos:

- ❖ Promoción de la salud integral.
- ❖ Prevención de afecciones y deformidades del pie.
- ❖ Dar respuesta mediante técnicas diagnósticas y tratamientos adecuados.

#### Mapa Web

- ❖ Inicio
- ❖ ¿Quiénes somos?
- ❖ Equipamiento
- ❖ Servicios
- ❖ Ofertas
- ❖ Contacto
- ❖ Buenas prácticas
- ❖ ¿Cómo llegar?

#### Galería de imágenes



Funciona con [Drupal](#)  
Copyright © 2015, Podología Deborah Garrido  
Theme by [Zymphonies](#)

## Mapa de geo-localización

Pantalla de localización de la clínica.

### Podología Deborah Garrido

Cuidado integral del pie

Inicio | ¿QUIÉNES SOMOS? | EQUIPAMIENTO | SERVICIOS | OFERTAS | CONTACTO | BUENAS PRÁCTICAS | **¿CÓMO LLEGAR?**

#### Horario de atención al cliente

Días: De Lunes a Viernes  
Mañanas: 09:00 – 13:30  
Tardes: 16:00 – 20:30

#### Contacto

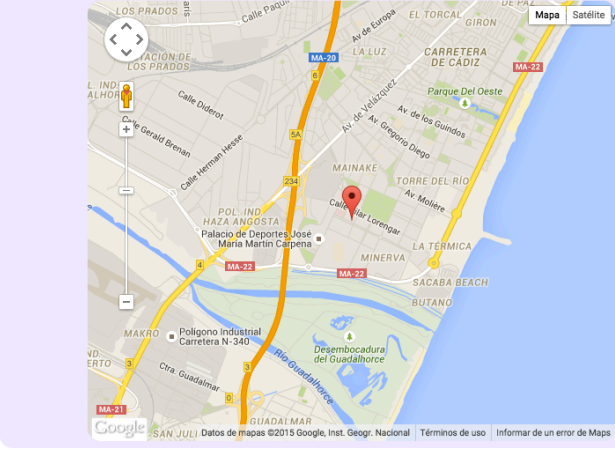
692 201 219 - 951 334 586  
podologiadeborahgarrido@gmail.com  
Ver página Facebook

#### ¿Dónde estamos?

Dirección: Avda Imperio Argentina 7, bloque 14, 2 A  
Código Postal: 29004  
Ciudad: Málaga

#### ¿Cómo llegar?

Puedes navegar por el mapa para encontrar nuestra ubicación.



Inicio » ¿Cómo llegar?

#### ¿Qué es la Podología?

La Podología es una rama de la medicina que tiene por objeto el estudio, diagnóstico y tratamiento de las enfermedades y alteraciones que afectan el pie. El podólogo es un especialista calificado, a través de su años de estudios y su entrenamiento, para el diagnóstico y tratamiento de diversas afecciones del pie y tobillo.


La podología engloba los siguientes aspectos:

- Promoción de la salud integral.
- Prevención de afecciones y deformidades del pie.
- Dar respuesta mediante técnicas diagnósticas y tratamientos adecuados.

#### Mapa Web

- Inicio
- ¿Quiénes somos?
- Equipamiento
- Servicios
- Ofertas
- Contacto
- Buenas prácticas
- ¿Cómo llegar?

#### Galería de imágenes

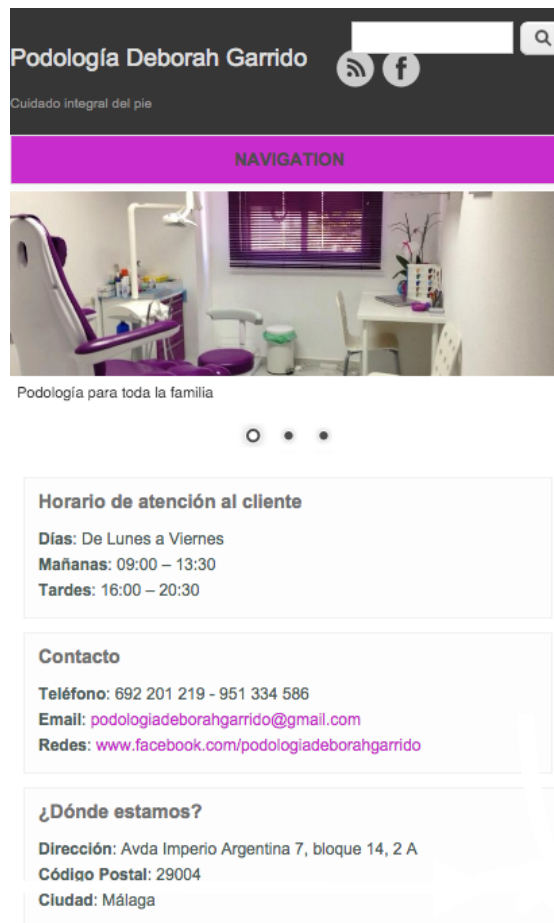


Funciona con [Drupal](#)  
Copyright © 2015, Podología Deborah Garrido  
Theme by [Zymphonies](#)

43

## Diseño responsivo

Diseño de la aplicación visto en dispositivos móviles:



### Podología Déborah Garrido

Podología Deborah Garrido proporciona el **mejor servicio a sus clientes** con el objetivo de alcanzar su **mayor grado de satisfacción**. Disponemos de los mas **avanzados equipamientos y tecnología** para detectar y tratar cualquier anomalía que esté relacionada con el pie. Poseemos los **certificados de calidad** necesarios para **garantizar el mejor servicio**.

Entre los **servicios principales que prestamos** destacan:

❖ DUREZAS / CALLOSIDADES	❖ VERRUGAS / PAPILOMAS
❖ ORTESIS PLANTARES	❖ HONGOS / MICOSIS
❖ UÑA ENCARNADA	❖ ORTESIS DE SILICONA
❖ PIE DIABETICO / RIESGO	❖ PODOLOGIA DEPORTIVA
❖ PODOLOGIA GERIATRICA	❖ PODOLOGIA INFANTIL
❖ ATENCIÓN A DOMICILIO	

Puede obtener un mayor detalle de nuestro equipamiento, servicios y calidad dirigiéndose a las secciones correspondientes de la Web.

### Equipamiento



Consulte nuestro avanzados equipamientos para proporcionar el mejor servicio a nuestros clientes.

### Servicios



Proporcionamos un cuidado integral del pie. Realizamos tratamiento podológico a todos los niveles.

### Atención a domicilio



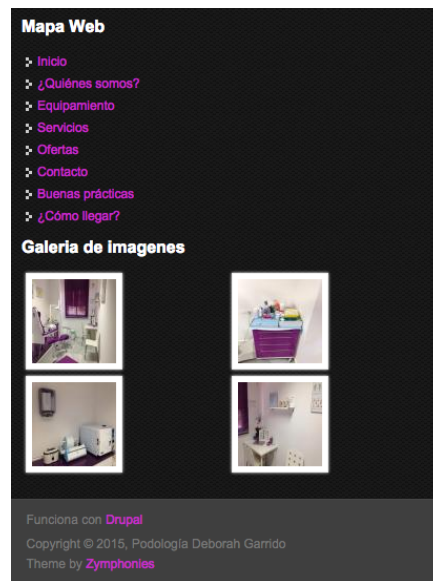
Realizamos atención podológica a domicilio para atender a personas con movilidad reducida.

## ¿Qué es la Podología?

La Podología es una rama de la medicina que tiene por objeto el estudio, diagnóstico y tratamiento de las enfermedades y alteraciones que afectan el pie. El podólogo es un especialista calificado, a través de su años de estudios y su entrenamiento, para el diagnóstico y tratamiento de diversas afecciones del pie y tobillo.

La podología engloba los siguientes aspectos:

- ❖ Promoción de la salud integral.
- ❖ Prevención de afecciones y deformidades del pie.
- ❖ Dar respuesta mediante técnicas diagnósticas y tratamientos adecuados.



### 3.5 Pruebas

#### Introducción

Las pruebas de software se refieren a las investigaciones técnicas cuyo objetivo es proporcionar información objetiva e independiente sobre la calidad del producto a la parte interesada, en este caso nuestro cliente. Es una actividad que pretende asegurar la calidad de nuestro producto.

Existen muchos tipos de pruebas, pruebas automáticas, manuales, funcionales, no funcionales, etc... pero nosotros en este caso vamos a centrarnos en pruebas automáticas y funcionales que testean la correcta construcción de un sitio Web de acuerdo a los estándares que Drupal establece.

#### Aplicación

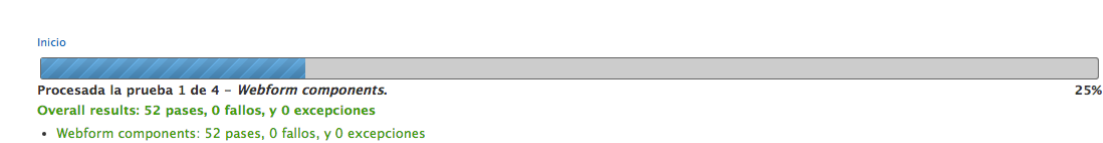
Para ello vamos a utilizar el módulo SimpleTest. Después de instalarlo, podemos acceder a su pantalla de configuración para darnos cuenta de la gran cantidad de pruebas automáticas que es capaz de lanzar sobre todas las estructuras de datos que tengamos en nuestro sitio Web. Por ejemplo, podemos lanzar pruebas automáticas sobre los menús, bloques, vistas, contenidos, etc...

La pantalla de configuración de las pruebas es:

PRUEBA	DESCRIPCIÓN
<input type="checkbox"/> ▶ User	
<input type="checkbox"/> ▶ Views	
<input type="checkbox"/> ▶ Views Handlers	
<input type="checkbox"/> ▶ Views Modules	
<input type="checkbox"/> ▶ Views Plugins	
<input type="checkbox"/> ▶ Views UI	
<input checked="" type="checkbox"/> ▼ Webform	
<input checked="" type="checkbox"/> Webform	Checks global Webform settings and content types.
<input checked="" type="checkbox"/> Webform components	Add and remove components from a webform.
<input checked="" type="checkbox"/> Webform permissions	Create webforms and check editing and access permissions.
<input checked="" type="checkbox"/> Webform submission	Submits a sample webform and checks the database integrity.
<input type="checkbox"/> ▶ XML-RPC	

Ejecutar pruebas

Para el caso anterior hemos indicado que queremos lanzar pruebas automáticas sobre todos los bloques creados en nuestro sitio Web. Cuando lanzamos las pruebas debemos esperar a que Drupal recorra todos los bloques para confirmar que no hay ningún problema:



Al final del test recibimos un informe detallado de cómo han ido todas las pruebas:

MENSAJE	GRUPO	NOMBRE DE ARCHIVO	LÍNEA	FUNCIÓN	ESTADO
Enabled modules: <i>webform, profile</i>	Other	webform.test	18	WebformTestCase->setUp()	✓
Created role of name: <i>fzCwi10p</i> , id: 4	Role	webform.test	55	WebformTestCase->setUp()	✓
Created permissions: access content, access own webform submissions	Role	webform.test	55	WebformTestCase->setUp()	✓
User created with name <i>vyXhjd1y</i> and pass <i>iq4Tqf9EoK</i>	User login	webform.test	55	WebformTestCase->setUp()	✓
Created role of name: <i>nAcLgpkz</i> , id: 5	Role	webform.test	55	WebformTestCase->setUp()	✓
Created permissions: access content, edit own webform submissions	Role	webform.test	55	WebformTestCase->setUp()	✓
User created with name <i>m1OG07kS</i> and pass <i>oqJf6oEcSS</i>	User login	webform.test	55	WebformTestCase->setUp()	✓
Created role of name: <i>eYfY75IY</i> , id: 6	Role	webform.test	55	WebformTestCase->setUp()	✓
Created permissions: access content, create webform content, edit own webform content, access all webform results	Role	webform.test	55	WebformTestCase->setUp()	✓
User created with name <i>mQYqTJ6t</i> and pass <i>VR77xtiYx2</i>	User login	webform.test	55	WebformTestCase->setUp()	✓
Created role of name: <i>bLo10Zmj</i> , id: 7	Role	webform.test	55	WebformTestCase->setUp()	✓
Created permissions: access content, administer nodes, create webform content, edit any webform content, access all webform results, edit all webform submissions, delete all webform submissions	Role	webform.test	55	WebformTestCase->setUp()	✓
User created with name <i>aHLKehMd</i> and pass <i>6aZrh4pLpJ</i>	User login	webform.test	55	WebformTestCase->setUp()	✓

De esta manera podemos hacer pruebas rápidas con Drupal y asegurarnos de que no tenemos inconsistencias ni comportamientos anómalos. Además de estas pruebas también realicé varios test unitarios para comprobar las principales funcionalidades de la aplicación. Los recorridos que definí se referían sobre todo a la creación, eliminación y edición de contenidos, menús y listados.

## **3.6 Documentación**

### Introducción

La documentación del diseño del software tiene como objetivo facilitar la mejora y mantenimiento de un proyecto a lo largo de su desarrollo. Esta documentación no tiene porque ser sólo de las funcionalidades de la aplicación, también puede entrar al nivel de detallar APIs utilizadas en el caso de que haya comunicaciones con servicios o aplicaciones externas.

### Aplicación

En nuestro caso, al ser un proyecto basado en el CMS Drupal, no necesitamos una documentación en cuanto a arquitectura del sistema, ya que toda esa parte siempre es común para todos los proyectos de Drupal y se puede encontrar mucha documentación por la Web.

Lo que si vamos a incluir es un pequeño manual de usuario administrador con el objetivo de que el cliente pueda también en la medida de lo posible gestionar por si mismo su sitio Web. Hablaremos de gestión de menús, contextos, nivo slider, tipos de contenido, vistas, bloques y usuarios.

#### **3.6.1 Gestión de menús**

Los menús son un elemento básico de Drupal ya que permiten definir la estructura de niveles de los contenidos. Al igual que el resto de las opciones que vamos a ver, para ir a ellas tenemos que ir a la sección de “Estructuras”.

Drupal define por defecto varios menús (desarrollo, navegación, gestión...) pero el que a nosotros nos interesa es el de Menú principal.

+ Añadir enlace

Mostrar pesos de la fila

ENLACE DEL MENU	ACTIVADO	OPERACIONES	
+ Inicio	<input checked="" type="checkbox"/>	editar	eliminar
+ ¿Quiénes somos?	<input checked="" type="checkbox"/>	editar	eliminar
+ Equipamiento	<input checked="" type="checkbox"/>	editar	eliminar
+ Servicios	<input checked="" type="checkbox"/>	editar	eliminar
+ Ofertas	<input checked="" type="checkbox"/>	editar	eliminar
+ Contacto	<input checked="" type="checkbox"/>	editar	eliminar
+ Buenas prácticas	<input checked="" type="checkbox"/>	editar	eliminar
+ ¿Cómo llegar?	<input checked="" type="checkbox"/>	editar	eliminar

Guardar configuración

Cómo se puede apreciar en la imagen, añadir o quitar enlaces es tan sencillo como pulsar en “Añadir enlace” o directamente pulsar en “eliminar”. También tenemos el botón “editar”.

### 3.6.2 Gestión de contextos

Ya vimos anteriormente el concepto de contexto. Un contexto en Drupal permite definir pantallas tipo con el objetivo de poder indicar que bloques queremos mostrar en cada una de ellas. En nuestra aplicación tenemos los siguientes contextos:

+ Agregar + Importar

Storage: - All - Activado: - All - Buscar:

NOMBRE	STORAGE	OPERACIONES				
<i>contacto</i>						
contacto Contexto para la página del formulario de contacto	Normal	Editar	Desactivar	Eliminar	Clone	Exportar
<i>home</i>						
home Contexto para la home	Normal	Editar	Desactivar	Eliminar	Clone	Exportar
<i>mapa</i>						
mapa Contexto para la página del mapa	Normal	Editar	Desactivar	Eliminar	Clone	Exportar
<i>página</i>						
detalle_pagina Contexto para las páginas en general	Normal	Editar	Desactivar	Eliminar	Clone	Exportar
<i>servicios</i>						
servicios Contexto para la página de servicios	Normal	Editar	Desactivar	Eliminar	Clone	Exportar

Mediante los botones de “Agregar” añadimos nuevos, de igual forma que también tenemos botones para “Editar”, “Desactivar”, “Eliminar”...

Dentro del detalle de un contexto en concreto, definimos el nombre, tags, descripción y lo mas importante, las condiciones y las reacciones. O lo que es lo mismo, bajo que url queremos que se ejecute el contexto y que bloques queremos que se muestren en él.

**Nombre \***  
  
 The unique ID for this context.

**Tag**  
  
 Example: t:heme  
 A tag to group this context with others.

**Descripción**  
  
 The description of this context definition.

**Require all conditions**

If checked, all conditions must be met for this context to be active. Otherwise, the first condition that is met will activate this context.

### Conditions

Trigger the activation of this context

<Add a condition>

Ruta

Ruta

<front>

Set this context when any of the paths above match the page path. Put each path on a separate line. You can use the \* character (asterisk) as a wildcard and the - character (tilde) to exclude one or more paths. Use <front> for the site front page.

### Reactions

Actions to take when this context is active

<Add a reaction>

Bloques

<p><b>Top Nav</b> + Agregar Mostrar pesos de la fila</p> <p>⊕ Ayuda del sistema X</p>	<p>▶ <b>BLOCK</b></p>
<p><b>First Sidebar</b> + Agregar Mostrar pesos de la fila</p> <p>⊕ Navegación X</p>	<p>▶ <b>COMMENT</b></p>
<p><b>Second Sidebar</b> + Agregar Mostrar pesos de la fila</p>	<p>▶ <b>CONTEXT_UI</b></p>
<p><b>Slideshow</b> + Agregar Mostrar pesos de la fila</p> <p>⊕ Nivo Slider X</p>	<p>▶ <b>DEVEL</b></p>
<p><b>Front Welcome</b> + Agregar Mostrar pesos de la fila</p>	<p>▶ <b>LOCALE</b></p>
<p><b>First top</b> + Agregar Mostrar pesos de la fila</p> <p>⊕ Bloque top horario de atención al cliente X</p>	<p>▶ <b>LOCATIONMAP</b></p>
<p><b>Second top</b> + Agregar Mostrar pesos de la fila</p> <p>⊕ Bloque top contacto X</p>	<p>▶ <b>MENU</b></p>
	<p>▶ <b>NIVO_SLIDER</b></p>
	<p>▶ <b>NODE</b></p>
	<p>▶ <b>SEARCH</b></p>
	<p>▶ <b>SHORTCUT</b></p>
	<p>▶ <b>SUPERFISH</b></p>
	<p>▶ <b>SYSTEM</b></p>

Añadir un bloque a un contexto es tan sencillo como seleccionar el tipo de bloque queremos añadir de la columna de la derecha y a continuación pulsar en el botón “Agregar” de la región donde lo vamos a insertar.

### 3.6.3 Gestión Nivo slider

Este módulo de Drupal nos permite gestionar banners en la plataforma. Es tan sencillo como ir creando imágenes y publicándolas, tal como se muestra en la siguiente imagen:

Configure slider content.

Upload a new slide image

Seleccionar archivo Ningún archivo seleccionado

NOMBRE	PUBLICADO
+ Image 1: Consulta de podología Deborah Garrido	<input checked="" type="checkbox"/>
+ Image 2: Podología deportiva	<input checked="" type="checkbox"/>
+ Image 3: Podología para toda la familia	<input checked="" type="checkbox"/>

**Título**  
Consulta de podología Deborah Garrido  
The title is used as alternative text for the slide image.

**Descripción**  
Consulta de podología Deborah Garrido

Formato de texto Plain text   
[Más información sobre los formatos de texto](#)

- No se permiten etiquetas HTML.
- Las direcciones de las páginas web y las de correo se convierten en enlaces automáticamente.
- Salto automático de líneas y de párrafos.

The description will be displayed with the slide image.

**Link slide to URL**

Specify a path or an absolute URL. An example path is *blog* for the blog page. An example absolute URL is *http://drupal.org* for the Drupal website. *<front>* is the front page.

**Show slide on specific pages**

Especificar páginas usando sus rutas. Escriba una ruta por línea. El carácter "\*" se usa como comodín. Ejemplos de rutas son *blog* para la página de blog y *blog/\** para cada blog personal. *<front>* es la página principal.

A partir de aquí lo único que queda es añadir el efecto de visualización que queremos en la transición, que se hace también desde la pantalla anterior de configuración y tiene este aspecto:

**Transition**

- Default -

Select a transition. Selecting an option other than *- Default -* will force this slide to use the selected transition every time it appears. It overrides all other effect settings.

### 3.6.4 Gestión tipos de contenido

Los contenidos son los elementos de información que Drupal muestra. Podemos definir nuevos tipos de contenido o usar los que Drupal nos da por defecto. En nuestro caso, básicamente nos manejamos con los contenidos de tipo “página”, aunque también manejamos los contenidos de tipo “banner”.

La página de gestión de contenidos es:

ACTUALIZAR OPCIONES							
Publicar el contenido seleccionado							Actualizar
<input type="checkbox"/>	TÍTULO	TIPO	AUTOR	ESTADO	ACTUALIZADO	IDIOMA	OPERACIONES
<input type="checkbox"/>	Podología Déborah Garrido <b>nuevo</b>	Página básica	admin	publicado	06/11/2015 - 00:38	Español	editar eliminar
<input type="checkbox"/>	Servicios	Página básica	admin	publicado	03/09/2015 - 18:09	Español	editar eliminar
<input type="checkbox"/>	Atención a domicilio	Página básica	admin	publicado	08/12/2014 - 18:07	Español	editar eliminar
<input type="checkbox"/>	Equipo	Imagen	admin	publicado	08/12/2014 - 17:50	Español	editar eliminar
<input type="checkbox"/>	Consulta general	Imagen	admin	publicado	08/12/2014 - 17:50	Español	editar eliminar
<input type="checkbox"/>	Escritorio	Imagen	admin	publicado	08/12/2014 - 17:49	Español	editar eliminar
<input type="checkbox"/>	Utensilios	Imagen	admin	publicado	08/12/2014 - 17:49	Español	editar eliminar
<input type="checkbox"/>	Sillón	Imagen	admin	publicado	08/12/2014 - 17:49	Español	editar eliminar
<input type="checkbox"/>	Ofertas	Página básica	admin	publicado	08/12/2014 - 17:27	Español	editar eliminar
<input type="checkbox"/>	Formulario de contacto	Webform	admin	publicado	05/23/2014 - 20:07	Español	editar eliminar
<input type="checkbox"/>	Buenas prácticas	Página básica	admin	publicado	05/22/2014 - 18:04	Español	editar eliminar
<input type="checkbox"/>	Equipamiento	Página básica	admin	publicado	05/22/2014 - 18:01	Español	editar eliminar
<input type="checkbox"/>	¿Quiénes somos?	Página básica	admin	publicado	05/22/2014 - 12:01	Español	editar eliminar

Aquí podemos tanto crear, como eliminar o editar contenidos. Si entramos al detalle de uno de ellos, encontramos su información tal que así:

**Editar Página básica Podología Déborah Garrido**
VISTA EDITAR DEVEL

Inicio » Podología Déborah Garrido

**Title \***  
Podología Déborah Garrido

**Resumen (Ocultar resumen)**

Podología Deborah Garrido proporciona el *mejor servicio a sus clientes* con el objetivo de alcanzar su *mayor grado de satisfacción*. Disponemos de los mas *avanzados equipamientos y tecnología* para detectar y tratar cualquier anomalia que esté relacionada con el pie. Poseemos los *certificados de calidad* necesarios para *garantizar el mejor servicio*.

Entre los *servicios principales que prestamos* destacan:

DUREZAS / CALLOSIDADES	VERRUGAS / PAPILOMAS	ORTESIS PLANTARES
HONGOS / MICOSIS	UÑA ENCARNADA	ORTESIS DE SILICONA
PIE DIABETICO / RIESGO	PODOLOGIA DEPORTIVA	PODOLOGIA GERIATRICA
PODOLOGIA INFANTIL	ATENCIÓN A DOMICILIO	

Puede obtener un mayor detalle de nuestro equipamiento, servicios y calidad dirigiéndose a las secciones correspondientes de la Web.

Déjelo en blanco para usar como resumen un fragmento recortado del texto completo.

Bastará con asignar un nombre al contenido, y luego introducir su contenido mediante el editor WYSIWYG.

### 3.6.5 Gestión de vistas

Las vistas son un elemento básico de Drupal y sirven para mostrar contenido de forma dinámica aplicando gran cantidad de filtros que nosotros podemos configurar (tipo de contenido, fecha publicación, etc...). Por ejemplo, en nuestro proyecto tenemos la vista de galería de imágenes que muestra un listado de contenidos de tipo imagen.

<b>TÍTULO</b> Título: <a href="#">Galería de imágenes</a>	<b>BLOCK SETTINGS</b> Block name: <a href="#">Ninguno</a> Access: <a href="#">Permiso</a>   <a href="#">Ver contenido publicado</a>	<b>Advanced</b> <b>CONTEXTUAL FILTERS</b> <a href="#">Agregar</a> <b>RELATIONSHIPS</b> <a href="#">Agregar</a> <b>NO RESULTS BEHAVIOR</b> <a href="#">Agregar</a> <b>EXPOSED FORM</b> Exposed form style: <a href="#">Básico</a>   <a href="#">Opciones</a>
<b>FORMATO</b> Formato: <a href="#">Grid</a>   <a href="#">Opciones</a> Mostrar: <a href="#">Fields</a>   <a href="#">Opciones</a>	<b>ENCABEZADO</b> <a href="#">Agregar</a> <b>PIE DE PÁGINA</b> <a href="#">Agregar</a> <b>PAGER</b> Use pager: <a href="#">Display a specified number of items</a>   <a href="#">4 elementos</a> More link: <a href="#">No</a>	<b>OTHER</b> Machine Name: <a href="#">block</a> Comentario: <a href="#">No comment</a> Use AJAX: <a href="#">No</a> Hide attachments in summary: <a href="#">No</a> Hide contextual links: <a href="#">No</a> Use aggregation: <a href="#">No</a> Query settings: <a href="#">Opciones</a> Field Language: <a href="#">Current user's language</a> Caché: <a href="#">Ninguno</a> Link display: <a href="#">Ninguno</a> CSS class: <a href="#">Ninguno</a> Theme: <a href="#">Information</a> Block caching: <a href="#">Do not cache</a>
<b>FIELDS</b> <a href="#">Agregar</a> Contenido: <a href="#">Título</a> Contenido: <a href="#">Image</a>		
<b>FILTER CRITERIA</b> <a href="#">Agregar</a> Contenido: <a href="#">Publicado (\$)</a> Contenido: <a href="#">Tipo (= Imagen)</a>		
<b>SORT CRITERIA</b> <a href="#">Agregar</a> Global: <a href="#">Random (asc)</a>		

A partir de esta vista se crea un bloque que será el que se posicione en la región del pie.

Mención especial también para el módulo Lightbox que hemos tenido que utilizar para la visualización de las imágenes a modo de popup y para poder iterar por ellas de forma intuitiva.

Cuando instalamos el módulo de vistas (views), automáticamente aparece un nuevo formato de visualización en la vista de imágenes que proporciona el efecto requerido. Se configura al editar el campo Contenido: Imagen de la vista:

**For** [All displays](#)

Appears in: [node:article](#), [node:banner](#), [node:imagen](#).

**Create a label**  
 Enable to create a label for this field.

**Exclude from display**  
 Enable to load this field as hidden. Often used to group fields, or to use as token in another field.

**Formatter**  
[Lightbox2: lightbox: galeria->large](#)

### 3.6.6 Gestión Bloques

Los bloques en Drupal vienen a ser como los principales elementos de información. Son como regiones de código que muestran información. La pantalla de gestión de bloques es:



Si queremos añadir un bloque sólo hay que pulsar en “Añadir bloque”. Destacamos también otra forma de crear bloques (en este caso dinámicos) y esto es directamente desde las vistas.

Cuando estamos en una vista (por ejemplo galería de imágenes), al un display (modo de visualización), podemos seleccionar que sea de tipo bloque, esto quiere decir que el listado de contenidos que ahí seleccionemos para mostrar se podrá visualizar dentro de un bloque que tendrá el mismo nombre del display. Esto es muy práctico cuando queremos mostrar información mediante PHP (módulo Views PHP). Por ejemplo, para mostrar información de banners de forma dinámica podríamos hacer lo siguiente:

#### Output code

```
<?php $nid = $data->nid; ?>
<?php $nodo = node_load($nid); ?>
<?php $alias = drupal_get_path_alias('node/'.$nid); ?>
<h2><?php print $data->node_title;?></h2>
<div class="block-image">
<?php if ($nodo->field_image){ ?>
  <img alt="<?php print $data->node_title;?>" src="/sites/default/files/banner/<?php print $data->field_field_image[0]
[rendered]['#item']['filename'];?>" typeof="foaf:Image" class="imageblock-image">
<?php }else{ ?>
  <img alt="<?php print $data->node_title;?>" src="/sites/default/files/clinica_2.jpg" typeof="foaf:Image" class="imageblock-
image">
<?php } ?>
</div>
<div class="block-body">
  <p><?php print $data->field_body[0][rendered]['#markup'];?></p>
</div>
```

Apply (this display)

Cancelar

Eliminar

Importante, a la hora de salvar un display de bloque, recordar seleccionar “Apply (this display)” porque si aparece “Apply (all displays)” sobrescribimos el comportamiento de todos los displays que tengamos definidos en la vista.

Recordar que la asignación de bloques a pantallas se hace mediante la gestión de contexto como hemos visto en un apartado anterior.

### 3.6.7 Gestión de Usuarios

Drupal define una jerarquía de usuarios en función de cuáles son los roles que van a tener cada uno de ellos en la plataforma. Por defecto crea usuarios de tipo administrador pero podríamos crear mas tipos de usuarios si quisiéramos y definir permisos para cada uno de ellos.

Esto es muy práctico cuando tenemos distintos tipos de usuarios donde unos quieren ver unas cosas pero no otras o viceversa. Con la creación de distintos tipos de roles de usuario estaría solucionado asignando los permisos pertinentes.

En el proyecto actual, sólo tenemos un único usuario administrador que se encarga de gestionar toda la información.

## 3.7 Implantación y Despliegue

Una vez que la aplicación ha sido lo suficiente testeada como para poder asegurar que su funcionamiento es correcto, podemos decir que la aplicación está lista para implantarse en su entorno de producción. El cliente de nuestro producto estaba satisfecho con las implementaciones desarrolladas hasta ahora y dio el visto bueno para que pudiéramos proceder a poner la aplicación en un entorno real.

El despliegue de una aplicación Drupal es relativamente sencillo. Por un lado se hace una copia del código y por otro se hace un backup de la base de datos. Posteriormente se configura el entorno de destino para que sea capaz de leer esta información correctamente.

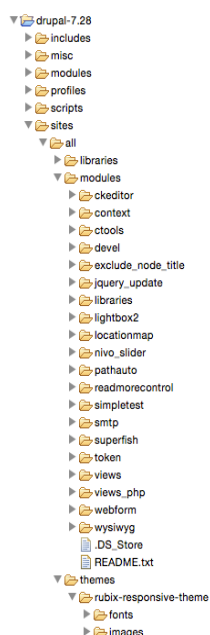
### 3.7.1 Instalación paquete XAMPP

Para asegurarnos de que el código y la base de datos pueden correr en el nuevo entorno, necesitamos un servidor Apache, un entorno PHP y un sistema gestor de base de datos MySQL. Estas soluciones nos las proporciona el paquete XAMPP directamente, aunque perfectamente se podrían instalar de forma separada.

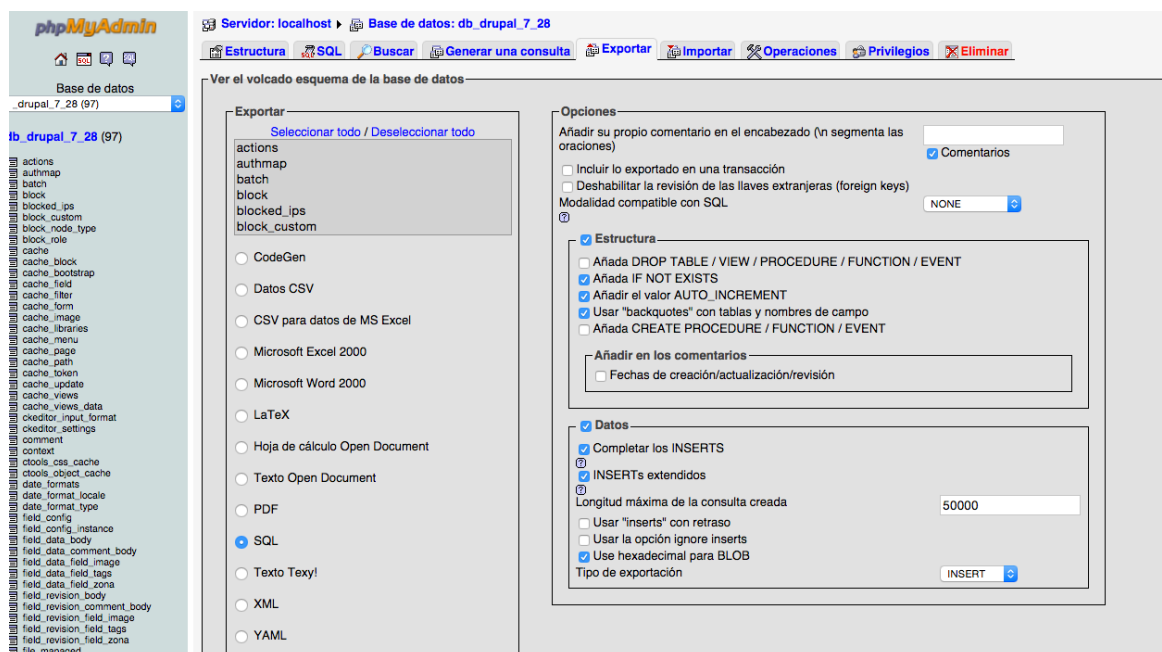
La instalación de XAMPP es muy sencilla y únicamente hay que descargar el paquete adecuado a nuestro sistema operativo desde su página oficial.

### 3.7.2 Copia de código y exportación de base de datos

Para el código, basta con hacer una copia de nuestra carpeta raíz de Drupal, mientras que la exportación de base de datos la podemos hacer incluso directamente desde la herramienta que proporciona XAMPP para la gestión de los datos, phpMyAdmin, que incluye una posibilidad de exportación bastante interesante donde podemos indicar si queremos exportar tablas, la base de datos completa, el formato del fichero de exportación (sql, csv, xml, yml, etc...).



La pantalla de gestión de phpMyAdmin tiene la siguiente apariencia:



### 3.7.3 Creación de virtual Host

Una vez instalado el paquete XAMPP, creamos el host virtual yendo al fichero:

“/XAMPP/xamppfiles/etc/extra/http-vhosts.conf” y nos aseguramos de tener lo siguiente:

```
...
...
# Use name-based virtual hosting.
NameVirtualHost *:80
...
...
<VirtualHost *:80>
    DocumentRoot "/Users/Antonio/WorkspacePhp/drupal-7.28"
    ServerName portal.podologia.localdomain
    <Directory "/Users/Antonio/WorkspacePhp/drupal-7.28">
        Options Indexes FollowSymLinks Includes ExecCGI
        AllowOverride All
        Order allow,deny
        Allow from all
    </Directory>
</VirtualHost>
```

En el caso del servidor de producción, sustituiremos la ubicación del código del proyecto “/Users/Antonio/WorkspacePhp/drupal-7.28” por la ubicación real del mismo. Para el servername igual, podemos cambiarlo para acceder a él desde la url que creamos conveniente.

### 3.7.4 Configuración de la base de datos

El fichero de exportación de base de datos tenemos que importarlo a nuestro nuevo entorno. Para ello podemos utilizar la misma herramienta phpMyAdmin que se utilizó para exportarlo.

Una vez hecho, hay que asegurarse de que el nombre de la base de datos esté bien configurado en Drupal. Para ello vamos al fichero general de configuración:

```
.../sites/default/settings.php
```

y añadimos nuestra configuración:

```
$databases = array (
  'default' =>
    array (
      'default' =>
        array (
          'database' => 'db_drupal_7_28',
          'username' => 'root',
          'password' => '',
          'host' => 'localhost',
          'port' => '',
          'driver' => 'mysql',
          'prefix' => '',
        ),
      ),
    ),
);
```

En este caso el nombre de la base de datos es “db\_drupal\_7\_28” mientras que el usuario es “root” y no tenemos contraseña.

### 3.7.5 Configuración de PHP

Aunque el proyecto no es muy grande en cuanto a consumo de recursos, conviene revisar la configuración del fichero “/XAMPP/xamppfiles/etc/php.ini” para asegurarnos de que las variables de limitaciones de recursos tienen valores aceptables. En nuestro caso una configuración correcta podría ser:

```
; Resource Limits ;
max_execution_time = 120 ; Maximum execution time of each script, in seconds
max_input_time = 240 ; Maximum amount of time each script may spend parsing request
data
memory_limit = 512M ; Maximum amount of memory a script may consume (8MB)
```

### 3.7.6 Configuración de Apache

Asegurarnos después de instalar Apache que el fichero de configuración “/XAMPP/xamppfiles/etc/httpd.conf” tiene descomentada la línea que carga el módulo rewrite. Para comprobar esto basta con buscar:

LoadModule rewrite\_module modules/mod\_rewrite.so

Si todos los pasos anteriores se han hecho de forma correcta, podremos insertar la url de nuestra aplicación en un navegador y acceder a ella.

<http://portal.podologia.localdomain>

Para autenticarnos, accedemos a la url: <http://portal.podologia.localdomain/user>

Las credenciales para entrar en Drupal son las que trae por defecto, con el usuario admin/admin

### **3.8 Mantenimiento**

#### *Introducción*

En ingeniería del software, el mantenimiento es la modificación de un producto después de la entrega, para corregir errores, mejorar el rendimiento, u otros atributos. Es una de las actividades más comunes de la ingeniería del software.

Una percepción común del mantenimiento es que siempre trata correcciones de errores, sin embargo, en muchos de los casos estos trabajos conllevan mejoras sobre el producto, es decir que no son acciones correctivas sino nuevas funcionalidades que se creen conveniente desarrollar o incluso actualizaciones de versiones de las tecnologías utilizadas con el objetivo de mejorar en rendimiento, seguridad, etc...

Mediante el mantenimiento, el cliente se asegura que su producto tiene cobertura de garantía ante cualquier imprevisto que pueda pasar en el software y que sea derivado de una mala implementación del mismo. Es por tanto, un plus de seguridad para el cliente.

#### *Aplicación*

En el caso de nuestro proyecto, existe un compromiso de mantenimiento de la aplicación durante 6 meses después de la puesta en producción.

Durante este tiempo, cualquier fallo detectado puede ser reportado por el cliente para que se corrija a la mayor brevedad posible.

Por otro lado, en caso de que el cliente quisiera incorporar nuevas funcionalidades, siempre existe esta opción sin tener en cuenta el tiempo. Eso sí, cada nueva funcionalidad deberá ser analizada y comprobada su viabilidad para poder llevarla a cabo.

## 4. Conclusiones y Trabajos Futuros

Con la realización de este trabajo fin de grado queda clara la importancia que tiene el ciclo de vida de desarrollo software para asegurar la calidad de un producto o servicio software. El objetivo que tiene todo ingeniero es hacer software de calidad que proporcione soluciones que satisfagan las necesidades de los clientes.

Como vimos en capítulos anteriores, existen distintos tipos de modelos a la hora de implementar este ciclo de desarrollo y por tanto no existe una única manera de buscar soluciones. Dependiendo de las circunstancias de cada proyecto (clientes, tecnología, complejidad de requisitos, coste...) nos adaptaremos a uno u otro modelo.

Lo mas importante es que todos los procesos estén establecidos y seamos capaz de llevarlos a cabo con la mayor solvencia posible. Podemos concluir sin ninguna duda que cualquier software de calidad que un ingeniero quiera proporcionar debería estar sujeto a un modelado de ciclo de vida software reconocible.

En cuanto a los trabajos futuros para la aplicación Web, se podría meter nuevas funcionalidades, como por ejemplo: encuestas, creación de nuevos tipos de contenidos como por ejemplo noticias, inclusión de información procedente de redes sociales, lectura de RSS procedente de diferentes fuentes de información relacionada con la podología, reservas de citas desde la aplicación o por ejemplo integración con Google Analytics para obtención de estadísticas.

Las posibilidades de trabajos futuros son múltiples y todo depende como siempre del cliente y de las necesidades que tenga. Lo mas importante de todo es que todo lo que se demandó al inicio del ciclo de vida se pudo cumplir de acuerdo a las necesidades.

## 5. Referencias

- [1] Página oficial de Drupal <http://www.drupal.org>
- [2] Drupal hispano. Comunidad de usuarios de Drupal <http://drupal.org.es>
- [3] Todd Tomlison, John K. VanDyk. Desarrollo con Drupal 7, ed Anaya, 2011.
- [4] Cristophe Aubry. Drupal 7, Crear y administrar sus sitios web, ed eni, 2012.
- [5] Angela Byron, Addison Berry y Bruno De Bondt, Sitios web dinámicos y atractivos con Drupal 7, ed O'Reilly Anaya, 2012.
- [6] Todd Tomlison. Manual imprescindible Drupal 7, ed Anaya, 2011.
- [7] Gregoria Gil García. El gran libro de Drupal 7, ed Marcombo, 2011.
- [8] Php Language Specification. <http://php.net>
- [9] Roger S. Pressman. Ingeniería del software, ed McGrawHill, 2014.

[10] Whitten Jeffrey. Análisis y de Diseño de Sistemas de la Información, ed McGrawHill, 2008.

[11] Steve McConnell. Desarrollo y Gestión de Proyectos Informáticos, ed McGrawHill, 1996.

## **6. Apéndice**

### **6.1 Contenido del CD**

- Código de la aplicación Drupal
- Resumen en español e inglés
- Export de la base de datos de la aplicación
- Paquete XAMPP

### **6.2 Manual de instalación y configuración**

Incluido dentro de la sección de implantación y despliegue 3.7.

### **6.3 Manual de Usuario**

Incluido dentro de la sección de documentación 3.6.