

Clínica Laurentis

Proyecto sobre una clínica de belleza

30/05/2014

José Manuel Cenador Herrera

2º ASIR - 03

CLÍNICA LAURENTIS

Porque la belleza lo es todo



© Clínica Laurentis | Desarrollado por José Manuel Cenador | Diseño: HTML5

Contenido

1) Información inicial.....	3
Nombre del proyecto y resumen del mismo	3
Planteamiento del problema	3
Propósito del proyecto.....	3
Responsable del proyecto	3
2) Diseño del Proyecto.....	3
Herramientas a utilizar	3
Orden de actuación.....	4
Cronograma.....	4
Copias de Seguridad	5
3) BBDD. Trabajo con el Modelo Entidad/Relación.....	5
Planteamiento	5
4) Trabajo con la Base de Datos	6
Explicación de Tablas.....	6
Medidas de Seguridad.....	8
Partes destacadas del código	8
5) Desarrollo WEB.....	9
Seguridad de la WEB	10
Encriptación de datos.....	11
6) Sección Pública	11
a) Página Principal	11
b) Quienes somos	13
c) Servicios que se ofrecen.....	13
d) Contacto	15
e) “Footer” o pie de página	16
7) Sección Privada.....	16
A) Login o Acceso	16
B) Menú de Opciones del personal médico.....	17
C) Insertar datos de un nuevo paciente	18
D) Modificar los datos de un Paciente.....	20
E) Eliminar los datos de un paciente	20
F) Listado de pacientes.....	21

G) Logout o Salida	22
H) Alta de Profesionales Médicos	22
8) <i>Comprobaciones con código PHP y aspectos relevantes</i>	24
1- Función para comprobar la fecha de nacimiento	24
2- Función para insertar imágenes y poder visualizarlas	24
3- Código para control de errores al insertar datos	25
4- Código para insertar datos de un nuevo paciente en la BBDD	26
5- Código del SLIDER en la portada	27
9) <i>Servidor de Backup y copias de seguridad (Anacron y Cron)</i>	27
10) <i>Instalación y configuración del Servidor web</i>	28
11) <i>Diseño de la red (Modelo Jerárquico)</i>	31
Beneficios de una red jerárquica.....	32
Principios para el diseño de una red	32
12) <i>Arquitectura de la RED</i>	33
Parte destacada: Configuración de un Switch	34
Parte Destacada: Configuración de un router.....	35
13) <i>Pruebas</i>	37
14) <i>Conclusiones</i>	37
15) <i>Aspectos a mejorar</i>	38

1) Información inicial

Nombre del proyecto y resumen del mismo

El presente proyecto lleva por nombre “Clínica Laurentis” y consiste en crear la infraestructura de una clínica de belleza y mejoras estéticas. Las diversas áreas que el proyecto abarca son entre otras: creación de la BBDD, desarrollo y mantenimiento de la página web, gestión de pacientes por los facultativos médicos mediante una interfaz web y arquitectura de la red (diseño de la red a nivel físico y lógico).

Planteamiento del problema

Clínica Laurentis es una gran compañía con sede en multitud de países que ha decidido dar el salto a nuestro país, para ello se han montado 3 sucursales (2 en Madrid y 1 en Sevilla). Como informático al cargo del proyecto me corresponden, entre otras, las tareas de crear las BBDD acorde a las necesidades de la empresa, una página web con un diseño atractivo para el público español y que a la vez sirva para que los profesionales médicos puedan consultar y modificar datos de los pacientes. También hace falta diseñar e implementar la conectividad entre las tres sedes de la empresa en nuestro país tanto a nivel de cableado como de la conectividad entre departamentos.

Propósito del proyecto

Tal como se indica mas arriba el propósito básico es asegurar que tanto clientes como trabajadores puedan acceder mediante el portal web a todos los servicios que se ofrecen. Por un lado los visitantes obtendrán información detallada sobre la empresa, los servicios que se ofrecen, profesionales médicos con los que se cuenta e información para encontrar su centro más cercano o como ponerse en contacto. Por otro lado los médicos podrán acceder a una sección privada de la web en la que encontrarán información detallada de los pacientes y tendrán la posibilidad de alterar dichos datos.

Responsable del proyecto

El encargado del seguimiento del proyecto y de supervisar su evolución mediante las tutorías es D. Samuel Arranz

2) Diseño del Proyecto

Herramientas a utilizar

Durante la elaboración de este proyecto han sido muchas las herramientas informáticas necesarias para llevar a buen puerto el trabajo. A continuación explico algunas de las mas importantes:

- **WAMP** (Servidor web). Este programa es el encargado de alojar el sitio web de cara al exterior.
- **MySQL**. Como lenguaje para programar la BBDD he optado por MySQL debido a su sencillez y a los conocimientos ya adquiridos durante el módulo. Va integrado como parte del servidor WAMP.
- **HTML5** y **CSS3**. Estos han sido los dos lenguajes con los que está desarrollada la página web. Se complementan a la perfección y permiten multitud de alteraciones para personalizar la web. Dado que no todos los navegadores son 100% compatibles con HTML5 la web ha sido desarrollada y probada con el navegador Google Chrome, por tanto en caso de utilizar otros navegadores no queda garantizado que el aspecto final sea el mismo.
- **PHP**. A la hora de conectar el sitio web con la BBDD se hace necesario usar este lenguaje que permite de manera transparente para el usuario conectar de manera segura con los datos de la empresa
- **Sublime Text**. Con este editor de texto, sencillo y extremadamente potente, se ha escrito todo el código tanto de MySQL, PHP, HTML5, CSS3 y JavaScript.
- **Packet Tracer**. Este programa propiedad de Cisco ha servido para diseñar la topología de la red y probar su funcionamiento.
- Sistemas operativos **Linux** y **Windows**. Durante la elaboración del proyecto se han utilizado ambos dependiendo de la tarea. Para programación web y MySQL se ha trabajado más con **Ubuntu**, mientras que en la comprobación final de la web y con Packet Tracer se ha utilizado **Windows 7** y **XP**.

Orden de actuación

Si bien el proyecto abarca diversas áreas, todas ellas acaban por interrelacionarse, de manera que es necesario llevar un orden apropiado para implementar cada bloque. Para comenzar se ha trabajado con el Packet Tracer diseñando la red de comunicaciones tanto a nivel físico como lógico. A continuación se ha trabajado sobre la BBDD con el modelo E/R y posteriormente con las tablas realizadas con MySQL. El siguiente paso ha sido el desarrollo web mediante HTML5 y CSS3, creando y modificando el aspecto de la página. Por último se ha seguido trabajando sobre la web pero en esta ocasión mediante el PHP que permite conectar la página con la BBDD creada previamente.

Cronograma

1. Se evalúan las necesidades de la empresa y se propone un plan de actuación
2. Trabajo sobre la infraestructura de las redes en las 3 sucursales
2. Diseño del modelo Entidad/Relación
3. Creación de las tablas en la BBDD.
4. Desarrollo de la página web (HTML5 y CSS3)
5. Implementar la conexión con la BBDD mediante el PHP
6. Análisis del funcionamiento del portal web.
7. Pruebas de funcionamiento y detección de errores.
8. Implementación de seguridad tanto en la web como en la BBDD

9. Corrección de los errores y nuevas pruebas de funcionamiento.
10. Conclusión del proyecto y resumen de los resultados

Copias de Seguridad

Dada la importancia del proyecto y de la abundante documentación, herramientas de desarrollo y pruebas realizadas, se hace necesario disponer de un riguroso sistema para controlar la información y evitar la pérdida de esta en caso de algún accidente. Por este motivo todos los días que se trabaja en el proyecto se toman las siguientes precauciones:

- Backup de todos los datos del proyecto en unidad extraíble USB y protegidos por contraseña.
- Sincronización de todos los movimientos diarios con la “nube” de Microsoft mediante el software gratuito de OneDrive.
- Copia de los backups diarios del punto 1 en el equipo doméstico en una partición separada del SO.

3) BBDD. Trabajo con el Modelo Entidad/Relación

Planteamiento

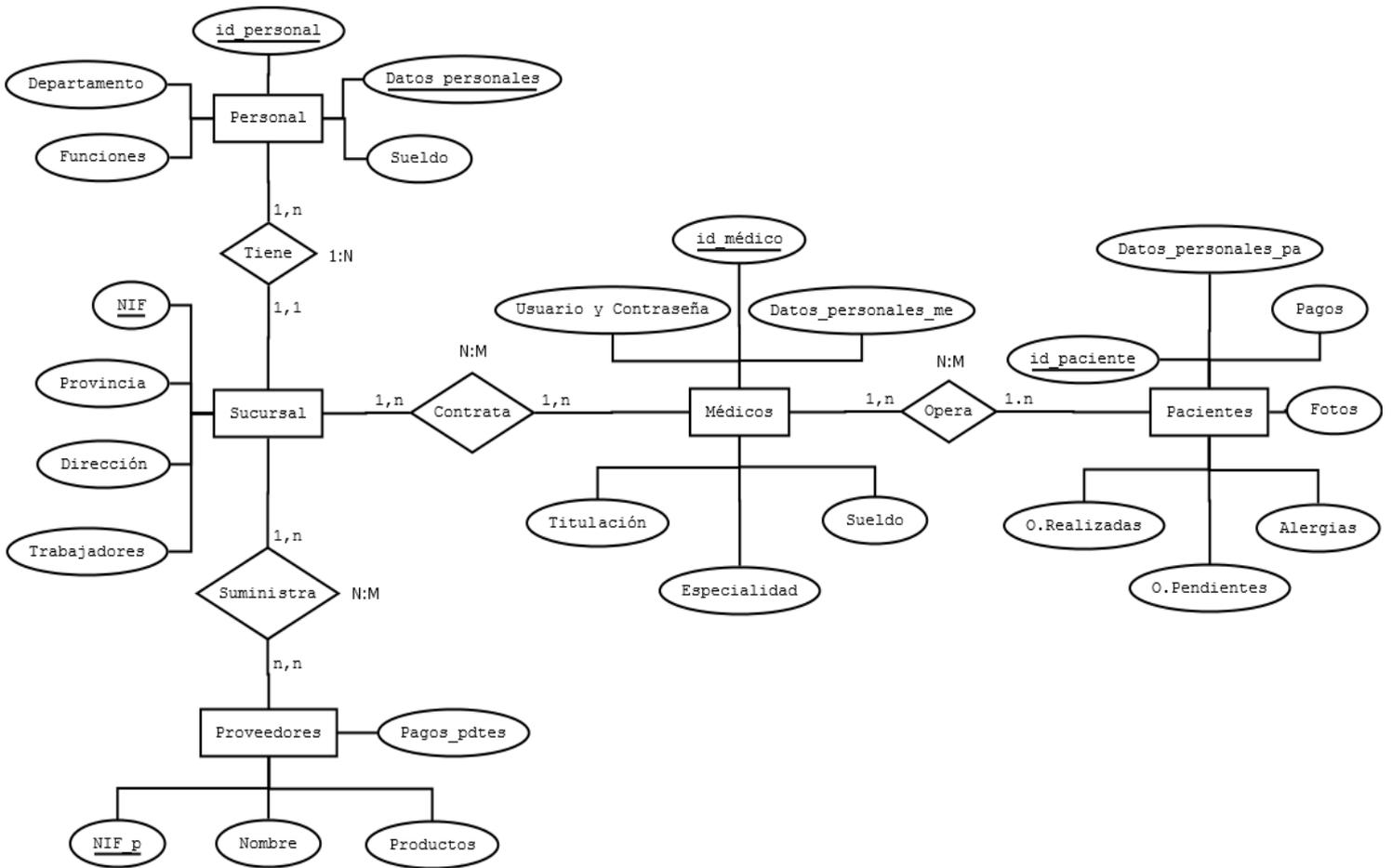
La empresa, dedicada a las operaciones cosméticas y tratamientos de belleza, dispone de 3 sucursales, dos de ellas en Madrid y otra en Sevilla que operan de forma autónoma. Cada sucursal tiene contratado personal no médico (gestores, administrativos, informáticos) así como los propios médicos. Estos últimos son los que tratan con el paciente y los operan.

Cada sucursal dispone de varios proveedores de suministros médicos y otros materiales. Cada proveedor puede ofrecer una cantidad variable de productos así como suministrar a más de una sucursal.

Tanto los médicos como el resto del personal están asignados a una sucursal, si bien los médicos pueden trabajar en las otras sucursales si se da el caso, el resto del personal está asignado fijo. Cada paciente puede ser operado por diversos médicos dependiendo de la mejora corporal que quiera adquirir, asimismo cada médico puede tratar a varios pacientes.

- **Sucursal** (NIF¹, provincia, dirección, trabajadores)
- **Personal** (id_personal², datos personales, departamento, funciones, sueldo, NIF¹)
- **Proveedores** (NIF_p³, nombre, productos, pagos_pdtes)
- **Médicos** (id_medico⁴, datos personales, titulación, especialidad, sueldo, usuario y contraseña)
- **Pacientes** (id_paciente⁵, datos personales, pagos, o_realizadas, o_pendientes, alergias, foto)
- **Opera** (datos personales me⁴, datos personales pa⁵)
- **Contrata** (NIF¹, datos personales me⁴)
- **Suministra** (NIF¹, NIF_p³)

```
mysql>
mysql> show tables;
+-----+
| Tables_in_Empresa |
+-----+
| contrata           |
| medicos           |
| opera             |
| pacientes         |
| personal          |
| proveedores       |
| sucursal          |
| suministra        |
+-----+
8 rows in set (0.00 sec)
mysql>
```



4) Trabajo con la Base de Datos

Explicación de Tablas

Tabla **MEDICOS**: Contiene los datos de los profesionales médicos que intervienen a los pacientes. Su "primary key" es "id medico" (código identificador único de cada médico), luego tenemos "datos_personales" que engloba los campos nombre, apellido1, apellido2 y fecha_nac. Por último están presentes las entradas "sueldo", "especialidad" y "titulación", "usuario" y "contraseña".

```

DROP TABLE IF EXISTS `medicos`;

CREATE TABLE `medicos` (
  `id_medico` int(11) NOT NULL AUTO_INCREMENT,
  `nombre` varchar(30) NOT NULL,
  `apellido1` varchar(30) NOT NULL,
  `apellido2` varchar(30) DEFAULT NULL,
  `titulacion` varchar(20) DEFAULT NULL,
  `especialidad` varchar(20) NOT NULL,
  `sueldo` int(11) DEFAULT NULL,
  `fecha_nac` date NOT NULL,
  `usuario` varchar(30) NOT NULL,
  `clave` varchar(60) NOT NULL,
  PRIMARY KEY (`id_medico`)
) ENGINE=InnoDB AUTO_INCREMENT=17 DEFAULT CHARSET=latin1;
    
```

```

DROP TABLE IF EXISTS `pacientes`;

CREATE TABLE `pacientes` (
  `id_paciente` int(11) NOT NULL AUTO_INCREMENT,
  `edad` int(3) NOT NULL,
  `nombre` varchar(30) NOT NULL,
  `apellido1` varchar(30) NOT NULL,
  `apellido2` varchar(30) DEFAULT NULL,
  `o_realizadas` varchar(50) DEFAULT NULL,
  `o_pendientes` varchar(30) DEFAULT NULL,
  `alergias` varchar(20) DEFAULT NULL,
  `foto` varchar(100) DEFAULT NULL,
  `pagos` float DEFAULT NULL,
  PRIMARY KEY (`id_paciente`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=latin1;
    
```

Tabla **PACIENTES**. Almacena toda la información de los pacientes de la empresa. El primary key es "id paciente" (código identificador único de cada paciente), luego está "datos_personales" que engloba los

campos *nombre*, *apellido1*, *apellido2* y *edad*. Por último están presentes las entradas “*o_realizadas*” (operaciones ya hechas), “*o_pendientes*” (futuras operaciones que el paciente desea), “*alergias*”, “*pagos*” (resumen de los pagos realizados hasta la fecha) y “*foto*” (lugar donde se almacenan las pruebas médicas como resonancias o radiografías).

Tabla **PERSONAL**. En esta tabla figuran los datos del resto de personas que trabajan en la empresa como pueden ser enfermeras, personal administrativo e informáticos. Los campos presentes son los siguientes: “*id_personal*” (código identificador único de cada empleado), “*datos_personales*” (conjunto de datos referentes al trabajador), “*departamento*”, “*funciones*”, “*sueldo*” y “*NIF*” (clave propagada desde la tabla SUCURSAL).

```
DROP TABLE IF EXISTS `personal`;

CREATE TABLE `personal` (
  `id_personal` int(11) NOT NULL AUTO_INCREMENT,
  `datos_personales` varchar(30) NOT NULL,
  `departamento` varchar(15) DEFAULT NULL,
  `funciones` varchar(20) DEFAULT NULL,
  `sueldo` float NOT NULL,
  `nif` varchar(9) references `sucursal`,
  PRIMARY KEY (`id_personal`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
DROP TABLE IF EXISTS `sucursal`;

CREATE TABLE `sucursal` (
  `nif` varchar(9) NOT NULL,
  `provincia` varchar(15) NOT NULL,
  `direccion` varchar(30) DEFAULT NULL,
  `trabajadores` int(11) DEFAULT NULL,
  PRIMARY KEY (`nif`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Tabla **SUCURSAL**. Esta tabla hace referencia a cada una de las 3 clínicas que la empresa tiene en territorio nacional (2 en Madrid y 1 en Sevilla), los campos de la tabla son: “*NIF*” (el primary key de la tabla), “*provincia*”, “*dirección*” y “*trabajadores*” (referido al nº de trabajadores que tiene cada sucursal).

Tabla **PROVEEDORES**. Aquí se registran los datos de los proveedores de la empresa (material médico, suministros, etc...), el primary key es “*NIF_P*” y el resto de campos son “*nombre*”, “*productos*” y “*pagos-pdtes*” (en caso de que no se abonasen en el momento oportuno quedarán aquí registrados).

```
DROP TABLE IF EXISTS `proveedores`;

CREATE TABLE `proveedores` (
  `nif_p` varchar(9) NOT NULL,
  `nombre` varchar(20) DEFAULT NULL,
  `productos` varchar(50) NOT NULL,
  `pagos_pdtes` int(11) DEFAULT NULL,
  PRIMARY KEY (`nif_p`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Tanto **OPERA**, **CONTRATA** Y **SUMINISTRA** son la “relación” entre las diversas entidades del modelo. Al haberse diseñado como un modelo sin propagación cada una de estas “relaciones” llevará el atributo principal o primary key de las entidades con las que está vinculada.

```
DROP TABLE IF EXISTS `opera`;

CREATE TABLE `opera` (
  `id_medico` int(11) references `medicos`,
  `id_paciente` int(11) references `pacientes`
);
```

```
DROP TABLE IF EXISTS `contrata`;

CREATE TABLE `contrata` (
  `id_medico` int(11) references `medicos`,
  `nif` varchar(9) references `sucursal`
);
```

```
DROP TABLE IF EXISTS `suministra`;

CREATE TABLE `suministra` (
  `nif` varchar(9) references `sucursal`,
  `nif_p` varchar(9) references `proveedores`
);
```

Medidas de Seguridad

Como medida de seguridad ante eventuales errores a la hora de modificar los datos de un paciente, o incluso de borrarlos por error, se han creado dos **triggers** que se encargarán de recoger los datos de los nuevos pacientes así como las modificaciones que los especialistas decidan hacer en cada caso. El primer paso consiste en crear una nueva tabla en la BBDD que llevará por nombre “**pacientes_copia**” y tendrá los mismos campos que la tabla “**pacientes**”.

```
create table pacientes_copia(id_paciente int(10) primary key, edad int(3) not null,
  nombre varchar(30) not null, apellido1 varchar(30) not null, apellido2 varchar(30),
  o_realizadas varchar(200), o_pendientes varchar(200), alergias varchar(50), pagos float);
```

Una vez creada dicha tabla se procede a lanzar los triggers antes mencionados, que recogerán las entradas y modificaciones hechas en la tabla “**pacientes**”. Llevan por nombre **pacientes_1** (trigger para los “**insert into**”) y **pacientes2** (“**trigger** para los “**update**”).

```
delimiter //
create trigger paciente1 before insert on pacientes
  for each row begin
  insert into pacientes_copia(id_paciente, edad, nombre, apellido1, apellido2, o_realizadas,
  o_pendientes, alergias) values(new.id_paciente, new.edad, new.nombre, new.apellido1,
  new.apellido2, new.o_realizadas, new.o_pendientes, new.alergias);
  end //
delimiter ;

delimiter //
create trigger paciente2 before update on pacientes
  for each row begin
  update pacientes_copia set edad = new.edad, nombre = new.nombre, apellido1 = new.apellido1,
  apellido2 = new.apellido2, o_realizadas = new.o_realizadas, o_pendientes = new.o_pendientes,
  alergias = new.alergias where id_paciente = new.id_paciente;
  end //
delimiter ;
```

Partes destacadas del código

- 1) **Conexión** con la BBDD para validar, mediante usuario y contraseña, a los médicos y administradores del sistema que desean acceder a los datos.

```
$conexion = mysql_connect("localhost", "root", "root") or die ("No se puede conectar con el servidor");
mysql_select_db("Empresa") or die ("No se pudo seleccionar la BBDD");
$instruccion1 = "select usuario, clave from medicos where usuario='$usuario' and clave='$clave'";
$consulta = mysql_query($instruccion1, $conexion) or die ("No se pudo realizar la instruccion");
```

- 2) Código para mostrar el **apellido** de cada profesional médico en su página inicial cada vez que inicia sesión.

```
$_SESSION['usuario_val'] = $usuario;

$saca_ape = "select apellido1 from medicos where usuario='$usuario'";
$consulta_ape = mysql_query($saca_ape, $conexion) or die ("No se pudo realizar la instruccion");
$id = mysql_fetch_array($consulta_ape);
$apel = $id['apellido1'];

$_SESSION['apellidos'] = $apel;
```

- 3) Comandos para **añadir** a un nuevo paciente a la tabla “pacientes” de la BBDD únicamente con los campos en los que los médicos pueden trabajar.

```
$conexion = mysql_connect("localhost","root","root") or die ("No se puede conectar con el servidor");
mysql_select_db("Empresa") or die ("No se pudo seleccionar la BBDD de pacientes");
$insertar = "insert into pacientes(edad,nombre,apellido1,apellido2,o_realizadas,o_pendientes,alergias,foto)
|values('$edad','$nombre','$apel1','$ape2','$realizado','$pendiente','$alergia','$nombreFichero')";
$consulta = mysql_query($insertar, $conexion) or die ("No se pudo insertar");
```

- 4) Código para la **actualización** de datos de un paciente cuando su facultativo así lo considere oportuno.

```
$conexion = mysql_connect("localhost","root","root") or die ("No se puede conectar con el servidor");
mysql_select_db("Empresa") or die ("No se pudo seleccionar la BBDD");
$modificar = "update pacientes set edad='$edad', nombre='$nombre', apellido1='$apel1', apellido2='$ape2',
o_realizadas='$realizado', o_pendientes='$pendiente', alergias='$alergia' where id_paciente='$id'";
$consulta = mysql_query($modificar, $conexion) or die ("No se pudo modificar");
```

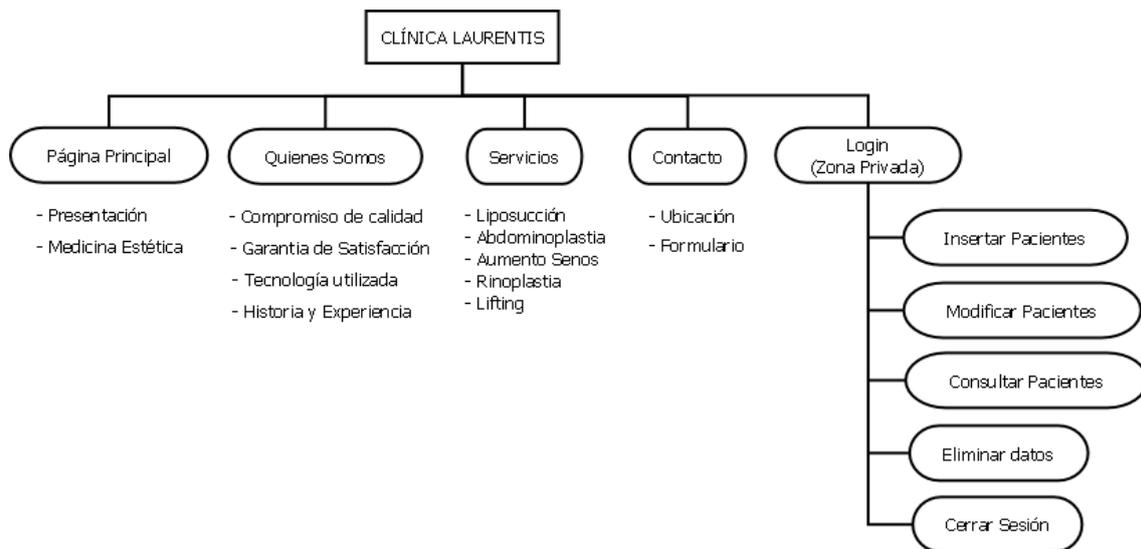
- 5) **Eliminar** a un paciente de la tabla “pacientes” es una opción que los profesionales médicos tienen. No obstante como se ha mencionado anteriormente se conservará una copia de dicha tabla (mediante triggers) para evitar “borrados accidentales”.

```
$conexion = mysql_connect("localhost","root","root") or die ("No se puede conectar con el servidor");
mysql_select_db("Empresa") or die ("No se pudo seleccionar la BBDD");
$borrar = "delete from pacientes where id_paciente='$id'";
$consulta = mysql_query($borrar, $conexion) or die ("No se pudo borrar");
```

5) Desarrollo WEB

La web ofrece un **diseño** serio, elegante y profesional, predominan los tonos azules y blancos y se busca la sencillez a la hora de navegar y leer la información. Por tanto el formato de la web va a ser prácticamente el mismo en todas sus secciones, se quiere transmitir una imagen corporativa que identifique las formas y colores con la propia empresa. Lo primero al hablar de esta sección es señalar que la página web se encuentra dividida en dos partes o bloques:

- A) **Parte PÚBLICA.** Esta sección está formada por el index y otras 4 páginas que nos hablan acerca de Clínica Laurentis, los servicios médicos que se ofrecen, formularios de contacto, la garantía para con el cliente así como el excepcional cuerpo médico que compone la empresa, etc...
- B) **Parte PRIVADA.** A través de una autenticación con usuario y contraseña tan solo podrán acceder a esta sección los administradores de sistema y los profesionales médicos.



Uno de los aspectos fundamentales dentro de la “zona privada” es el uso de sesiones, esto nos permite almacenar las variables de una página a otra fácilmente como puede ser “usuario y contraseña” o los valores para modificar en una tabla. Este sencillo código debe ser siempre lo primero que se introduzca al programar.

```

<?php
//Inicio de sesion
session_start();
?>
  
```

Seguridad de la WEB

Lo primero es implementa un filtro de seguridad para evitar que un usuario no identificado pueda acceder a través de la barra del navegador a una parte a la que no debería poder acceder. A tal efecto tenemos el siguiente código que comprueba que el usuario que intentan entrar ha sido previamente registrado en la BBDD.

```

if (isset($_REQUEST['usuario']) && isset ($_REQUEST['clave']))
  
```

Pero esto no es suficiente, hace falta asegurarse que efectivamente cada página de la zona “privada” es inaccesible para un usuario externo. Para ello tenemos este añadido al código:

```
if (isset($_SESSION['usuario_val']) && $_SESSION['usuario_val'] == TRUE)
```

De esta manera aseguramos la privacidad de los datos de nuestros clientes y asimismo evitamos pérdida de información si se eliminasen entradas de la BBDD.

Encriptación de datos

A la hora de enviar la información del acceso a la web por internet, esta podría ser vulnerable si viaja como texto plano sin cifrar. Para ello la contraseña debe ir cifrada, en este caso se ha optado por el sistema **SHA1**.

Mucho más seguro que sistemas tradicionales como el **MD5**, el sha1 garantiza una mayor protección a la hora de enviar contraseñas por internet. Su uso es tremendamente sencillo, por un lado a nivel de la página web y por otro en la propia BBDD para que lo reconozca, tan solo es necesario encriptar la clave cuando se introduce en la base de datos y posteriormente cada vez que un usuario intenta identificarse en la web. En la BBDD se vería de la siguiente manera la clave de acceso una vez encriptada con sha1:

```
$usuario = $_REQUEST['usuario'];
$clave = sha1($_REQUEST['clave']);
```

```
mysql> select id_medico,nombre,apellido1,usuario,clave from medicos where id_medico=25;
+-----+-----+-----+-----+-----+
| id_medico | nombre | apellido1 | usuario | clave |
+-----+-----+-----+-----+-----+
| 25 | Oscar | Sanchez | oscar.sanchez | f36ef2bb8487fe2b5741194ab8597e2245228d71 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

6) Sección Pública

a) Página Principal

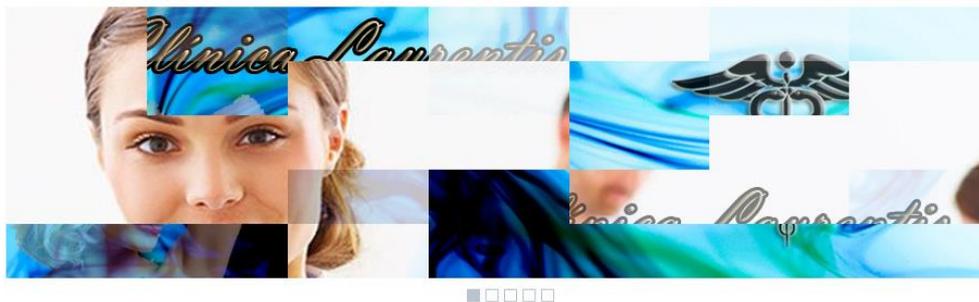
Como portal de inicio se ofrece el nombre y logo de la empresa junto con información básica sobre Clínica Laurentis y a lo que se dedica. Los colores corporativos y un aspecto limpio y elegante son la carta de presentación de la empresa a través de Internet.

Una vez que se accede a la web se puede ver la cabecera con el nombre de la empresa resaltando en color blanco sobre el fondo corporativo de color azul que usa la empresa.

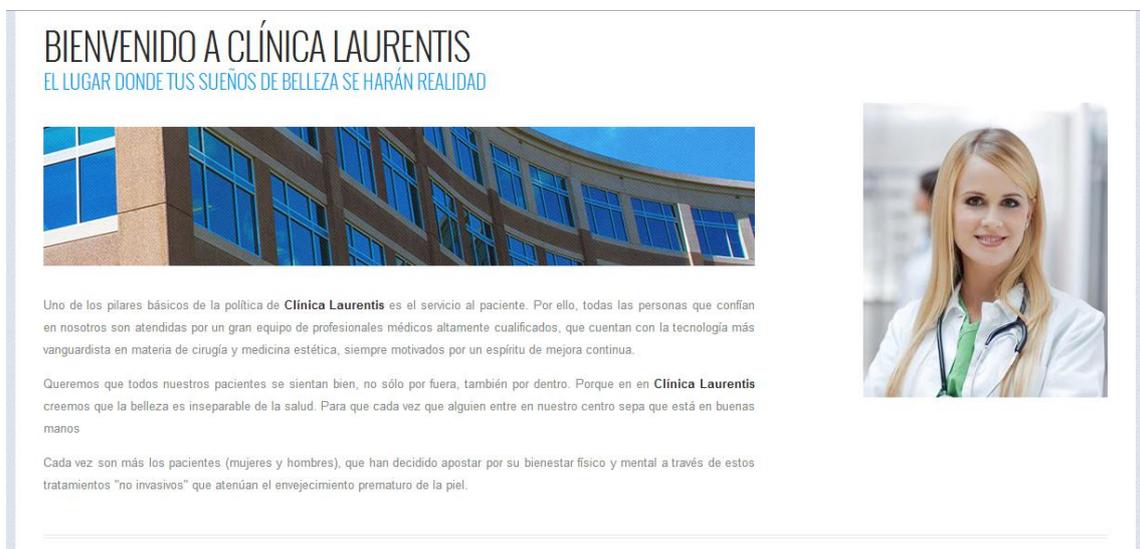
Lo siguiente que podrá verse es un SLIDER con imágenes asociadas a la empresa, los servicios y las instalaciones que irán mostrándose en sucesión dotando de dinamismo y un aspecto profesional a la web.



El slider en cuestión va a ir cambiando de imágenes usando una transición en forma de mosaicos de una imagen a otra con un efecto realmente interesante.



En la imagen superior puede apreciarse el momento en que se produce el cambio de una fotografía a otra y los mosaicos van cambiando aleatoriamente para mostrar la siguiente imagen.



b) Quienes somos

Una detallada explicación acerca de la profesionalidad del equipo médico, de las últimas tecnologías aplicadas al campo de la medicina estética y los orígenes de la empresa es la información que los visitantes pueden hallar en esta parte

CLÍNICA LAURENTIS
Porque la belleza lo es todo

PÁGINA PRINCIPAL QUIENES SOMOS SERVICIOS CONTACTO LOGIN

ESTÁS EN BUENAS MANOS
UN EQUIPO DE GRANDES PROFESIONALES A TU ENTERA DISPOSICIÓN

NUUESTRO COMPROMISO Y GARANTÍA

Los profesionales de **Clínica Laurentis** trabajan a diario con una premisa, ofrecer la máxima calidad y dar los mejores resultados a nuestros clientes.

Para ello, sólo trabajamos con profesionales cualificados y con los proveedores de material quirúrgico y médico líderes en su campo, lo que nos permite ofrecer las máximas garantías en todos nuestros servicios.

Aunque tratándose de Medicina, es posible que en algunos casos, haya diferencias o casos que no terminan como ambas partes deseáramos, por ello disponemos de un Servicio de Atención al Cliente cuyo compromiso es atender las llamadas que recibe en un plazo máximo de 48 horas y ofreciendo una resolución de 10 días. Cualquier duda que tenga, podrá ser resuelta por nuestros compañeros del Servicio de Atención Al Cliente.

TECNOLOGÍA DE VANGUARDIA

Clínica Laurentis dispone de quirófanos propios, perfectamente equipados, en las clínicas de las principales

Nuestro equipo médico especialista en cirugía estética, está a la vanguardia en las últimas técnicas quirúrgicas, para ofrecer a todas las personas el mejor resultado.

c) Servicios que se ofrecen

Un resumen de los distintos tratamientos médicos que los pacientes pueden encontrar en las instalaciones como pueden ser:

- **Liposucción.** Es una intervención de cirugía estética que permite remodelar la silueta en aquellas partes del cuerpo afectadas por una acumulación excesiva de grasa
- **Abdominoplastia** o reducción de abdomen es el procedimiento quirúrgico consistente en la corrección de la flaccidez muscular mediante la extirpación del exceso de piel y grasa abdominal
- **Aumento de senos** es una cirugía sencilla que permite realzar el contorno de la mujer. Es una de las operaciones de cirugía estética más común
- **Rinoplastia** o cirugía de nariz es una de las operaciones de cirugía estética más habituales y permite cambiar la forma de la nariz, creando más armonía entre la nariz y el resto de la cara.

- **Lifting** facial, también conocido como ritidectomía, es un procedimiento de cirugía estética facial que permite elevar la piel, tensarla y corregir el descolgamiento.

PÁGINA PRINCIPAL
QUIENES SOMOS
SERVICIOS
CONTACTO
LOGIN



TE OFRECEMOS LOS MEJORES TRATAMIENTOS

DISPONEMOS DE LO ÚLTIMO EN TECNOLOGÍA A TU DISPOSICIÓN

Nuestro equipo médico especialista en cirugía estética, está a la vanguardia en las últimas técnicas quirúrgicas, para ofrecer a todas las personas el mejor resultado.

Nuestros cirujanos valoran de forma gratuita a cada paciente en la primera consulta, informándole en profundidad sobre la solución más adecuada en su caso, así como el tipo de anestesia a utilizar. Nuestras técnicas son poco traumáticas: la mayoría de intervenciones se realizan con anestesia local y sedación y de forma ambulatoria, lo que no requiere el ingreso del paciente y permite reducir el tiempo de intervención y recuperación.

Además de la cirugía facial, contamos con los mejores tratamientos de medicina estética tales como Botox, Bioplastia, Terapia Celular y Medicina Antiejejecimiento, entre otros, para potenciar los resultados de una cirugía facial y conseguir un efecto aún más espectacular.

La preocupación por nuestro cuerpo ha adquirido especial importancia en los últimos años. Mejorar aquella parte de la que no nos sentimos a gusto, repercute no sólo en nuestra imagen, si no también, en la autoestima.

En **Clinica Laurentis**, llevamos más de 20 años ayudando a que todas las personas se sientan bien. Para eso es fundamental contar con un equipo médico de especialistas en cirugía estética que basen su trabajo en la profesionalidad y el trato personalizado destinado a dar a cada paciente la solución que necesita.



LIPOSUCCIÓN

EN QUE CONSISTE LA LIPOSUCCIÓN:

La **liposucción** o **lipoescultura** es una intervención de cirugía estética que permite remodelar la silueta en aquellas partes del cuerpo afectadas por una acumulación excesiva de grasa que es extraída mediante unas finas cánulas conectadas a una máquina de vacío. Para introducir las cánulas sólo se requieren unas incisiones mínimas, por lo que las posibles cicatrices serán prácticamente imperceptibles. Generalmente, la operación de liposucción se realiza con **anestesia local** y sedación, aunque puede variar según sean las características propias de cada paciente y del área a tratar. La liposucción permite obtener un resultado óptimo y casi permanente en la medida en que las células grasas no tienen la capacidad de multiplicarse por lo que los adipocitos no volverán a producirse.

ABDOMINOPLASTIA

EN QUE CONSISTE LA TÉCNICA DE LA ABDOMINOPLASTIA:

La **abdominoplastia** o reducción de abdomen es el procedimiento quirúrgico consistente en la corrección de la flaccidez muscular mediante la extirpación del exceso de piel y grasa abdominal. La abdominoplastia es el tratamiento utilizado para obtener un vientre más firme, plano y una cintura más estrecha.

La intervención quirúrgica en la abdominoplastia se realiza normalmente con **anestesia general** o **epidural**, dura aproximadamente de dos a tres horas, realizando una incisión por encima de la zona púbica que se prolonga hacia las caderas, para liberar el exceso de piel del abdomen y si el paciente lo requiere, se refuerzan con suturas los músculos abdominales. Se realiza en un hospital o clínica acreditada y requiere de ingreso hospitalario.

Existen diversas técnicas a aplicar, según cada caso, que el cirujano le expondrá de forma clara y comprensible. Habitualmente, la incisión para una abdominoplastia se realiza justo encima de la región púbica, por lo que la cicatriz quedará oculta bajo la ropa interior o bañador. La piel del vientre es liberada y tensada, y la pared muscular restaurada. Mientras el paciente esté ingresado, llevará un drenaje temporal para eliminar fácilmente el exceso de líquidos que pueda acumularse en la zona intervenida.





AUMENTO DE SENOS

EN QUE CONSISTE LA OPERACIÓN DE AUMENTO DE SENOS

El aumento de mamas o senos, que también se puede denominar **mamoplastia** de aumento, es una cirugía sencilla que permite realzar el contorno de la mujer. Es una de las operaciones de cirugía estética más común ya que permite mejorar el aspecto de pechos pequeños, equilibrar una diferencia en el tamaño de las mamas, como cirugía reconstructiva después de una intervención mamaria o para corregir una reducción de mamas después de un embarazo.

La **técnica de mamoplastia** consiste en la realización de una pequeña incisión por la que se introducirá la prótesis. Para insertar y posicionar las prótesis se pueden emplear diferentes incisiones: la periareolar, que se hace alrededor de la areola; la axilar, a nivel de la axila y en algunos casos a nivel del pliegue de debajo de la mama. En todos los casos, la incisión es muy corta por lo que la cicatriz posterior, en la inmensa mayoría de los casos resulta casi invisible.

A través de esta incisión, el cirujano crea una cavidad donde irá alojada la prótesis, directamente debajo del tejido mamario o debajo del músculo pectoral. La mamoplastia de aumento de pechos se realiza habitualmente con **anestesia general**, y en algunos casos con anestesia local con sedación, según criterio médico. La intervención dura

14

d) Contacto

En esta sección se explica donde se puede encontrar la sucursal más cercana a tu ciudad de Clínica Laurentis junto a un pequeño mapa (“Google Maps”) que lo indica.

FORMULARIO DE CONTACTO

CUNÉNTANOS LO QUE TU QUIERAS

Datos de contacto

Nombre:

Apellidos:

Teléfono:

E-mail:

Comentarios:

Escribe tu opinión...

Datos adicionales

País de residencia:

Ciudad de residencia:

¿Cuántas operaciones estéticas te has hecho?

- Ninguna
 De 1 a 3
 Mas de 3

Enviar

Asimismo se dispone de un formulario con el que ponerse en contacto con la empresa y que sirve a la vez de herramienta estadística con una serie de preguntas de carácter voluntario.

Preguntas como lugar de residencia o el número de operaciones estéticas realizadas ayudan al departamento de marketing a conocer mejor a los futuros clientes.

Los datos de contacto como nombre, apellidos, teléfono o e-mail pasan a formar parte (siempre que el cliente acepte darlos) de un fichero interno de la empresa con el que poder ofrecer servicios y publicidad a medida a los usuarios. La confidencialidad con la información facilitada es absoluta y nunca será revelada o usada por terceros.

e) “Footer” o pie de página

Como parte del diseño de la página se ha optado con un pie de página común en toda la parte pública de la web. En esta parte se ofrece información acerca de la política de cookies tal y como la nueva ley respecto a esta materia ha establecido recientemente.

También se ofrecen links de redirección a las zonas mas importantes del portal web para mayor comodidad de los usuarios.



Asimismo al final de todas las páginas de la empresa (tanto públicas como privadas) figura el **Copyright** de la página a nombre de **José Manuel Cenador** indicando también las tecnología principal en las que se basa la web: HTML5.

© Clínica Laurentis | Desarrollado por José Manuel Cenador | Diseño: HTML5

7) Sección Privada

A) Login o Acceso

Desde esta página se puede acceder al contenido privado de la compañía, tan solo los administradores y médicos cualificados podrán continuar navegando y acceder en cada caso una u otra sección.

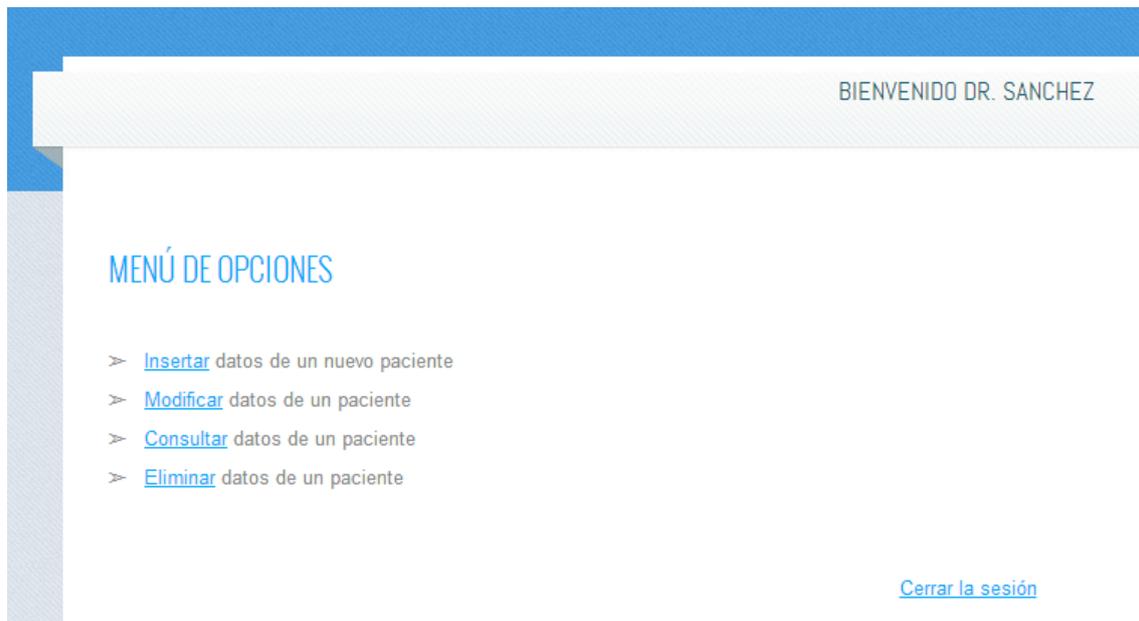
Mediante los campos “usuario” y “contraseña” se comprobará la identidad del usuario que intente acceder a la parte privada del portal, donde se albergan zonas sensibles de la empresa así como información de carácter privada de todos los clientes.

Como se indicó anteriormente, la información aquí introducida viaja cifrada por medio del algoritmo SHA1 lo que dificulta cualquier intento por suplantar la identidad de los usuarios registrados. Si el intento de identificación no resulta exitoso se procede a informar de ello y ofrecer la vuelta a la web.

B) Menú de Opciones del personal médico

Una vez autenticado correctamente, el médico accede a un sencillo e intuitivo menú en el cual puede consultar datos sobre pacientes, modificarlos, introducir otros nuevos, o si se diera la necesidad el borrarlos (como se mencionó anteriormente hay implementadas medidas de seguridad para evitar pérdidas accidentales de información).

Asimismo en la cabecera de la web figura un mensaje de bienvenida personalizado con el apellido del médico en cuestión (Bienvenido Doctor "X").



C) Insertar datos de un nuevo paciente

Mediante el uso de esta página el facultativo médico va a poder introducir los datos de un nuevo paciente en la base de datos de la empresa, que pasará a estar disponible para el resto de médicos en caso de que el cliente decida someterse a operaciones adicionales con otros especialistas de la empresa.

```
<center><p class="subtitle">Añadir un nuevo Médico</p></center>
<br><br>
<form action="mete_medicos.php" name="insertar" method="post">
  <table border="1px solid black">
    <tr><td>Nombre</td><td><input type="text" name="nombre" maxlength="30" size="30"></td></tr>
    <tr><td>Apellido1</td><td><input type="text" name="apellido1" maxlength="30" size="30"></td></tr>
    <tr><td>Apellido2</td><td><input type="text" name="apellido2" maxlength="30" size="30"></td></tr>
    <tr><td>Titulacion &nbsp;&nbsp;&nbsp;</td><td><input type="text" name="titulacion" size="50"></td></tr>
    <tr><td>Especialidad &nbsp;&nbsp;&nbsp;</td><td><input type="text" name="especialidad" size="50"></td></tr>
    <tr><td>Sueldo</td><td><input type="text" name="sueldo" size="30"></td></tr>
    <tr><td>Fecha Nacimiento &nbsp;&nbsp;&</td><td><input type="text" name="fecha_nac" maxlength="10" size="10"></td></tr>
    <tr><td>Usuario</td><td><input type="text" name="usuario" size="30"></td></tr>
    <tr><td>Clave</td><td><input type="password" name="clave" size="30"></td></tr>
  </table>
  <br><br>
  <INPUT TYPE="submit" NAME="insertar" VALUE="Insertar datos">
</form>
```

La imagen superior es un ejemplo del código HTML usado para introducir los campos del formulario.

BIENVENIDO DR. SANCHEZ

AÑADIR UN NUEVO USUARIO

Edad
 Nombre
 Apellido1
 Apellido2
 Operaciones Realizadas
 Operaciones pendientes
 Alergias
 Imágenes Ningún archivo seleccionado

[Regresar](#)

Nombre, apellidos, operaciones ya realizadas o pendientes de hacerse y un largo etcétera son las opciones que podrán introducirse. También se ofrece la opción de subir imágenes de pruebas médicas (como radiografías o resonancias) para su estudio.

LOS DATOS DEL PACIENTE SE HAN INGRESADO CORRECTAMENTE

[Regresar](#)

Mensaje informativo cuando el proceso concluye satisfactoriamente. En caso de no haberse introducido alguno de los campos mas relevantes (edad, nombre y apellido1) la web mostrará un **mensaje de error** y será necesario escribirlos. Esto es así deliberadamente dado que ciertos campos del paciente son imprescindibles.

No se ha podido realizar la modificacion debido a los siguientes errores:

El campo NOMBRE no puede quedar vacío
 El campo APELLIDO1 no puede quedar vacío
 El campo EDAD debe ser un valor numérico

[\[Volver \]](#)

El código y funcionamiento de este sistema para evitar errores se explicará más adelante en la “comprobación de errores de PHP”

D) Modificar los datos de un Paciente

Lo primero que se puede ver al acceder a esta sección es el listado de pacientes registrados en la base de datos, a continuación se ofrece la opción de modificar todos los campos deseados de un paciente concreto. Para ello lo primero es introducir el ID de dicho paciente y a continuación ya se pueden modificar los datos.

INTRODUZCA 1º EL ID DEL PACIENTE QUE DEBE MODIFICAR Y LUEGO LOS NUEVOS DATOS

ID del paciente:	<input type="text"/>
Nueva Edad	<input type="text"/>
Nuevo Nombre	<input type="text"/>
Nuevo Apellido1	<input type="text"/>
Nuevo Apellido2	<input type="text"/>
Nuevas Operaciones Realizadas	<input type="text"/>
Nuevas Operaciones pendientes	<input type="text"/>
Nuevas Alergias	<input type="text"/>

[Regresar](#)

En caso de que no se introdujese la ID de un paciente o se escribiera mal no se podría continuar y sería necesario escribirla si se quiere alterar algún paciente.

No se ha podido realizar la modificación debido a los siguientes errores:

El campo ID no puede quedar vacío

[\[Volver \]](#)

E) Eliminar los datos de un paciente

Si un paciente se da de baja en la compañía, fallece en la operación (¡Dios no lo quiera!) o el médico lo considera oportuno, se puede eliminar la entrada de dicho paciente de la base de datos de forma sencilla. Tras mostrarse el listado de pacientes por pantalla se solicita el ID del usuario, una vez introducido se confirma la selección y

el paciente queda borrado. Tras ello se puede regresar al menú principal de manera habitual.

INTRODUZCA EL ID DEL PACIENTE QUE DESEA BORRAR DE LA BBDD

ID:

[Regresar](#)

Tras ello se puede regresar al menú principal de la manera habitual.

LOS DATOS DEL PACIENTE SE HAN BORRADO CORRECTAMENTE DE LA BBDD

[Regresar](#)

Como se ha mencionado en puntos anteriores existe un mecanismo para evitar pérdidas de información accidentales en la base de datos. Mediante un trigger que se dispara cada vez que se inserta o modifica un paciente, se genera una copia de dicha información en otra tabla paralela que lleva por nombre "pacientes_copia" desde la que se puede recuperar la información.

F) Listado de pacientes

Desde esta sección de la web un médico puede consultar el listado de pacientes de la clínica y también visualizar las pruebas médicas (resonancias, radiografías, escáner,...) que un paciente pueda tener.

LISTADO DE PACIENTES

ID	Edad	Nombre	Apellido1	Apellido2	Op.Realizadas	Op.Pendientes	Alergias	Imágenes
18	43	Emilio	Butragueño	Perez	clavicula rota, menisco izquierdo	menisco derecho	Acaros	Imagen
4	31	Iker	Casillas	Fernandez	radio y cúbito rotos	lifting		Imagen
2	35	Belen	Esteban		rinoplastia, lifting, botox, aumento de senos	blanqueamiento dental, abdomino		
5	43	Casimiro	Garcia	Perez		rinoplastia		Imagen
1	46	Brad	Pitt		implante capilar, botox, lifting facial	abdominoplastia	Penicilina	
3	28	Cristiano	Ronaldo		menisco derecho	implante capilar	latex	

[Regresar](#)

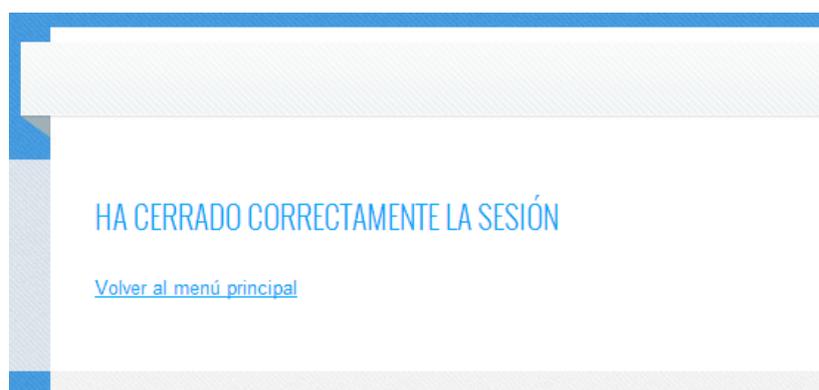


Haciendo clic en la columna “Imágenes” se abre una nueva ventana del navegador donde se pueden consultar dichas pruebas médicas.

Cuando un especialista sube una fotografía desde cualquier ubicación, se genera automáticamente una copia de dicha prueba médica que va almacenada en el servidor y a la que una función llama cada vez que se necesita consultarla. El código de esta función se verá más adelante.

G) Logout o Salida

Una vez que el usuario ha terminado su trabajo o desea salir de la interfaz web puede cerrar su sesión y regresar a la página web en su sección del “login”. Al salir, la sesión se destruye y cualquier información de las variables en uso desaparece evitando fugas de información o intrusiones no deseadas.



H) Alta de Profesionales Médicos

Al igual que los especialistas pueden dar de alta a los pacientes e introducir sus datos, un **administrador del sistema** está también capacitado para generar la ficha de los médicos y darlos de alta en el sistema. Rellenando los diversos campos que la interfaz solicita, un médico dado de alta en el sistema podrá inmediatamente acceder con sus

datos de usuario y contraseña a la base de datos de los pacientes. Los campos a rellenar incluyen desde datos personales hasta el título del médico o su especialidad.

El nombre, primer apellido, fecha de nacimiento, especialidad médica, fecha de nacimiento, usuario y contraseña son de carácter obligatorio. En caso de no insertarse ningún dato mostrará un mensaje de error.

No se ha podido realizar la modificación debido a los siguientes errores:

El campo NOMBRE no puede quedar vacío
El campo APELLIDO 1 no puede quedar vacío
El campo ESPECIALIDAD no puede quedar vacío
El campo USUARIO no puede quedar vacío. Por defecto debe ser nombre.apellido1
El campo CLAVE no puede quedar vacío
El Formato de Fecha introducido NO resulta Válido

[[Volver](#)]

Asimismo, en el caso de no introducir la fecha con el formato correcto (Año/Mes/Día) saldrá un mensaje avisando de dicho fallo, evitando así introducir errores de fecha en la base de datos (el código de esta sección se verá más adelante). Una vez se ha rellenado la ficha saldrá un mensaje avisando de que el proceso concluye correctamente.

LOS DATOS DEL MÉDICO SE HAN INGRESADO CORRECTAMENTE

8) Comprobaciones con código PHP y aspectos relevantes

1- Función para comprobar la fecha de nacimiento

Para evitar que se introduzca una fecha con el formato incorrecto, día/año/mes en vez de “año/mes/día” y que de error en la BBDD, se inserta un código que dará error si esto ocurre, avisando al usuario y evitando que estos valores se inserten en la tabla. También se evitará que se introduzcan letras u otros caracteres especiales en vez de la fecha separada por guiones. Por tanto la forma correcta es: “Año-Mes-Día”.

```
// Función para comprobar errores en la fecha de nacimiento:
if (!preg_match('/^\d{4}\-\d{1,2}\-\d{1,2}$/', $fecha)) {
    $errores = $errores . "    <LI>El Formato de Fecha introducido NO resulta Válido\n";
}
if (preg_match('/^\d{4}\-\d{1,2}\-\d{1,2}$/', $fecha)) {
    list($ano,$mes,$dia) = explode("-", $fecha);

    if ($ano > '2014') {
        $errores = $errores . "    <LI>El año introducido en la Fecha No es correcto\n";
    }
    if ($mes > '12') {
        $errores = $errores . "    <LI>El mes introducido en la Fecha No es correcto\n";
    }
    if ($dia > '31') {
        $errores = $errores . "    <LI>El día introducido en la Fecha No es correcto\n";
    }
}
}
```

2- Función para insertar imágenes y poder visualizarlas

Dado que los médicos necesitan incorporar imágenes de las pruebas médicas con cada paciente, se hace necesario incluir en el PHP una función que se encargue de seleccionar la imagen desde cualquier ubicación del disco duro, que compruebe su tamaño (evitando subir imágenes excesivamente pesadas), renombre el fichero si su formato no es el adecuado, y por último que copie dicha imagen a una carpeta en el servidor donde quedará guardada y vinculada a cada paciente.

Para poder mostrar las imágenes previamente subidas hay que añadir un par de líneas de código en la web que muestra los datos de los pacientes, en caso de haber una imagen debe indicarlo, de no ser así dejará en blanco dicho campo. El nombre de la imagen aparecerá como un hipervínculo, haciendo clic en él se abrirá una nueva ventana del navegador que muestra dicha prueba médica.

```
if ($resultado['foto'] != "")
    print ("<TD style='border-bottom: 1px solid grey; padding-top: 8px; padding-bottom: 8px;'><center>
        |<A TARGET='_blank' HREF='fotos/' . $resultado['foto'] . ". ">&#9755; Imagen &#9754;</a></center></TD>\n");
else
    print ("<TD style='border-bottom: 1px solid grey; padding-top: 8px; padding-bottom: 8px;'><center>&nbsp;&nbsp;&nbsp;</TD>\n");
```

Dentro de la web encargada de insertar los datos dentro de la tabla es donde tenemos la función descrita con anterioridad en este punto, su código es el que viene a continuación y se integra junto al resto de comprobación de errores.

```
// Subir fichero o imagen
$copiarFichero = false;

if (is_uploaded_file ($_FILES['foto']['tmp_name']))
{
    $nombreDirectorio = "fotos/";
    $nombreFichero = $_FILES['foto']['name'];
    $copiarFichero = true;
}
elseif (is_uploaded_file ($_FILES['foto']['tmp_name']))
{
    $nombreDirectorio = "fotos/";
    $nombreFichero = "craneo.jpg";
    $copiarFichero = true;
}
// Mover la foto a su ubicación final
if ($copiarFichero){
    move_uploaded_file ($_FILES['foto']['tmp_name'], $nombreDirectorio . $nombreFichero);
}

// Con esto insertamos/guardamos el nombre de la imagen en la tabla
$imagen = $_FILES['foto']['name'];
```

3- Código para control de errores al insertar datos

Dentro de la tabla “médicos” o “pacientes” hay varios campos que, dada su importancia, no pueden quedar vacíos o bien su formato debe ser algo específico (por ejemplo, la edad debe ser un valor numérico). Para controlar que no se produzcan errores en la inserción de datos se hace necesario comprobar estos campos tan relevantes y avisar al usuario de que debe adecuarse al formato correcto o introducir un valor en dicho campo (evitando que quede vacío).

Campos como el nombre, primer apellido, edad, fecha de nacimiento o la especialidad de los médicos se deben comprobar en cada inserción de datos, el código utilizado se encarga de comprobar que no hay errores, en caso de haberlos saltará una notificación para que se modifiquen. Si todo está correcto se procede con el envío de la información.

```
// Código para detectar errores en campos que no pueden quedar vacíos
$errores = "";
if (trim($nombre) == "")
    $errores = $errores . " <LI>El campo NOMBRE no puede quedar vacío\n";
if (trim($apel1) == "")
    $errores = $errores . " <LI>El campo APELLIDO1 no puede quedar vacío\n";
if (!is_numeric($edad))
    $errores = $errores . " <LI>El campo EDAD debe ser un valor numérico\n";
```

Con `trim($nombre_de_la_variable)` se comprueba que el campo no quede vacío. Por su lado `!is_numeric($nombre_de_la_variable)` evita que se inserten datos que no sean de carácter numérico.

Si se introdujesen los datos incorrectos, tras mostrar la advertencia por pantalla, con la lista de errores cometidos, se ofrece “volver” a la página anterior conservando el resto de campos que si estaban bien escritos. Para ello se usa un botón volver con una función de JavaScript.

```
print("<P>No se ha podido realizar la modificacion debido a los siguientes errores:</P>\n<br>");
print("<UL>\n");
print($errores);
print("</UL>\n<br>");
print("<P>[ <A HREF='javascript:history.back()'>Volver</A> ]</P>\n");
```

4- Código para insertar datos de un nuevo paciente en la BBDD

Si bien hay mucho código PHP para modificar, eliminar y consultar los datos, dado que comparten un patrón parecido se va a poner como ejemplo la sección para insertar un paciente nuevo por parte de un médico.

Lo primero es nombrar las variables de los campos que el usuario ha rellenado en el HTML cuando pulsa el botón de “enviar”. Tras esta parte del código vendrían las comprobaciones de error para los campos en blanco o mal escrito, luego se verificaría que en caso de subirse una imagen esta cumpla con los requisitos (estas dos comprobaciones ya se han mostrado en los puntos anteriores y no hay necesidad de duplicar la información). En caso de que todo este correcto se pasaría a enviar toda la información a la tabla “pacientes” de la BBDD como se puede apreciar en el código de más abajo.

```
if (isset($_REQUEST['insertar']))
{
    $edad = $_REQUEST['edad'];
    $nombre = $_REQUEST['nombre'];
    $ape1 = $_REQUEST['ape1'];
    $ape2 = $_REQUEST['ape2'];
    $realizado = $_REQUEST['realizado'];
    $pendiente = $_REQUEST['pendiente'];
    $alergia = $_REQUEST['alergia'];
```

```
$conexion = mysql_connect("localhost","root","root") or die ("No se puede conectar con el servidor");

mysql_select_db("Empresa") or die ("No se pudo seleccionar la BBDD de pacientes");

$insertar = "insert into pacientes(edad,nombre,apellido1,apellido2,o_realizadas,
o_pendientes,alergias,foto) values('$edad','$nombre','$ape1','$ape2',
|$realizado','$pendiente','$alergia','$nombreFichero)";

$consulta = mysql_query($insertar, $conexion) or die ("No se pudo insertar");
```

5- Código del SLIDER en la portada

Para conseguir un efecto de transición interesante y novedoso entre las imágenes, se ha optado por un slider que trabaje con JavaScript, CSS3 y HTML5. Tras añadir parte del código a las correspondientes carpetas de “js” y “css” se pone la llamada a las funciones y la personalización del código en nuestro “index”.

```
<script type="text/javascript" src="js/jquery-1.4.2.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<link rel="stylesheet" href="css/coin-slider-styles.css" type="text/css" />

<script type="text/javascript">
  $(document).ready(function() {
    $('#coin-slider').coinslider({ width: 900, height: 250 });
  });
</script>
```

Puede apreciarse como se han modificado los parámetros para ajustar el tamaño del slider a un valor apropiado. Pueden añadirse mas campos para cambiar la velocidad de cambio entre imágenes, forzar que siempre sea un efecto concreto, etc... Como se ha mencionado en varias ocasiones he optado por la simplicidad y elegancia en el diseño eliminando funciones accesorias y que tampoco aportan una mejora evidente.

Lo último que queda es añadir el código de HTML indicando las imágenes que se mostrarán en el slider. Existe la opción de redirigir a otras páginas cada vez que se pinche en una imagen, lo cual he decidido desactivarlo. También se puede mostrar una “leyenda” bajo cada imagen del slider, función también desactivada. Simplicidad y elegancia frente a diseños mas recargados, esa debe ser siempre la premisa del presente proyecto.

```
<div class="12u"><center>
<div id='coin-slider'>
  <a href="#" target="_blank">
    <img src='sliders/slider0.png' >
  </a>
  <a href="#" target="_blank">
    <img src='sliders/slider1.jpg' >
  </a>
  <a href="#" target="_blank">
    <img src='sliders/slider2.jpg' >
  </a>
  <a href="#" target="_blank">
    <img src='sliders/slider3.jpg' >
  </a>
  <a href="#">
    <img src='sliders/slider4.jpg' >
  </a>
</div></center>
```

9) Servidor de Backup y copias de seguridad (Anacron y Cron)

Pese a los controles de errores que el proyecto presenta, para salvaguardar la información de intrusiones externas o pérdidas de información por errores de los usuarios, se hace necesario disponer tanto de un sistema de copia de archivos así como de un servidor independiente para almacenar todos estos datos.

A tal efecto se ha programado que todas las noches se efectúe un backup de la información contenida en las tablas de la base de datos en un servidor dedicado para tal propósito.

A través de un script que se ejecuta todas las noches a las 02:00am se volcará toda la información del día comprimida a un directorio. Posteriormente dicho backup se mueve al servidor dedicado a las copias de seguridad.

```
#!/bin/sh

fecha1=`cat /var/spool/anacron/cron.daily`
fecha2=`date +%Y%m%d`

if [ $fecha1 != $fecha2 ]; then

    fecha=`date +%Y/%m/%d---%H:%M`
    tar -czvf /home/backups/backup$fecha.tar /var/lib/mysql/empresa.sql
    echo $fecha2 > /var/spool/anacron/cron.daily
    echo "Backup del dia $fecha2 realizado correctamente" >>
/home/backups/trazas.txt

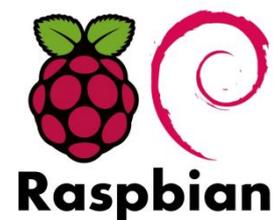
else
    echo "El comando ya se ha hecho anteriormente hoy $fecha2" >>
/home/backups/trazas.txt

fi
```

Utilizando las funciones de **CRON** y **ANACRON** se establece un pequeño programa que se encargará de automatizar la tarea de hacer un backup todos los días y creando además un archivos de trazas donde quedará registrado el proceso. El archivo diario con el backup llevará en su nombre puesta la fecha de cada día, así será tremendamente sencillo encontrar una copia de seguridad en caso de necesitarse.

10) Instalación y configuración del Servidor web

Para que la web del proyecto tenga salida al exterior hacen falta una serie de pasos a seguir. Procedemos a instalar un servidor web como el "Apache" que va a alojar el portal de Clínica Laurentis ofreciendo servicio hacia fuera de nuestra red local. La empresa tiene un servidor dedicado funcionando bajo Linux para esta tarea, aunque para la realización de este proyecto se va a utilizar la **Raspberry** con una distribución de **Debian** como método de simulación.



Lo primero que debe hacerse es ejecutar como root el comando:

sudo apt-get install apache2

A continuación hay que editar el archivo de configuración del programa, para ello basta escribir

sudo nano /etc/apache2/apache2.conf

Ahora es el momento de poner en marcha el sitio web en el servidor, lo primero es crear la carpeta contenedora donde ira almacenado el sitio web, para eso escribimos:

sudo mkdir /var/www/clínica

Dentro se mueve las web con todo su contenido, pues será en esa ruta donde buscará Apache para dar servicio.

Lo siguiente es ir a la ruta **/etc/apache2/sites-available** y se copia el fichero **"default"** renombrado a **"clínica"**. Ahora se procede a editar dicho archivo y modificar ciertos parámetros. Se va a cambiar la dirección de correo del webmaster, se indicará el nombre del servidor (servername) así como la ruta a la carpeta contenedora de la página web y todos sus recursos.

```

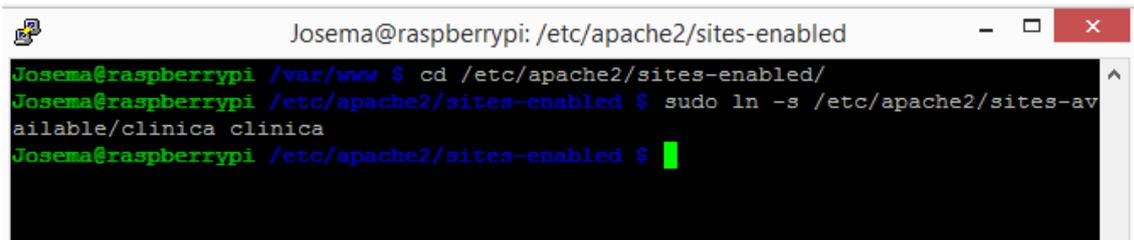
Josema@raspberrypi: /etc/apache2/sites-available
GNU nano 2.2.6 Fichero: clinica
<VirtualHost *:80>
  ServerAdmin administrador@clinica-laurentis.org
  ServerName www.clinica-laurentis.org
  DocumentRoot /var/www/clinica/
  <Directory />
    Options FollowSymLinks
    AllowOverride None
  </Directory>
  <Directory /var/www/clinica>
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
  </Directory>

  ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
  <Directory "/usr/lib/cgi-bin">
    AllowOverride None
    Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
  </Directory>

```

[^]G Ver ayuda [^]O Guardar [^]R Leer Fich [^]Y Pág Ant [^]K CortarTxt [^]C
[^]X Salir [^]J Justificar [^]W Buscar [^]V Pág Sig [^]U PegarTxt [^]T

Es necesario crear un enlace de este archivo de configuración, desde la carpeta sites-available hasta el directorio sites-enabled



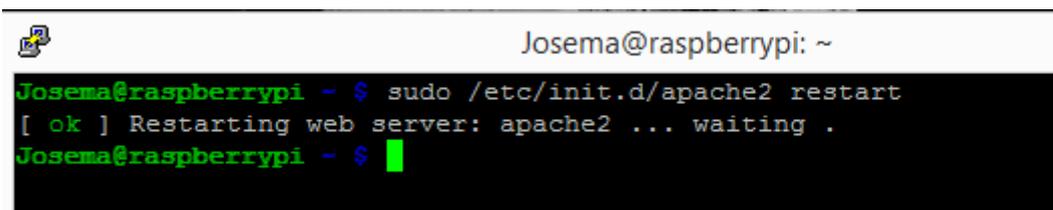
```

Josema@raspberrypi: /etc/apache2/sites-enabled
Josema@raspberrypi /var/www $ cd /etc/apache2/sites-enabled/
Josema@raspberrypi /etc/apache2/sites-enabled $ sudo ln -s /etc/apache2/sites-available/clinica clinica
Josema@raspberrypi /etc/apache2/sites-enabled $ █

```

En este punto es necesario reiniciar el servicio Apache para lo que tenemos la siguiente orden:

sudo /etc/init.d/apache2 restart



```

Josema@raspberrypi: ~
Josema@raspberrypi ~ $ sudo /etc/init.d/apache2 restart
[ ok ] Restarting web server: apache2 ... waiting .
Josema@raspberrypi ~ $ █

```

Tras finalizar este último paso el portal web de Clínica Laurentis está operativo y listo para ofrecer su servicio.

En la simulación que supone tener la página web en una Raspberry se hace necesario un paso mas, para poder dar **servicio al exterior** de la red local se necesita tener una ip fija a la que los clientes puedan acceder. A tal efecto existen multitud de servicios orientados a solucionar este problema, uno de los más utilizados es el de **http://www.noip.com** que será el que se utilice aquí.

Nos va a proporcionar una ip fija en internet para que siempre podamos acceder desde cualquier parte del mundo a la Raspberry y los contenidos que almacena. Tras darnos de alta en su web e instalar el programa en la raspberry se han de modificar una serie de parámetros en su archivo de configuración. Por último se crea un script que permita al programa arrancar con el inicio de sistema y por tanto estar siempre operativo:

```

case "$1" in
start)
echo "Iniciando noip2..."
/usr/local/bin/noip2
;;
stop)
echo -n "Apagando noip2..."
killproc -TERM /usr/local/bin/noip2
;;
*)
echo "Uso: $0 {start|stop}"
exit 1
esac
exit 0

```

Guardamos y procedemos a darle permisos de ejecución al script así como actualizar la configuración de arranque con este script

```
sudo chmod 755 /etc/init.d/noip2
sudo update-rc.d noip2 defaults
```

Podemos ahora reiniciar la Raspberry Pi para comprobar que efectivamente se ejecuta el script. El proceso de montar el proyecto web en un servidor concluye satisfactoriamente.

11) Diseño de la red (Modelo Jerárquico)

Tal y como se ha aprendido durante la asignatura de “Redes” y siguiendo las indicaciones del curso CCNA, se debe instaurar el diseño de la red siguiendo unos parámetros. El modelo de diseño jerárquico típico se separa en tres capas:

- **Capa de acceso:** Es la interfaz con los dispositivos finales. Esta capa de acceso puede incluir routers, switches, puentes, hubs y puntos de acceso inalámbricos. Los switches de la capa de acceso facilitan la conexión de los dispositivos de nodo final a la red. Por esta razón, necesitan admitir características como VLAN, Fast Ethernet/Gigabit Ethernet, PoE, QOS...
- **Capa de distribución:** La capa de distribución controla el flujo de tráfico de la red con el uso de políticas y traza los dominios de broadcast al realizar el enrutamiento de las funciones entre las VLANs definidas en la capa de acceso. Presentan disponibilidad y redundancia altas para asegurar la fiabilidad. Los switches de capa de distribución recopilan los datos de todos los switches de capa de acceso y los envían a los switches de capa núcleo. Los switches de capa de distribución proporcionan funciones de enrutamiento entre las VLAN (y por lo tanto funciones de la Capa 3), utilizan listas de control de acceso (ACL), agregado de los enlaces (tanto a la capa de acceso como a la capa de núcleo), deben respetar la QOS aplicadas en los switch es de acceso y deben tener una tasa de envío alta.
- **Capa núcleo:** Interconecta los dispositivos de la capa de distribución y puede conectarse a los recursos de Internet. El núcleo debe estar disponible y ser redundante. Los switches de capa núcleo requiere manejar tasas muy altas de reenvío, agregación de enlaces (de 10Gb con los switch de agregación), soporte de capa 3, suelen tener redundancia a nivel 3 (más rápida en converger que a nivel 2), enlaces Gigabit Ethernet/10 Gigabit Ethernet, componentes redundados (como la fuente de alimentación), suelen contar con opciones de refrigeración más sofisticadas (alcanzan

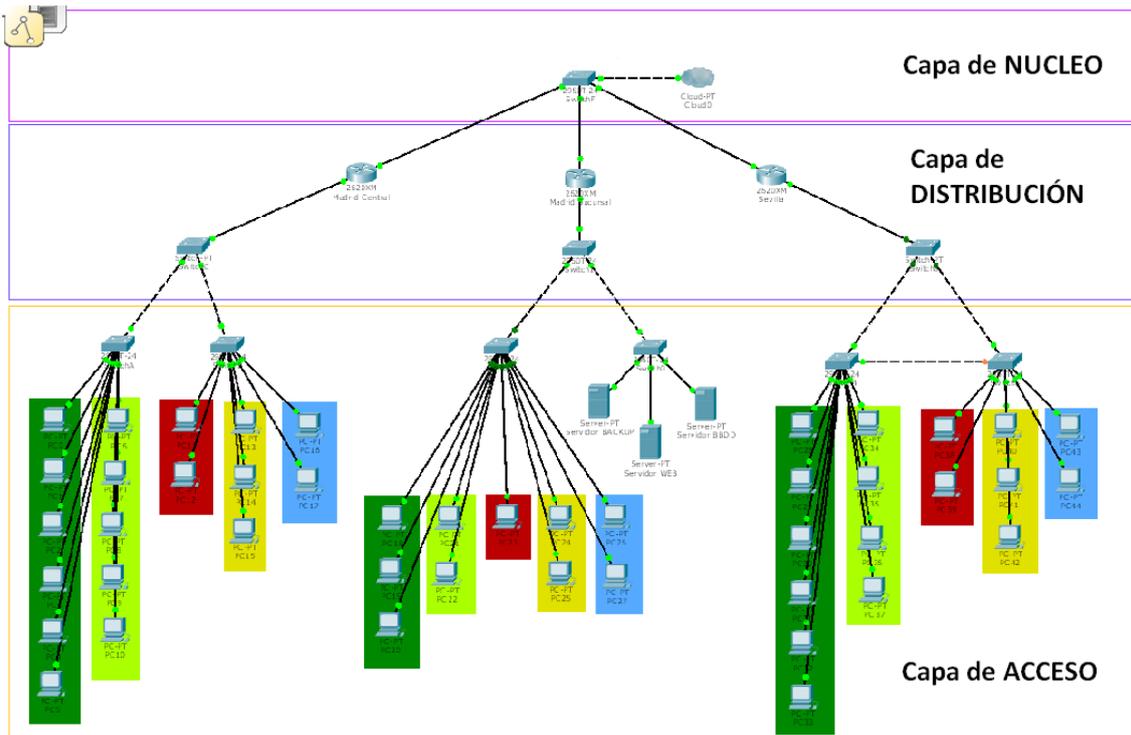
mayor temperatura por la carga de trabajo), con hardware que permite el cambio en caliente y QOS.

Beneficios de una red jerárquica

- **Escalabilidad:** Pueden expandirse con facilidad.
- **Redundancia:** Los switch se suelen conectar a varios switches de la capa superior.
- **Rendimiento:** El alto rendimiento de los switches de distribución y núcleo hacen que la información vaya a casi la velocidad del cable.
- **Seguridad:** Es posible configurar los switches de la capa de acceso con varias opciones de seguridad del puerto que proveen control sobre qué dispositivos se permite conectar a la red. Además, se cuenta con la flexibilidad de utilizar políticas de seguridad más avanzadas en la capa de distribución.
- **Facilidad de administración:** Los switches de una misma capa tienen una configuración similar.
- **Capacidad de mantenimiento.**

Principios para el diseño de una red

- **Diámetro de la red:** Es el número de dispositivos que un paquete debe atravesar antes de alcanzar su destino. Mantener bajo el diámetro de la red asegura una latencia baja y predecible.
- **Agregado de ancho de banda:** En una red jerárquica si se ven necesidades de ancho de banda se puede crear un EtherChannel (agregado de enlaces) y aumentarlo.
- **Redundancia:** Se puede proveer redundancia duplicando las conexiones de red entre los dispositivos o duplicando los propios dispositivos.
- **Se comienza por la capa de acceso:** Una vez analizado el número de dispositivos y los requerimientos de estos se elegirán cuáles y cuantos equipos de acceso se pondrán. En base al número de equipos de acceso se determina cuantos switches de capa de distribución son necesarios y en base a estos los de capa del núcleo.



Verde Oscuro: Equipos de los médicos

Verde Claro: Terminales para enfermeras

Rojo: Ordenadores para administradores y directores

Amarillo: Equipos para los contables

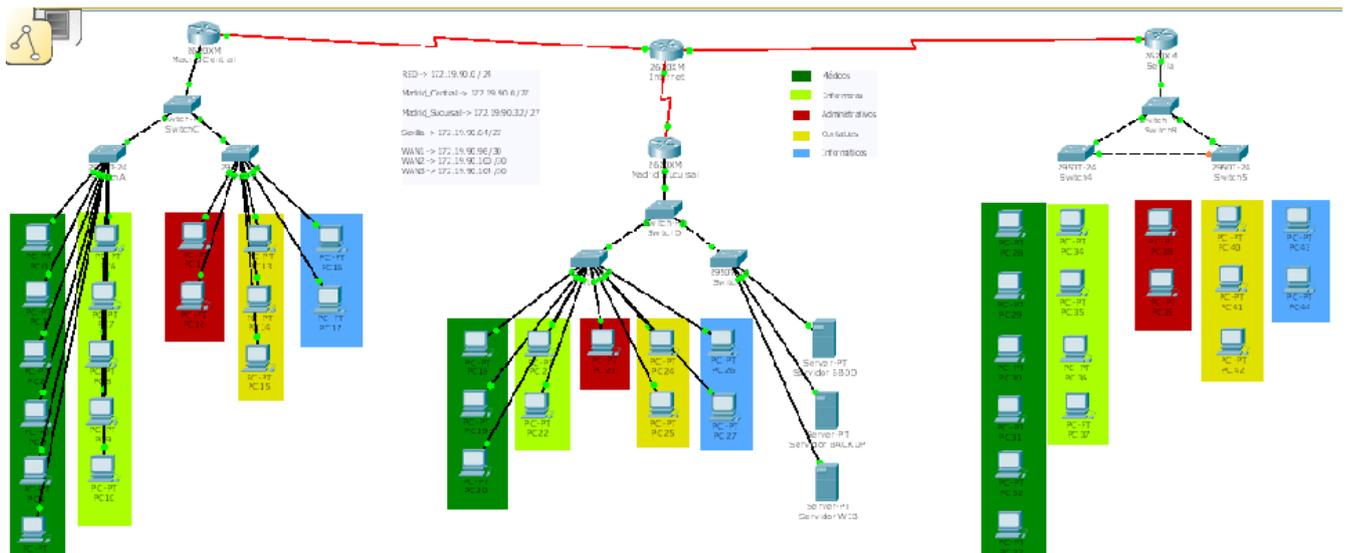
Azul: Ordenadores de los informáticos

12) Arquitectura de la RED

- **Oficinas:** Tenemos en total 3 oficinas en España, de las cuales 2 están en Madrid y 1 en Sevilla.
- **Routers:** 3, uno por cada oficina.
- **Switches:** 9 en total, 3 por cada oficina de entre los cuales 1 de ellos en cada caso pertenece a la capa de distribución.
- **Servidores:** Un servidor para la BBDD, otro para el portal WEB y un tercero que servirá como un backup de los anteriores. Los 3 servidores estarán ubicados en “Madrid Sucursal”.
- **Terminales:** 18 en “Madrid Central”, 10 en “Madrid Sucursal” y 17 en Sevilla. Estarán repartidos entre los diversos departamentos o grupos de la empresa
- **Grupos:** Médicos, enfermeras, administradores, gestores e informáticos. Cada uno de los grupos anteriores va a ir en su propia VLAN

- Conectividad:** Se va a utilizar cable “Fast Ethernet” de 100 mb en todo lo que es la capa de acceso a la hora de conectar los terminales con sus respectivos switch. El cableado entre switches y contra los router será “Gigabit Ethernet”, al igual que la conexión de los servidores.

Todos los equipos, servidores y routers deberán funcionar con una IP fija, de esta forma la gestión de las VLAN y de la seguridad resultará posible. La configuración de las tres sedes de Clínica Laurentis en España se muestra en el siguiente gráfico:



Datos de la RED y Rangos	vlan 10 name médicos
RED -> 172.19.90.0 / 24	vlan 20 name enfermeras
Médicos -> 172.19.90.0/27 (.1 a .30)	vlan 30 name administrativos
Enfermeras -> 172.19.90.32/27 (.33 a .62)	vlan 40 name contables
Administrativos -> 172.19.90.64/27 (.65 a .94)	vlan 50 name informaticos
Contables -> 172.19.90.96/27 (.97 a .126)	
Informáticos -> 172.19.90.128/27 (.129 a .158)	

Parte destacada: Configuración de un Switch

De entre las muchas configuraciones que lleva cada switch se procede a mostrar una parte del código de gran relevancia en la gestión de las VLAN y los enlaces. En concreto es uno de los switch en modo cliente de “Madrid Central”.

```
Switch(config)#vtp mode client
Setting device to VTP CLIENT mode.
```

```
Switch(config)#int gig1/1
Switch(config-if)#switchport mode trunk
Switch(config-if)#switchport trunk native vlan 99
Switch(config-if)#exit
```

```
Switch(config)#int gig1/2
Switch(config-if)#switchport mode trunk
Switch(config-if)#switchport trunk native vlan 99
Switch(config-if)#exit
```

```
Switch(config)#int range fa0/1-fa0/6
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 10
Switch(config-if-range)#exit
```

```
Switch(config)#int range fa0/7-fa0/12
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 20
Switch(config-if-range)#exit
```

```
Switch(config)#int range fa0/13-fa0/16
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 30
Switch(config-if-range)#exit
```

```
Switch(config)#int range fa0/17-fa0/20
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 40
Switch(config-if-range)#exit
```

```
Switch(config)#int range fa0/21-fa0/24
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 50
Switch(config-if-range)#exit
```

Parte Destacada: Configuración de un router

A modo de ejemplo de los varios routers de la empresa se procede a mostrar capturas con la configuración del router principal de “Madrid Central” con algunos de sus aspectos mas relevantes:

Comenzamos con la configuración de las interfaces y sub-interfaces asociadas a cada una de las VLAN descritas en puntos anteriores.

```

IOS Command Line Interface
!
interface FastEthernet0/0.10
 encapsulation dot1Q 10
 ip address 172.19.90.1 255.255.255.224
!
interface FastEthernet0/0.20
 encapsulation dot1Q 20
 ip address 172.19.90.33 255.255.255.224
!
interface FastEthernet0/0.30
 encapsulation dot1Q 30
 ip address 172.19.90.65 255.255.255.224
!
interface FastEthernet0/0.40
 encapsulation dot1Q 40
 ip address 172.19.90.97 255.255.255.224
!
interface FastEthernet0/0.50
 encapsulation dot1Q 50
 ip address 172.19.90.129 255.255.255.224
!
!

```

Se establece enrutamiento dinámico con OSPF apuntando a los otros dos routers de la empresa mas una ruta estática.

```

IOS Command Line Interface
!
router ospf 5
 log-adjacency-changes
 network 172.19.90.0 0.0.0.31 area 0
 network 172.19.90.32 0.0.0.31 area 0
 network 172.19.90.64 0.0.0.31 area 0
 network 172.19.90.96 0.0.0.31 area 0
 network 172.19.90.128 0.0.0.31 area 0
 network 172.19.90.160 0.0.0.3 area 0
!
router rip
!
ip classless
ip route 172.19.90.164 255.255.255.252 172.19.90.162
!
!

```

Por último una reseña a los parámetros habituales de seguridad para evitar accesos no deseados a los router de la empresa.

```

IOS Command Line Interface
!
line con 0
 password cisco
 login
!
line aux 0
!
line vty 0 4
 password cisco
 login
!
!
!
end

```

13) Pruebas

El proceso de realizar pruebas sobre el proyecto ha sido continuo y ha abarcado todas las áreas que se indican en este documento. Desde la BBDD con MySQL, HTML5, PHP, Packet Tracer, etc...

Cada módulo o grupo de código que se ha ido implementando se ha probado, tanto en su funcionamiento como buscando su optimización (comodidad al manejar la interfaz, mejor aspecto, solucionar errores en el código, ...).

- **MySQL.** Antes de implementar cada tabla se ha probado su funcionamiento, mejorado el nombre de las entradas o añadiendo posteriormente nuevos campos para dotar al conjunto de un aspecto mas profesional.
- **HTML5.** Cada página se ha testeado numerosas veces, antes de añadir una nueva hay que asegurar que la presente no solo no presente fallos sino que además tenga el mejor aspecto posible.
- **PHP.** Cada bloque de comprobaciones, sesiones y envío de datos contra las tablas ha sido debidamente probado con los posibles errores que pudiera presentar, intentando así ofrecer un producto “a prueba de fallos”.
- Con el **Packet Tracer** se ha utilizado el método ensayo-error. A la hora de configurar VLANS y su enrutamiento por internet ha requerido gran cantidad de horas de pruebas y modificaciones sobre la idea original del proyecto a fin de lograr los objetivos iniciales. Por desgracia esta herramienta presenta ciertas limitaciones y errores a la hora de llevar a cabo ciertas tareas.

14) Conclusiones

El presente proyecto de fin de grado ha sido una experiencia gratificante, me ha permitido poner en práctica muchos de los conocimientos que he adquirido durante estos dos años que ha durado el ciclo formativo. También ha supuesto la posibilidad de aprender nuevas técnicas y adquirir mayor profundidad en algunas áreas que antes resultaban mas oscuras. La necesidad constante de documentarse y buscar solución a pequeños o grandes errores ha sido por un lado satisfactorio (cuando logras que funcione) o tremendamente frustrante cuando día tras día chocas contra el mismo obstáculo.

Al final el presente proyecto dista en algunos puntos del planteamiento inicial que hice allá por el mes de Noviembre, cuando entregué el borrador. Secciones como la de “redes” han visto reducido su peso en el presente proyecto tras los continuos fallos o limitaciones que presentaban herramientas como el “Packet Tracer” para llevar a cabo algunas tareas. Otras en cambio como el diseño web y su relación con la BBDD han resultado mas intensas en su desarrollo y han acabado por abarcar más de lo esperado.

Personalmente estoy contento por el resultado final y considero que las cerca de 500 horas empleadas para unas cosas u otras han estado bien empleadas y me dejan con la sensación del trabajo bien hecho. Son muchas las cosas que podría seguir añadiendo al presente proyecto, pero es importante saber poner un límite y no dejar que las ganas de ampliar o mejorar acaben por volver interminable el trabajo.

15) Aspectos a mejorar

Son varios los ámbitos de mejora dentro de este proyecto. Empezando por la página web, sería recomendable habilitar una sección para personal administrativo donde puedan acceder a una serie de recursos compartidos en red en los que intercambiar información (servicio semejante a un FTP).

A nivel de red se deberían implementar más medidas de seguridad como unas listas ACL y un funcionamiento óptimo de la conexión VPN.

En cuanto al PHP también podrían implementarse más medidas de protección para evitar la inyección de código o ampliar ciertos campos den las tablas de MySQL para poder trabajar con más valores.