



Music Band Center!!

Tus Instrumentos a medida!!.

Proyecto Integrado.

Curso: Desarrollo de Aplicaciones Informáticas (DAI).

Alumno: José Manuel Fernández Verano.

Año: 2011/2012.

Índice.

<u>Apartado</u>	<u>Numero Pagina.</u>
Descripción del Sistema	1
Catalogo de Requisitos	2-4
Modelo Conceptual de Datos (ER)	5
Modelo Lógico de Datos (Modelo Relacionar)	6
Tecnologías Empleadas y Justificación de su Uso	7
Implantación del Software	8-9

1. Descripción del Sistema.

El sistema de gestión realizado tiene como principal función el ofrecer la creación de un instrumento de forma asistida para los cliente y la administración de productos para la empresa. A continuación se explica las operaciones que realizan cada unos de los sistemas, cabe destacar que los sistemas son independientes, ya que solo tiene en común la base de datos.

1. Web Administración (Background).

En este sistema se controlara todo lo referente a productos, en él se realizaran las operaciones de alta, baja, modificación y consultas, tanto para los productos como para los clientes registrados, también se establecerá el control de pedidos y realización del mismo, al que tendrá acceso solo el personal encargado de la creación del instrumento personalizado.

2. Web Cliente (Foreground).

Este sistema tiene como fin la obtención de un instrumento ya sea bajo o guitarra totalmente personalizado, y diseñado pieza a pieza por el cliente, además el sistema le permitirá ver todos los productos registrados en nuestra web, pudiendo ver con detalle la forma física del mismo a través de las imagenes disponibles.

El proceso de creación de instrumento consiste en la selección un tipo de cuerpo concreto de guitarra o bajo y sobre él ir seleccionando todas las piezas necesarias para el montaje completo de la guitarra o bajo, también es posible la creación de un instrumento de forma parcial, ya que el sistema permitirá continuar con el proceso de creación sin seleccionar todos los productos.

Además también se permitirá volver atrás en el proceso para seleccionar una pieza que pudo no ser seleccionada o bien si se quiere cambiar por otra. Tras la selección del primer elemento se activara la cesta de compra pudiendo mirar su contenido en todo momento, además del precio total de los productos seleccionados.

Tras realizar el pedido el usuario podrá ver en todo momento el estado en el que se encuentra el pedido, obteniendo una factura mediante pdf o bien mediante web.

Además el usuario podrá en cualquier momento borrar su usuario, eliminado todos los datos almacenados en momento del registro. Lo único que se quedará almacenado serán las facturas.

2. Catálogo de Requisitos.

1. Requisitos Funcionales (RF) y Objetivos.

1. Web Administración (Background).

1. OBJ1 Gestión de Cliente.

1. RF1 – Login.
2. RF2 – Consulta.
3. RF3 – Baja.
4. RF4 – Modificación.
5. RF5 – Alta.

2. OBJ2 Gestión de Productos.

1. RF1 – Alta.
2. RF2 – Baja.
3. RF3 – Consulta.
4. RF4 – Modificación.

3. OBJ3 Gestión de Pedido.

1. RF1 – Modificación.
2. RF2 – Consulta.
3. RF3 – Baja.

4. OBJ4 Gestión de Proveedores.

1. RF1 – Alta.
2. RF2 – Baja.
3. RF3 – Consulta.
4. RF4 – Modificación.

5. OBJ5 Gestión de Factura.

1. RF1 – Modificación.
2. RF2 – Consulta.
3. RF3 – Baja.

2. Web Cliente (Foreground).

1. OBJ1 Gestión de Cliente.

1. RF1 – Login.
2. RF2 – Alta.
3. RF3 – Baja.
4. RF4 – Modificación.

2. OBJ2 Gestión de Producto.

1. RF1 – Consulta.

3. OBJ3 Gestión de Pedido.

1. RF1 – Alta.
2. RF2 – Consulta.

4. OBJ4 Gestión de Factura.

1. RF1 – Alta.
2. RF2 – Consulta.

2. Requisitos No Funcionales.

1. RNF1 – Apariencia de la Web Cuidada (Prioridad Media).
2. RNF2 – Interactuar con el usuario (Prioridad Media).
3. RNF3 – Procesos eficientes (Prioridad Alta).
4. RNF4 – Acceso eficiente a la Base de Datos (Prioridad Alta).
5. RNF5 – Compatibilidad con PHP5 y Servidor Apache2 (Prioridad Alta).
6. RNF6 – Aplicación segura, estable y fácilmente ampliable (Prioridad Media).

3. Requisitos de Almacenamiento de Información (RAI).

Los datos indicados a continuación serán los que se deberán guardar, para la gestión y tratamiento del sistema.

1. Usuarios.

1. Email (Deberá ser Único).
2. DNI.
3. Nombre.
4. Apellidos.
5. Dirección.
6. Teléfono.

2. Proveedores.

1. Nombre.
2. CIF/NIF.
3. Teléfono.
4. Email.

3. Productos.

1. Nombre.
2. Stock.
3. Descripción.
4. Fabricante.
5. Precio.
6. Familia de Instrumento.
7. Compatibilidad con estilo.

4. Pedidos.

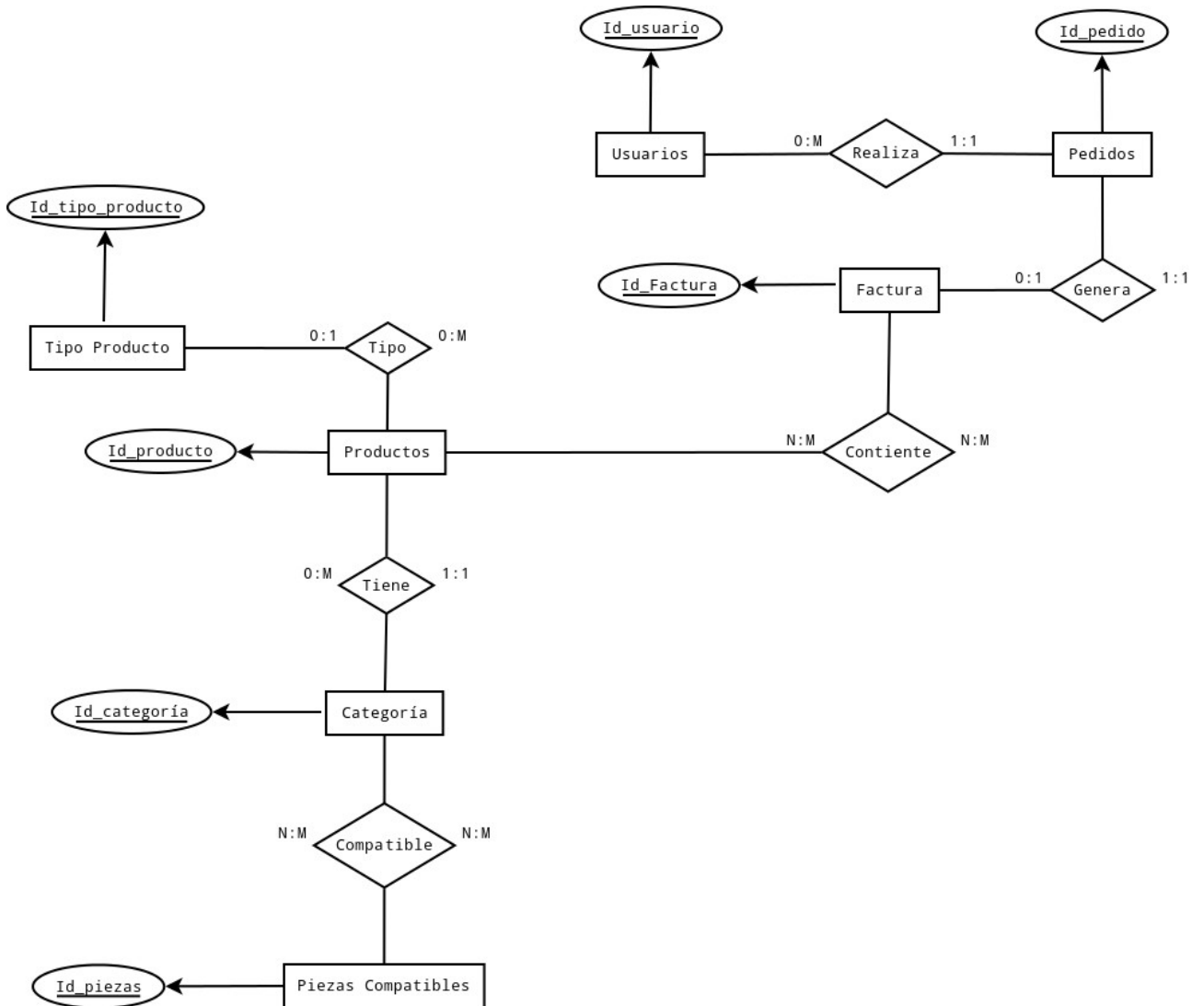
1. Usuario que realiza el pedido.
2. Fecha.
3. Estado del Pedido.

5. Facturas.

1. Numero de Pedido.
2. Nombre del Usuario (*).
3. DNI del Usuario (*).
4. Apellidos del Usuario (*).
5. Dirección del Usuario (*).
6. Teléfono del Usuario (*).
7. Fecha.
8. Todos los Productos del Pedido con su cantidad y precio.

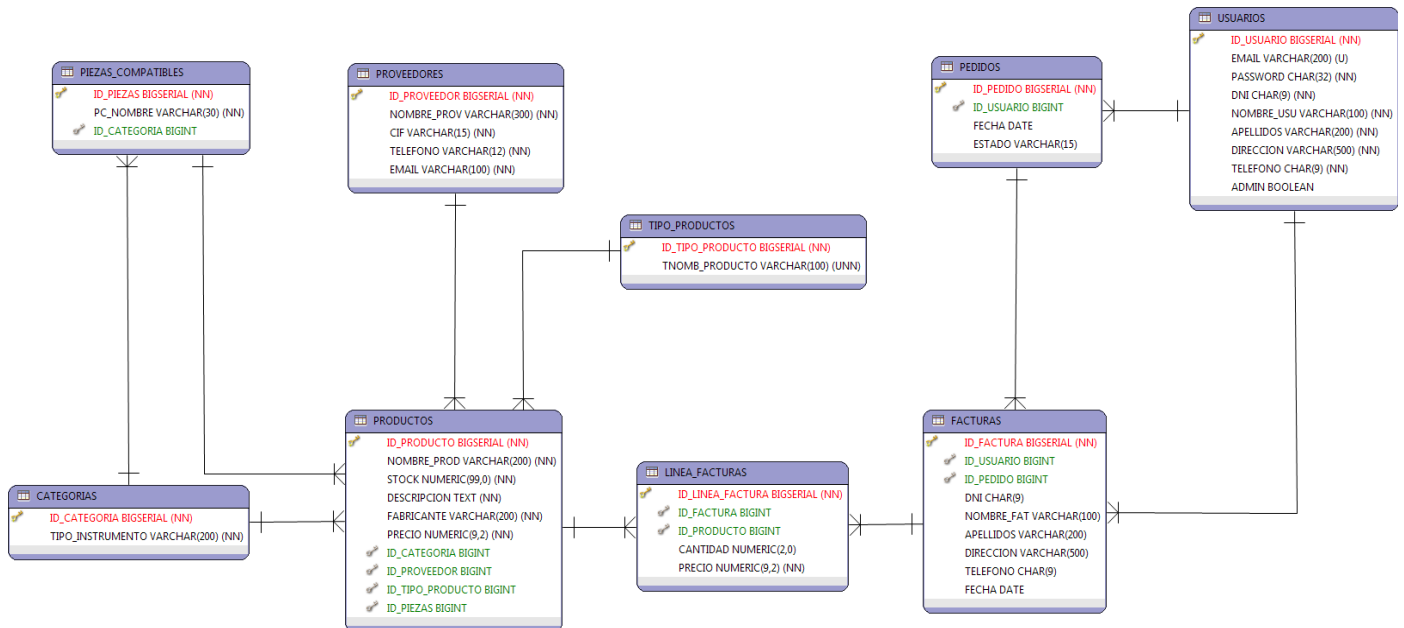
(*) Estos datos copiarán por la posible eliminación del usuario en un futuro.

3. Modelo Conceptual de Datos (Modelo ER).



Solo he incluido los atributos que serán clave primaria en las tablas para aumentar la legibilidad del diagrama.

4. Modelo Lógico de Datos (Modelo Relacional).



usuarios (id_usuario, email, password, dni, nombre_usu, apellidos, direccion, telefono, admin).

proveedores (id_proveedor, nombre_prov, cif, telefono, email).

categorias(id_categoria, tipo_instrumento).

piezas_compatibles (id_piezas, pc_nombre, id_categoria).

tipo_productos (id_tipo_producto, tnomb_producto).

productos(id_producto, nombre_prod, stock, descripcion, fabricante, precio, id_categoria, id_proveedor, id_tipo_producto, id_piezas).

pedidos(id_pedido, id_usuario, fecha, estado).

facturas(id_factura, dni, nombre_fat, apellidos, direccion, telefono, fecha, id_usuario, id_pedido).

linea_facturas (id_linea_factura, cantidad, precio, id_producto, id_factura).

Las palabras subrayadas en color negro son las claves primarias y las de color azul son las claves foráneas.

5. Tecnologías Empleadas y Justificación de su Uso.

1. Sistema Operativo.

El sistema operativo seleccionado es GNU/Linux Ubuntu 12,04 ya que por su estabilidad y sus grandes prestaciones, lo hace idóneo para alojar nuestra aplicación. Además este sistema operativo es de software libre, evitando así los posibles pagos por las actualizaciones del mismo.

2. Sistema de Base de Datos.

El sistema de base de datos seleccionado es PostgreSQL, este sistema de base de datos es uno de los más potentes del mercado, además de ser software libre, el cual no tendremos que pagar ni por su adquisición ni por sus actualizaciones, PostgreSQL está totalmente integrado en Ubuntu, permitiendo la gestión rápida y fluida a través del terminal. PostgreSQL tiene soporte para orientación a objetos y cumple con los estándares de SQL.

3. Servidor Web.

El servidor web seleccionado es por excelencia el Apache2 ya que es uno de servidores más potentes que podemos encontrar. Apache2 contiene la licencia GNU GPL, está totalmente integrado en Ubuntu, además le tendremos que instalar los módulos adicionales de PHP5 y PostgreSQL entre otros.

4. Selección del Lenguaje de Programación.

El lenguaje seleccionado para la creación de la aplicación es PHP5, este lenguaje nos permitirá de forma sencilla y rápida la implementación de todas las librerías necesarias para la creación de nuestra web, además de un completo acceso a la base de datos. PHP es un lenguaje interpretado que se usará solo en el servidor, al cliente solo llegará código HTML.

Para la ejecución en el cliente usaremos JavaScript, al igual que PHP es un lenguaje interpretado pero esta vez por el cliente que recibirá la web, con él trataremos todos los procesos que sea necesario ya sea adaptación de estilo, abrir web emergentes, mostrar fecha, etc...

5. Selección de Framework.

En este caso buscamos un framework tipo Modelo, Vista, Controlador (MVC) en PHP, en este caso seleccionamos el framework CodeIgniter, más que nada por su sencillez y la libertad que nos da a la hora de diseñar las vistas, además CodeIgniter contiene "Helpers" que nos ayudarán a realizar determinadas operaciones de forma asistida. Otra cosa a destacar es la completa documentación totalmente detallada e incluso en algunos métodos con las recetas de elaboración.

6. Hoja de Estilo.

Para generar el estilo de la web usaremos las hojas de estilo CSS, creando un fichero que luego añadiremos a nuestra web, el fichero CSS estará formado por clase e id, para así controlar de forma sencilla todas las capas de la Web.

7. Lenguaje de Marca.

El lenguaje de Marca seleccionado es el HTML5 ya que es la última versión e incorpora muchas novedades sobre su antecesor. Además con él podremos seleccionar si desarrollamos en modo clásico o en modo XHTML, esta última deberá ser servida como XML, pudiendo desarrollar en paralelo HTML y XML.

6. Implantación del Software.

La implantación del software de deberá realizar en un sistema operativo Ubuntu 12,04 ya sea de arquitectura x86 o x86_64, también se podrá instalar en un sistema operativo que este basado en Debian y que además disponga de los programas necesarios en su repositorio.

El proceso de instalación del software se podrá hacer de forma automatizado mediante una series de script, hay que tener en cuenta que estos script solo han sido probados en el sistema operativo Ubuntu 12,04

1. Instalación mediante script (Ubuntu 12.04).

Para realizar la instalación mediante el script, introduciremos el cd y buscaremos en la raíz un archivo que se llama "instalador.sh", lo tendremos que ejecutar como super usuario en el terminal, el script instalara y las aplicaciones Apache2, PHP5, php5-pgsql y PostgreSQL 9,1.

Cuando el proceso termine el terminal nos indicara diciéndonos que ejecutemos el siguiente script.

Ahora ejecutaremos un archivo que se llama "preparacion_directorio.sh", este script creara los usuarios de la base de datos y creara la estructura de directorios necesaria para la ejecución correcta de la aplicación. Durante el proceso nos pedida que el especifiquemos la contraseña para el usuario, la contraseña preconfigurada en la web "usuario", por tanto le pondremos esa. Si por cualquier motivo queremos usar otra, solo tendremos que cambiar el archivo "config" del framework.

Cuando el proceso termine pasaremos al paso 3.

2. Instalación manual (Ubuntu u otro sistema base Debian).

Si tenemos algún problema con el script y no podemos ejecutarlo estos son los pasos a seguir para instalar todos los programas necesarios. Hay que tener en cuenta que es necesario que el sistema operativo sea Ubuntu o en si defecto que este basado en Debian y que disponga de los paquetes en su repositorio.

1. Instalación PostgreSQL.

Para la instalación de PostgreSQL teclearemos la siguiente linea en el terminal.

```
sudo apt-get install postgresql-9.1
```

2. Instalación Apache2 y PHP.

Ahora instalaremos el Apache2, PHP5 y el modulo de comunicación de PHP5 para PostgreSQL, para ello tecleamos esto en el terminal.

```
sudo apt-get install apache2 php5 php5-pgsql
```

3. Ahora copiamos los datos de la web en el servidor, ejecutaremos la siguiente linea

```
sudo cp -rv proyecto/ /var/www
```

4. Ahora crearemos la base de datos y el usuario que utilizara la web, para realizar las consultas, para ello ejecutaremos las siguientes lineas seguidas.

```
su -c "createuser -P usuario" postgres
```

```
su -c "createdb -O usuario tienda" postgres
```

La clave preconfigurada para el usuario de la base de datos es "usuario", así que cuando te pida que teclees la clave tendrás que poner esa o bien si pones otra tendrás que cambiarla en el archivo config del framework.

5. Solo queda añadir las tablas a la base de datos, las tablas se encuentran en el mismo directorio de la web, para ello teclearemos la siguientes linea en el terminal.

```
psql -U usuario tienda < /var/www/proyecto/comun/sql/tienda.sql
```

Ahora continua con el paso 3.

3. En este paso revisaremos que la codificación del apache es la correcta y que podemos acceder a la base de datos desde la misma maquina en la que esta alojada, este proceso no es obligatorio pero si es interesante para realizar posibles consultas en el futuro.

1. Comprobar la codificación del Apache2.

En el terminal ponemos la siguiente linea...

```
sudo gedit /etc/apache2/conf.d/charset
```

Cuando se abra el gedit comprobamos que AddDefaultCharset UTF-8 este sin almohadillar, si por el contrario la tiene la quitamos, guardamos y introducimos la siguiente linea para reiniciar el servidor.

```
sudo service apache2 restart
```

2. Configuración de PostgreSQL para acceder en modo localhost.

Para ello realizaremos una modificación en el archivo del sistema, teclearemos las siguiente linea en el terminal.

```
sudo gedit/etc/postgresql/9,1/main/pg_hba.conf
```

En el documento que se no abre buscamos la siguiente linea.

```
# "local" is for Unix domain socket connections only
```

```
local all all ident
```

```
# IPv4 local connections:
```

```
host all all 127.0.0.1/32 md5
```

```
# IPv6 local connections:
```

```
host all all ::1/128 md5
```

En la primera que pone "local all all ident", cambiamos el "ident" por "md5", quedaría así.

```
# "local" is for Unix domain socket connections only
```

```
local all all md5
```

```
# IPv4 local connections:
```

```
host all all 127.0.0.1/32 md5
```

```
# IPv6 local connections:
```

```
host all all ::1/128 md5
```

Ahora ya solo queda reiniciar el sistema de base datos para que podamos acceder desde nuestro equipo local, teclearemos la siguiente linea en el terminal.

```
sudo service postgresql restart
```

Proceso de instalación completado.

