



UNIVERSIDAD  
POLITECNICA  
DE VALENCIA



Escola Tècnica  
Superior d'Enginyeria  
Informàtica

# Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

---

Memoria del Proyecto Final de Carrera

**Titulación: Ingeniería Superior de Informática**

**Fecha: 1 de Julio de 2011**

**Autor: Adrián Castillo Sánchez**

**Director: Sergio Sáez Barona**

**Codirector: Javier Vicente Jiménez Belenguer**

<b>1</b>	<b>INTRODUCCIÓN.....</b>	<b>4</b>
<b>2</b>	<b>ESPECIFICACIÓN DE REQUISITOS .....</b>	<b>6</b>
2.1.	INTRODUCCIÓN .....	6
2.1.1.	PROPÓSITO.....	6
2.1.2.	ÁMBITO.....	6
2.1.3.	DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS.....	7
2.1.4.	REFERENCIAS.....	7
2.1.5.	VISIÓN GENERAL .....	8
2.2.	DESCRIPCIÓN GENERAL .....	8
2.2.1.	PERSPECTIVAS DEL PRODUCTO.....	8
2.2.2.	FUNCIONES DEL PRODUCTO .....	9
2.2.3.	CARACTERÍSTICAS DEL USUARIO.....	10
2.2.4.	RESTRICCIONES GENERALES.....	10
2.2.5.	SUPUESTOS Y DEPENDENCIAS .....	11
2.2.6.	REQUISITOS PENDIENTES.....	11
2.3.	REQUISITOS ESPECÍFICOS .....	11
2.3.1.	REQUISITOS FUNCIONALES .....	12
2.3.1.1.	USUARIO .....	12
2.3.1.2.	USUARIO CON PERMISOS DE ALMACÉN .....	12
2.3.1.3.	USUARIO CON PERMISOS DE COMPRAS.....	13
2.3.1.4.	USUARIO CON PERMISOS DE INGENIERÍA .....	13
2.3.1.5.	USUARIO ADMINISTRADOR .....	14
2.3.2.	REQUISITOS DE INTERFACES EXTERNOS .....	14
2.3.2.1.	INTERFAZ DE USUARIO.....	14
2.3.2.2.	INTERFACES HARDWARE .....	14
2.3.2.3.	INTERFACES SOFTWARE.....	15
2.3.2.4.	INTERFACES DE COMUNICACIONES.....	15
2.3.3.	REQUISITOS DE EFICIENCIA .....	15
2.3.4.	OBLIGACIONES DE DISEÑO.....	15
2.3.4.1.	ESTÁNDARES .....	15
2.3.4.2.	LIMITACIONES HARDWARE.....	16
2.3.5.	ATRIBUTOS.....	16
2.3.5.1.	SEGURIDAD .....	16
2.3.5.2.	MANTENIMIENTO .....	16
2.3.6.	OTROS REQUISITOS.....	17
2.3.6.1.	BASE DE DATOS.....	17
<b>3</b>	<b>ANÁLISIS .....</b>	<b>18</b>
3.1	CASOS DE USO .....	18
3.2	DIAGRAMAS DE SECUENCIA .....	46
3.3	DIAGRAMA UML.....	49
3.4	DISEÑO ENTIDAD-RELACIÓN .....	51
<b>4</b>	<b>DISEÑO .....</b>	<b>52</b>
4.1	NIVEL DE INTERFAZ.....	53
4.1.1	DIAGRAMAS DE NAVEGABILIDAD .....	57
4.1.1.1	USUARIO .....	57
4.1.1.2	USUARIO CON PERMISO DE ADMINISTRACIÓN EN ÓRDENES DE TRABAJO.....	58
4.1.1.3	USUARIO CON PERMISO ALMACÉN .....	59
4.1.1.4	USUARIO CON PERMISO DE INGENIERÍA .....	60

## Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

4.1.1.5	USUARIO CON PERMISO DE COMPRAS.....	61
4.1.1.6	USUARIO ADMINISTRADOR .....	62
4.2	NIVEL LÓGICO .....	63
4.3	NIVEL DE PERSISTENCIA.....	63
4.3.1	ESQUEMA DE LA BASE DE DATOS .....	64
<b>5</b>	<b>IMPLEMENTACIÓN E INTEGRACIÓN .....</b>	<b>66</b>
5.1	TECNOLOGÍAS.....	66
5.2	HERRAMIENTAS.....	67
5.3	DETALLES DE LA IMPLEMENTACIÓN .....	68
5.3.1	PERFILES DE USUARIO.....	68
5.3.2	AUTENTICACIÓN DE USUARIOS.....	69
5.3.3	ASPECTOS GENERALES .....	69
5.4	INTEGRACIÓN .....	70
<b>6</b>	<b>EVALUACIÓN Y PRUEBAS .....</b>	<b>72</b>
6.1	EVALUACIÓN .....	72
6.2	PRUEBAS.....	72
6.2.1	PRUEBA SOLICITUD DE ALTA DE UN NUEVO COMPONENTE .....	72
6.2.2	PRUEBA CONSULTA SOBRE INFORMACIÓN DE CUALQUIER COMPONENTE.....	74
6.2.3	PRUEBA CONSULTA SOBRE INFORMACIÓN DE CUALQUIER ÓRDEN DE FABRICACIÓN .....	76
6.2.4	PRUEBA EXTRAER O QUITAR STOCK DE UN COMPONENTE EN UNA UBICACIÓN DETERMINADA .....	79
6.2.5	PRUEBA HACER UN KITTING .....	81
6.2.6	PRUEBA SOLICITUD DE COMPRA .....	83
<b>7</b>	<b>CONCLUSIONES .....</b>	<b>86</b>
	<b>BIBLIOGRAFÍA.....</b>	<b>87</b>
	<b>ANEXO A. DESCRIPCIÓN DE TECNOLOGÍAS .....</b>	<b>88</b>
A.1.	HTTP .....	88
A.2.	PHP .....	89
	<i>Ventajas .....</i>	<i>90</i>
	<i>Inconvenientes .....</i>	<i>91</i>
A.3.	CSS.....	91
	LOS TRES TIPOS DE ESTILOS .....	91
	VENTAJAS DE USAR LAS HOJAS DE ESTILO .....	92
A.4.	LINUX.....	92
A.5.	HTML .....	93
	NOCIONES BÁSICAS DE HTML .....	94
	ACCESIBILIDAD WEB .....	95
A.6.	SERVIDOR HTTP APACHE .....	95
	VENTAJAS .....	96
	USO.....	96
A.7.	MYSQL .....	97
	APLICACIONES .....	97

# 1 INTRODUCCIÓN

La aplicación en PHP supone la creación de una herramienta, orientada al Web, para la administración de un inventariado de componentes comerciales como de diseño propio, codificación automática de los mismos, consultoría de información sobre fabricantes, proveedores, ofertas, stock y en que estructuras de productos se están utilizando, gestión global de solicitudes de compra y posteriores pedidos, organización del almacén (adición y extracción de stock), realización de “kitting” para obtener las diferentes cantidades de los componentes que forman una estructura de un producto, gestión de las órdenes de trabajo dentro de la empresa.

Otros requisitos más básicos son de tipo administrativo que conllevan el alta de un nuevo usuario en el sistema, eliminación y modificación de información de los diferentes componentes dentro del sistema, consultoría de la base de datos mediante el lenguaje SQL.

Nuestra aplicación tendrá como finalidad llegar a ser una E.R.P para la empresa que se irá expandiendo con diferentes módulos como la posible recepción de los pedidos, solicitudes de compra automáticas, gestiones financieras según la orden de trabajo o la verificación de los pedidos comprobando los diferentes estados por los que tiene que pasar (IPC, cantidades correctas, precios, etc..)

- **Las motivaciones:**

Las motivaciones que me han llevado a emprender la realización de este Proyecto Final de Carrera podríamos definir las en los siguientes puntos, técnicos y humanos, los cuales me han servido como ensayo general antes de la inminente salida de la Universidad hacia la próspera vida laboral.

Mis aspiraciones técnicas fueron el hecho de afrontar un reto importante como la programación de una aplicación de esta índole dentro de un ámbito empresarial y la adaptación a unas herramientas ya impuestas dentro de la empresa, como la programación en PHP, uso de las bases de datos en MYSQL y la creación de plantillas CSS y páginas en HTML.

Uno de los factores más importante que hizo que el proyecto fuera motivador para mí fue el hecho de la usabilidad de las Bases de Datos durante toda mi carrera y en cuyo reto ésta parte era muy importante. El hecho de que durante todo este tiempo haya estado cumpliendo por parte de la empresa y por mi parte objetivo tras objetivo me ha ido generando una satisfacción personal muy gratificante y provocando que dentro del proyecto inicial propusiese diversas funcionalidades futuras para mejorar procedimientos y que se estudiarán por parte de la empresa.

Mis aspiraciones humanas fueron enfocadas a la preparación a la hora de mantener reuniones con entidades importantes para acordar objetivos sobre el proyecto, mejorar o modificar cosas sobre el mismo, innovar o intentar hacer realidad diversas ideas para mejorar en lo posible el funcionamiento de diversos procedimientos dentro de la empresa.

A nivel personal me ha servido para mejorar en diversos aspectos como la organización del código, pensar en la futura administración del mismo, facilidad a la hora de modificar ciertos aspectos del proyecto,

tener iniciativa propia aunque se escape del objetivo principal por la posibilidad de incorporarse dentro de la aplicación y a cumplir plazos de entrega.

- **Planteamiento técnico:**

El objetivo del planteamiento técnico ha sido dotar a la aplicación de varios puntos:

1. Interfaz de usuario amigable y eficiente
2. Portabilidad y capaz en un futuro de adaptarse a nuevas tecnologías.
3. Desarrollo sobre tecnología Linux, Apache, PHP y MySQL.
4. Interfaz Web que pueda tener acceso desde cualquier navegador web.

Se ha desarrollado sobre estos puntos una aplicación que simplificase la interacción entre el usuario y la máquina al mismo tiempo que resultase llamativa, elegante, entendible y eficaz. Para ello he tenido en cuenta un entorno gráfico muy detallado, potente, intuitivo y de rápido aprendizaje por parte del usuario. Diversos estudios hechos anteriormente por Internet demuestran que existen aplicaciones potentes pero el motivo de la complejidad de las mismas hace que sean rechazadas por lo que en nuestro caso hemos evitado al máximo todo ello.

De esta manera he llegado a la compatibilidad y portabilidad entre navegadores web, haciendo más exacta la representación web cuanto más exacto sea el navegador. Los mejores resultados se obtienen con los navegadores “Mozilla” sin dejar de representarse correctamente con otros navegadores de versiones “Internet Explorer de Microsoft o Opera”.

Quiero destacar de las tecnologías utilizadas lo denominado como “futurabilidad” y la portabilidad, entendiendo por la misma la capacidad para ser multiplataforma y de alternarlas sin perder funcionalidad, y por “futurabilidad” la capacidad del portal para sostenerse en el futuro al basarse en tecnologías estándares y lenguajes de amplia difusión y establecidos.

El desarrollo sobre las tecnologías “Linux + Apache + PHP + MySQL” permite instalar la aplicación bajo un mínimo presupuesto: coste mínimo del hardware para el funcionamiento del servidor: todo ordenador podría solucionar las necesidades que, en caso de ser mayores, se podría recurrir a hospedaje compartido o servidor profesional dedicado a ello. Ambas opciones están ampliamente respaldadas por la práctica totalidad de servidores de pago o contrato con soporte PHP y MySQL, siendo las mudanzas sencillas y claras para cualquier usuario. En nuestro caso nos encontramos con un servidor profesional dedicado a ello que incluso en la actualidad se está pensando cambiar con lo que no conllevará grandes cambios por lo mencionado anteriormente.

Por otro lado, los gastos del cliente son nulos ya que para utilizar la aplicación sólo se necesitaría un navegador web de los disponibles en mercado, ahorrándonos la instalación de cualquier aplicación que actúe de cliente, siendo incluso necesario para la gestión administrativa de la misma.

Debido a su amplia funcionalidad desde el navegador web, se puede acceder desde cualquier ordenador conectado a Internet haciendo que el potencial de usuarios sea casi global.

# 2 ESPECIFICACIÓN DE REQUISITOS

## 2.1. INTRODUCCIÓN

En este punto se hará una descripción completa del comportamiento del sistema que se va a desarrollar a lo largo de todo el documento. Aspectos tan importantes como el ámbito donde se va a desenvolver la aplicación, el propósito u objetivo por el cual se ha desarrollado, perspectivas del producto, requisitos funcionales y de usuario así como posibles mejoras.

### 2.1.1. PROPÓSITO

El propósito del siguiente apartado consiste en la especificación de requisitos software que será útil para establecer unos prerrequisitos y el principio de todo el trabajo realizado y servirá como fuente de información para que cualquiera pueda conocer de una forma mejor y más detallada todo el proyecto.

### 2.1.2. ÁMBITO

La aplicación que se accede mediante Web en la intranet propia de la empresa y consistirá en una interfaz dinámica que facilitará la consulta, compras, pedidos, órdenes de fabricación y gestión de stock dentro del almacén. Dependiendo del tipo de usuario se dispondrá de diferentes operaciones según el nivel de privilegios o permisos que se tenga.

Las operaciones básicas de usuario será la consulta de tipo informativo sobre cualquier componente insertado en la base de datos (fabricantes, proveedores, ofertas, donde está utilizado, ubicaciones y stock). Ofrecerá también un servicio de altas de componentes para su posterior codificación dentro del sistema e inserción en la base de datos después de una supervisión del administrador de ésta área. Tendrá también un acceso a sus datos personales, módulo de consultas mediante consultas SQL y consultoría sobre todas las órdenes de fabricación con sus respectivos datos como proyecto al que pertenece, fechas de realización, responsable, componentes afectados, facturación y pedidos relacionados.

El usuario administrador tendrá a parte de todas las funcionalidades anteriores se encargará de la modificación de diversos datos importantes de cualquier componente como puede ser la inserción de un nuevo fabricante o modificación de alguno existente, modificación o eliminación de algún componente, insertar, eliminar o modificar alguna oferta, insertar, modificar o eliminar alguna ubicación, supervisar el alta de componentes nuevos o de órdenes de fabricación nuevas y dar de alta masivamente componentes respetando el formato acordado para hacerlo.

Según el nivel de permisos que se obtenga se puede ser un usuario intermedio a los dos casos extremos anteriores. Si el usuario tiene un perfil financiero se podrá tener acceso al módulo de compras donde se tienen funcionalidades como solicitudes de compra y obtener información de cómo progresa cada uno de ellas sobre los distintos niveles que tienen. El administrador en este tipo de perfil tendrá acceso a la firma y aceptación de las solicitudes convirtiéndose en pedidos y a toda la información adicional que se pueda introducir.

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

Otro perfil es el de usuario de almacén donde tendrá accesos a la adición o extracción de stock para su consumo. Se podrá crear nuevas ubicaciones para componentes nuevos ayudados de una aplicación de creación de ubicaciones basada en el mapa de la empresa. Otra funcionalidad que tiene es la posibilidad de hacer un “kitting” que es como un carrito de la compra de varios componentes para su posterior extracción, pudiéndose hacer parcialmente y posteriormente recuperarlo para finalizarlo.

La funcionalidad principal de todo lo descrito es la posibilidad de administrar o trabajar de forma remota sobre los recursos sin ser necesario el acceso físico para realizar las operaciones aunque es recomendable no por motivo informático si no organizativo en el módulo de almacén hacer sus operaciones posteriores a las informáticas de manera física.

#### 2.1.3. DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS

- **HTML:** Acrónimo inglés de Hyper Text Markup Language es un lenguaje informático diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web.
- **CSS:** Las hojas de estilo en cascada (Cascading Style Sheets, CSS) son un lenguaje formal de ordenador usado para definir la presentación de un documento estructurado escrito en HTML o XML.
- **HTTP:** es el protocolo de la Web (WWW), usado en cada transacción. Las letras significan Hyper Text Transfer Protocol, es decir, protocolo de transferencia de hipertexto.
- **PHP:** Acrónimo de "PHP: Hypertext Preprocessor". Es un lenguaje de programación de scripts y se utiliza principalmente para la programación de CGIs para páginas web, destaca por su capacidad de ser embebido en el código HTML.
- **Web:** Es un sistema de hipertexto que funciona sobre Internet. Para ver la información se utiliza una aplicación llamada navegador web para extraer elementos de información de los servidores web y mostrarlos en la pantalla del usuario.
- **Web dinámica:** existen dos tipos de páginas web, de contenido estático (HTML) y de contenido dinámico que se generan a partir de lo que un usuario introduce en un web o formulario y que utiliza el servidor para construir una web personalizada que envía al cliente.
- **Apache:** Es un servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc), Windows y otras, implementa el protocolo HTTP/1.1 (RFC2616) y la noción de sitio virtual.
- **MYSQL:** MySQL es el servidor de bases de datos relacionales más popular, desarrollado y proporcionado por MySQL AB.
- **API:** (del inglés Application Programming Interface - Interface de Programación de Aplicaciones, interfaz de programación de la aplicación) es un conjunto de especificaciones de comunicación entre componentes software.
- **Usuario:** Se llamará usuario a cualquier persona que acceda a la aplicación. Se diferenciarán entre ellos por los permisos que puedan tener.
- **Kitting:** Se referenciará a este término a la acción de extracción de unidades de cualquier componente identificado en un listado dentro de la sección correspondiente en la aplicación.

#### 2.1.4. REFERENCIAS

La realización del siguiente documento se ha basado en las siguientes fuentes:

[1] Wikipedia en español: <http://es.wikipedia.org>

[2] Manual de PHP: <http://www.php.net>

[3] Especificaciones de HTML 1.0: <http://www.w3schools.com/html/default.asp>

[4] Documentación de Apache: <http://httpd.apache.org>

### 2.1.5. VISIÓN GENERAL

En el resto del documento se tratará todos los objetivos que se han conseguido así como se detallará y se desarrollará toda la información sobre las operaciones que se pueden hacer según qué tipo de usuario.

## 2.2. DESCRIPCIÓN GENERAL

En esta sección se describen todos aquellos factores que afectan al producto y a sus requisitos. No se describen los requisitos, sino su contexto. Esto permitirá definir con detalle los requisitos en la sección 3, haciendo que sean más fáciles de entender. Esta sección consta de los siguientes puntos: Perspectiva del producto, funciones del producto, características de los usuarios, restricciones, factores que se asumen y futuros requisitos.

### 2.2.1. PERSPECTIVAS DEL PRODUCTO

En el momento de nuestra llegada ya estaba implantado un Servidor Web y la base de datos MySQL dentro de una máquina Linux por lo que opté por desarrollar la aplicación en HTML con programación en PHP para generar el contenido dinámico producido por los usuarios con la introducción de datos. Posteriormente se accedería a la base de datos MySQL para comprobar, obtener o eliminar los datos necesarios para generar las páginas correspondientes. Por todo ello comuniqué la forma de proceder con este diseño de sistema por el soporte que proporcionaba al lenguaje PHP, MySQL y Servidor Web.

Se buscó la compatibilidad entre navegadores (Microsoft Internet Explorer versión 5.5 o superiores, Mozilla Firefox, Safari, Opera) para hacer más potente la aplicación.

Otro punto importante fue la forma de presentar la aplicación. Conllevó hacer una pequeña consulta a las directrices de la empresa sobre formatos, estándares y pantones para poder desarrollar tipos de letra, colores, formas de botones, tablas. La resolución a la que se optó fue la de 800x600 aunque lo más recomendable era 1024x768 para que todos los elementos representados cupieran sin problema alguno para cualquier máquina. Se trató de que fuese lo más intuitivo y amigable para el usuario ofreciendo una interfaz llamativa a la vista y separando los privilegios que tuviese cada uno.

Fue importante saber que para alojar la aplicación, tanto el servidor Apache como MySQL han de trabajar sobre un sistema cualquiera de Linux. Gran parte de las distribuciones Linux funcionan sobre máquinas 386, a excepción, en el caso de utilizar un PC Pentium II o mayor, porque hubiera que, usar un servidor que debe ejecutar PHP junto con base de datos MySQL. Se puede implantar los servidores con arquitectura diferente a Intel x86 (Sparc, Alpha,..). El software utilizado está formado por: distribución Linux, servidor web Apache con módulo PHP, servidor de base de datos MySQL y navegadores tales como Internet Explorer o Mozilla Firefox. Cualquier comunicación entre cliente y aplicación se gestiona a través del navegador web y el servidor de la misma a través de TCP/IP sobre el protocolo HTTP. Hay un límite de páginas para utilizar

simultáneamente para controlar el consumo de memoria que está relacionado con el hardware del PC, el ancho banda de la conexión y la velocidad del procesador.

#### 2.2.2. FUNCIONES DEL PRODUCTO

Las funciones que realizará la aplicación web serán las siguientes:

- I. **Inicio de sesión como usuario autenticado.** Se comprobará así los permisos que tiene cada uno de los usuarios a la hora de interactuar con la aplicación
- II. **Búsquedas sencillas sobre un dato o más avanzadas sobre varios especificándolos de forma estricta o parcial.** Esto dotará a la aplicación de un potente motor de búsqueda. Dichas búsquedas servirán para encontrar componentes, ofertas, fabricantes, proveedores, stocks u órdenes de fabricación.
- III. **Consultas de archivos adjuntos de los componentes dados de alta.**
- IV. **Dar de alta diversos componentes** con sus datos más relevantes, incluso archivo adjunto, así como sus fabricantes, proveedores, ofertas y ubicaciones y **la posibilidad de tener un seguimiento** del mismo así como de **posibles modificaciones**.
- V. **Dar de alta diversas órdenes de fabricación** para un mejor desarrollo de trabajo en la empresa. Se agregarán a la base de datos con la información más importante como el responsable, que componentes son afectados, fechas de trabajo, cantidades, facturación, horas y proyecto relacionado.
- VI. **Búsquedas sencillas o avanzadas sobre datos de las órdenes de fabricación.** Se podrá consultar cualquier información como por ejemplo los pedidos que existen con esa orden de fabricación, responsable, componentes...Para potenciar la búsqueda se podrá filtrar la misma por diferentes campos o por alguno de ellos.
- VII. **Se podrá operar sobre el almacén.** Las operaciones que se podrán hacer es la adición o extracción de stock, listados en formato Excel sobre alguna ubicación o ubicaciones buscadas o hacer kittings de manera parcial o completa.
- VIII. **Acceder a los datos personales para cambiarlos.**
- IX. **Posibilidad de sacar informes o búsquedas con el formato Excel** de forma personal bajo consultas SQL.
- X. **Operaciones sobre compras.** Se podrá hacer solicitudes de compra sobre componentes o cualquier otro equipo. La posibilidad de un seguimiento de firmas para su posterior pedido.
- XI. **A nivel administrativo de cualquier módulo o aplicación.** Según los permisos que se obtengan se podrá alguna o varias de las siguientes operaciones. Se tendrá acceso a la modificación, eliminación e inserción de cualquier componente, orden de fabricación, divisas, proveedores o de cualquier dato de una tabla (componente, fabricante del componente, oferta y ubicaciones). Se podrá dar de alta o baja a un usuario o dar permisos nuevos o quitarlos. A nivel de compras se podrán firmar las solicitudes de los usuarios. A nivel almacén se podrá dar de alta nuevas ubicaciones para diversos componentes.

#### 2.2.3. CARACTERÍSTICAS DEL USUARIO

Existen diversos tipos de usuario según los permisos que obtengan o se le atribuyan en cada momento. A continuación se relatarán diversos tipos de usuarios:

Uno de ellos será el usuario básico el cual podrá consultar cualquier información sobre cualquier componente de la base de datos. Obtendrá información sobre fabricantes, proveedores, ofertas, ubicaciones, estructuras donde se está utilizando, stock. Otra funcionalidad será el solicitar dar de alta un componente nuevo y mirar en qué estado se encuentra su petición de alta. También podrá modificar datos personales y hacer cualquier tipo de consulta sobre cualquier orden de trabajo dada de alta (responsable, componentes, pedidos, fechas de orden, proyecto, horas dedicadas y dinero dedicado).

Otro usuario sería el que obtiene permisos de almacén que teniendo todas las acciones del usuario básico además tiene funcionalidades como la adición y extracción de stock de cualquier componente, la creación de nuevas ubicaciones para componentes ayudado de una aplicación de codificación de ubicaciones sobre un mapa de la empresa, modificación y eliminación de ubicaciones y la posibilidad de hacer kitting completos o parciales (parecido a un carrito de la compra).

Otro tipo de usuario es el que obtiene los permisos de compras con los que podrá hacer solicitudes de compra y verificar el estado en el que se encuentran y donde cabe destacar otro usuario del tipo administrador que se encargará de firmar esas solicitudes para que posteriormente se conviertan en pedidos. Este último usuario aparte de tener las funcionalidades del usuario de compras básico también se encarga de la actualización del valor de las divisas.

Otro tipo de usuario es el encargado de la administración de las órdenes de fabricación las cuales podrá dar de alta en cualquier momento sin supervisores por el medio. Tendrá permisos de modificación de datos sensibles de la orden así como de la eliminación de la misma. Este tipo de usuario podrá obtener información de todos los pedidos que tenga una orden de fabricación a diferencia del usuario básico que solo podrá ver los pedidos hechos por él en cada una de ellas.

Por último existe un usuario administrador del sistema que se encargará de todo lo anterior y aparte tendrá acceso a la parte de administración de permisos para cada uno de los usuarios así como el dar de alta a un usuario nuevo dentro del sistema. Este usuario también se responsabilizará de dar de alta una lista de componentes mediante altas masivas si el fichero que las contiene está correctamente creado bajo la estructura acordada.

#### 2.2.4. RESTRICCIONES GENERALES

El usuario administrador podrá utilizar la base de datos de forma básica (modificar, eliminar, añadir,..) pero no modificar la estructura de la base datos al ser fija. Una modificación en la estructura de base de datos se efectuará a nivel de gestión de base de la propia Base de Datos utilizando una actualización del software y sólo podrá ser ejecutada por el administrador de base de datos.

Supongamos que se realiza una actividad frecuente sobre la aplicación, esto puede resultar en un acceso inadecuado a la misma por la propia saturación del servidor. Esta saturación puede deberse a los accesos a la

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

base de datos como a la ejecución del código PHP en el servidor web. Se difieren varias soluciones: la más simple sería aumentar la potencia del servidor con nuevo hardware o más compleja: pasar el servidor de MySQL a otro PC en la red local, introducir un proxy inverso en la entrada de la aplicación para que actúe como acelerador HTTP o construir un clúster de servidores web.

Tenemos que tener en cuenta el ancho de banda disponible del servidor para servir páginas ya que se puede convertir en el principal cuello de botella de la aplicación sino elegimos el adecuado, basándonos en la media de accesos producidos tanto al día como en horas punta.

Tiene que existir una política de seguridad adecuada en el servidor para evitar intrusiones al sistema, cabe destacar la configuración del servidor Apache. De la misma manera actuaríamos con el servidor MySQL y PHP. El sistema operativo debe disponer de un sistema de seguridad estricto para evitar intrusiones tales como cortafuegos, permisos, detectores de intrusiones,... Realizaremos de forma periódica copias de seguridad de: base de datos, páginas del servidor y configuraciones del servidor MySQL Y Apache, como medida básica de seguridad.

#### 2.2.5. SUPUESTOS Y DEPENDENCIAS

En un futuro podría establecerse un sistema más seguro y probado de autenticación de cualquier persona de la empresa mediante algún certificado digital firmado (por ejemplo, un certificado de la FNMT (CERES)) que probará la identidad de la persona para poder firmar por ejemplo las solicitudes de compra o cualquier operación sensible en el sistema.

#### 2.2.6. REQUISITOS PENDIENTES

Se deja como requisitos pendientes todo lo relacionado con la recepción de pedidos y los diferentes controles por los que pasan. Otro aspecto a mejorar serían las solicitudes de compra automáticas correspondientes a necesidades de stock a la hora de extracción o kitting pendientes. Otro aspecto importante para el futuro estará relacionado con la contabilidad que supone todos estos movimientos dentro del sistema e intentar relacionar todas las operaciones del mismo con la contabilidad de la empresa.

### 2.3. REQUISITOS ESPECÍFICOS

En este apartado se presentan los requisitos funcionales que deberían ser satisfechos por el sistema. Todos los requisitos aquí expuestos son esenciales, ya que, no sería aceptable un sistema que no satisfaga alguno de los requisitos presentados. Contiene los requisitos a un nivel de detalle suficiente como para permitir a los diseñadores poder diseñar un sistema que satisfaga estos requisitos, y que permita al equipo de pruebas planificar y realizar las pruebas que demuestren si el sistema satisface, o no, los requisitos. Todo requisito aquí especificado describirá comportamientos externos del sistema, perceptibles por parte de los usuarios, operadores y otros sistemas.

### 2.3.1. REQUISITOS FUNCIONALES

Esta sección se especificará todas aquellas acciones y funciones que deberá llevar a cabo el software.

#### 2.3.1.1. USUARIO

- **Búsquedas sobre uno o varios campos de cualquier dato disponible en el sistema**

Este tipo de acciones dota a la aplicación de una potente herramienta a la hora de encontrar información deseada o la posibilidad de hacer informes. El usuario introducirá los datos a buscar en los correspondientes campos de texto y en ese instante se producirá una consulta SQL a la base de datos, devolviendo en un formulario el resultado de la búsqueda.

- **Solicitud de alta de componentes**

Esta acción hace que el usuario rellene un formulario con los datos importantes del sistema para su posible alta en el mismo. Después de cumplimentar debidamente se introducirá en la base de datos de forma provisional hasta que un administrador la revise. En esta funcionalidad se podrá modificar por parte del usuario cualquier dato erróneo y reenviar la solicitud de nuevo con un mensaje explicativo. Todo lo ocurrido estará notificado por correo.

- **Modificación de la información personal**

El usuario podrá modificar toda la información relevante para el sistema a su agrado.

- **Informes o consultas mediante SQL**

Esta funcionalidad hace que el usuario pueda hacer sus propias consultas para hacer cualquier tipo de informe sobre datos del sistema. Está restringido a cierta información confidencial.

#### 2.3.1.2. USUARIO CON PERMISOS DE ALMACÉN

Tendrá las mismas funcionalidades que el usuario estándar y además:

- **Extracción o agregación de stock para cualquier componente**

Esta acción está restringida a usuarios con permisos de almacén donde se podrá, mediante una búsqueda previa, seleccionar un componente al que se desea extraerle o agregarle unidades recibidas en su stock. Esta misma operación se podrá hacer mediante una búsqueda por ubicaciones y en el cual se podrá hacer un informe por ubicación de los stocks de los componentes ubicados en la misma. El usuario se podrá apoyar en una pequeña aplicación de ubicaciones donde se mostrarán las diferentes zonas en un mapa correctamente codificadas.

- **Hacer kitting de un listado de componentes o de una estructura**

El usuario podrá extraer de forma masiva unidades de un listado previamente introducido en el formulario correspondiente indicando además una serie de datos importantes para el sistema. Esta misma

acción se podrá hacer introduciendo el componente raíz de una estructura, el cual formaría el principal componente dentro de un producto.

#### **2.3.1.3. USUARIO CON PERMISOS DE COMPRAS**

Tendrá las mismas funcionalidades que el usuario estándar y además:

- **Solicitud de compra**

Esta acción hace que el usuario rellene un formulario con los datos importantes para el sistema como pueden ser el proveedor, proyecto, orden de fabricación etc..y un desglose de todos los componentes referentes a ese proveedor eligiendo previamente para cada uno de ellos la oferta anteriormente introducida en el sistema. El sistema avisará de cualquier anomalía en unidades mínimas de pedido, tiempo mínimo de entrega y cambios de precio. El usuario podrá hacer un pdf con toda la información e incluso se podrán hacer solicitudes de compra parciales antes de ser definitivas para su posterior firma por parte del administrador.

- **Análisis de solicitudes de compra o pedidos**

Esta funcionalidad potencia la aplicación en el sentido que el usuario tendrá noticia en todo momento de cómo se encuentran sus solicitudes o pedidos (firmados, recibidos, inspeccionados, etc..).

- **Dar de alta un proveedor**

El usuario con estos permisos podrá dar de alta un proveedor con los respectivos datos relevantes para el sistema.

- **Dar de alta una oferta**

El usuario con estos permisos podrá dar de alta una oferta sobre algún componente buscado anteriormente y eligiendo sobre que fabricante es. Esto permite también un control sobre componentes defectuosos en alguna compra ya que estará fuertemente relacionado el fabricante con la oferta y la ubicación que se le dé a la hora de recibirlos.

- **Consulta de proveedores**

Esta funcionalidad potencia la aplicación en el sentido que el usuario tendrá noticia en todo momento de cómo se encuentran sus solicitudes o pedidos (firmados, recibidos, inspeccionados, etc..).

#### **2.3.1.4. USUARIO CON PERMISOS DE INGENIERÍA**

Tendrá las mismas funcionalidades que el usuario estándar y además:

- **Modificación o inserción de huellas**

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

Esta acción está restringida a usuarios con permisos de ingeniería donde se podrá, mediante una búsqueda previa, seleccionar un componente al que se desea modificar o insertar su huella y con posterioridad utilizar para su producción.

- **Insertar nuevo fabricante**

Esta acción está restringida a usuarios con permisos de ingeniería donde se podrá, mediante una búsqueda previa, seleccionar un componente al que se desea insertar un nuevo fabricante.

#### **2.3.1.5. USUARIO ADMINISTRADOR**

Este tipo de usuario no tiene unas funcionalidades concretas si no que son usuarios privilegiados para hacer todo lo posible para el buen funcionamiento del sistema. Su labor puede ser desde modificar, insertar o eliminar registros de cualquier tabla, firmar solicitudes de compra, dar de alta componentes, dar de alta órdenes de fabricación, tener permisos de almacén, tener permisos de compras, tener permisos de ingeniería hasta obtener información de cualquier dato del sistema.

#### **2.3.2. REQUISITOS DE INTERFACES EXTERNOS**

Se describirían los requisitos que afecten a la interfaz de usuario, interfaz con otros sistemas (hardware y software) e interfaces de comunicaciones.

##### **2.3.2.1. INTERFAZ DE USUARIO**

El acceso a la aplicación se realiza a través de la web mediante la intranet de la propia empresa. Por lo que para su utilización es necesario un navegador que permita conectarse al servidor y mostrar la interfaz de la aplicación sin problema alguno en los diferentes navegadores ya que el código se ciñe a los estándares. Dependiendo del nivel de privilegios o permisos que tenga el usuario la aplicación le mostrará todas las acciones que puede hacer pero cabe destacar que dicha aplicación está preparada para interactuar con ella solamente a través del navegador por ello la importancia de tener uno.

##### **2.3.2.2. INTERFACES HARDWARE**

Para acceder a la aplicación los usuarios deberán disponer de un ordenador con algunos requisitos importantes como un procesador potente para ejecutar cualquier navegador web, una tarjeta gráfica, de video y un monitor que pueda soportar una resolución de 800x600. Además tiene que tener una conexión de red ya sea Ethernet, ADSL, Wifi, módem debido a que la aplicación es vía Web.

#### **2.3.2.3. INTERFACES SOFTWARE**

Los usuarios podrán visualizar la aplicación en cualquier navegador actual sin la necesidad de usar un sistema operativo concreto. El servidor de la misma se basa en Apache con PHP y MySQL, disponibles para los sistemas Windows, BSD y Linux, éste último será el que usaremos.

#### **2.3.2.4. INTERFACES DE COMUNICACIONES**

Estas comunicaciones entre la aplicación y los clientes las realizaremos a través de la red interna de la empresa. Los usuarios actuarán trabajando desde la propia empresa a través de la red local. El mecanismo de comunicación se realiza a través de conexiones TCP/IP y el protocolo HTTP para comunicarse con el servidor de la aplicación.

#### **2.3.3. REQUISITOS DE EFICIENCIA**

La aplicación se ha realizado para el acceso simultáneo de varios usuarios a los servicios ofrecidos. Primero, el servidor web Apache donde encontramos la aplicación, proporciona acceso a un considerable nº de clientes, de fácil configuración desde el propio Apache y que permite visualizar el límite máximo de conexiones a la vez y del rendimiento que genera.

En cualquier caso, la aplicación asegura un acceso beneficioso en condiciones de carga del servidor normales. No es así en situación adversa, donde la aplicación podría errar y provocar problemas para navegar y por la cantidad de usuarios que accedan en corto espacio, viéndose el servidor saturado.

#### **2.3.4. OBLIGACIONES DE DISEÑO**

En esta sección se informará de todo aquello que restrinja las decisiones relativas al diseño de la aplicación: Restricciones de otros estándares, limitaciones del hardware, etc.

##### **2.3.4.1. ESTÁNDARES**

El diseño de la aplicación se ha realizado siguiendo los estándares fijados por el W3C. En este caso la aplicación se desarrolla mediante XHTML y CSS. He basado la elección del HTML en base a:

- Compatibilidad media con navegadores antiguos, es decir, visualizamos la información pero sin formato.
- Facilidad para editar directamente del código y mantenimiento.
- Utilización de formato abierto y compatible con los nuevos estándares desarrollados en W3C para futuros agentes y navegadores.

Con respecto al CSS se pretende separar la estructura de un documento de su presentación. He elegido el CSS por:

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

- Control centralizado de la presentación de un sitio web agilizando la actualización del mismo.
- En concreto, diversos navegadores permiten que los usuarios especifiquen su propia hoja de estilo aplicándolos a un sitio web de manera que, aumenta la accesibilidad.
- En una misma página se puede disponer de varias hojas de estilo según el dispositivo o la elección del usuario.
- El documento HTML reduce su tamaño y es de mejor comprensión.

#### **2.3.4.2. LIMITACIONES HARDWARE**

El sistema no requiere de un hardware especial extra, a excepción de un elemento para la conexión de red junto con la conexión a Internet. Los sistemas de los clientes deben permitir la ejecución de un navegador gráfico para acceder a la aplicación y la conexión a red, por lo demás pueden ser variables.

Como he comentado con anterioridad, podemos instalar un sistema basado en Linux con Apache, PHP y MySQL en sistemas 386, 486, Pentium,... Aplicado en un entorno real habría que ir mejorándolo cuánto mayores fuesen las necesidades del sistema.

#### **2.3.5. ATRIBUTOS**

##### **2.3.5.1. SEGURIDAD**

Llegamos a un punto básico para la aplicación. Ofrece diferentes funcionalidades supeditadas el usuario que acceda a la misma. Para identificar al cliente de forma fiable requerimos una cuenta con nombre de usuario y contraseña que asegure la validez del usuario en la aplicación. Dependiendo del usuario facilitaremos unas herramientas u otras.

La información acerca de las cuentas se guarda en la base de datos, con las contraseñas cifradas en MD5. En un futuro, la autenticación del usuario irá cifrada mediante el protocolo HTTPS, que conlleva el uso seguro de la aplicación durante su sesión.

##### **2.3.5.2. MANTENIMIENTO**

El mantenimiento de la aplicación se lleva a cabo por parte de los administradores de la misma, al igual que los cambios introducidos o modificaciones requeridas de la base de datos o funcionalidad, que se gestionarán por parte del administrador del sistema.

### 2.3.6. OTROS REQUISITOS

#### 2.3.6.1. BASE DE DATOS

La aplicación utiliza una base de datos en MySQL, donde se almacenará toda la información referida a componentes, ofertas, solicitudes de compra, pedidos, stock, kittings, etc.....Para poder consultarla se hará mediante el servidor web con PHP y la API de acceso de MySQL.

# 3 ANÁLISIS

En la fase de análisis se especificará qué tiene que hacer el programa que se va a desarrollar, tanto en lo que se refiere al comportamiento interno (gestión de los datos) como al externo (interacción con el usuario y con otras aplicaciones). Por eso esta sección se tratará con casos de uso que describen requisitos del sistema y cómo estos interactúan con el usuario, así como, los diagramas de secuencia, que muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo modelando cada caso de uso. Mientras que el caso de uso permite el modelado de una vista negocio del escenario, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y mensajes intercambiados entre los objetos. Por último se utilizará el diagrama UML para construir, documentar, visualizar y especificar el sistema de software de una forma gráfica.

## 3.1 CASOS DE USO

Explicaré con más detalle cada una de las funciones más importantes dentro de la aplicación. Adjunto un gráfico intuitivo (Figura 3.1) para una mejor comprensión y posteriormente los casos de uso con un nivel de detalle más minucioso.

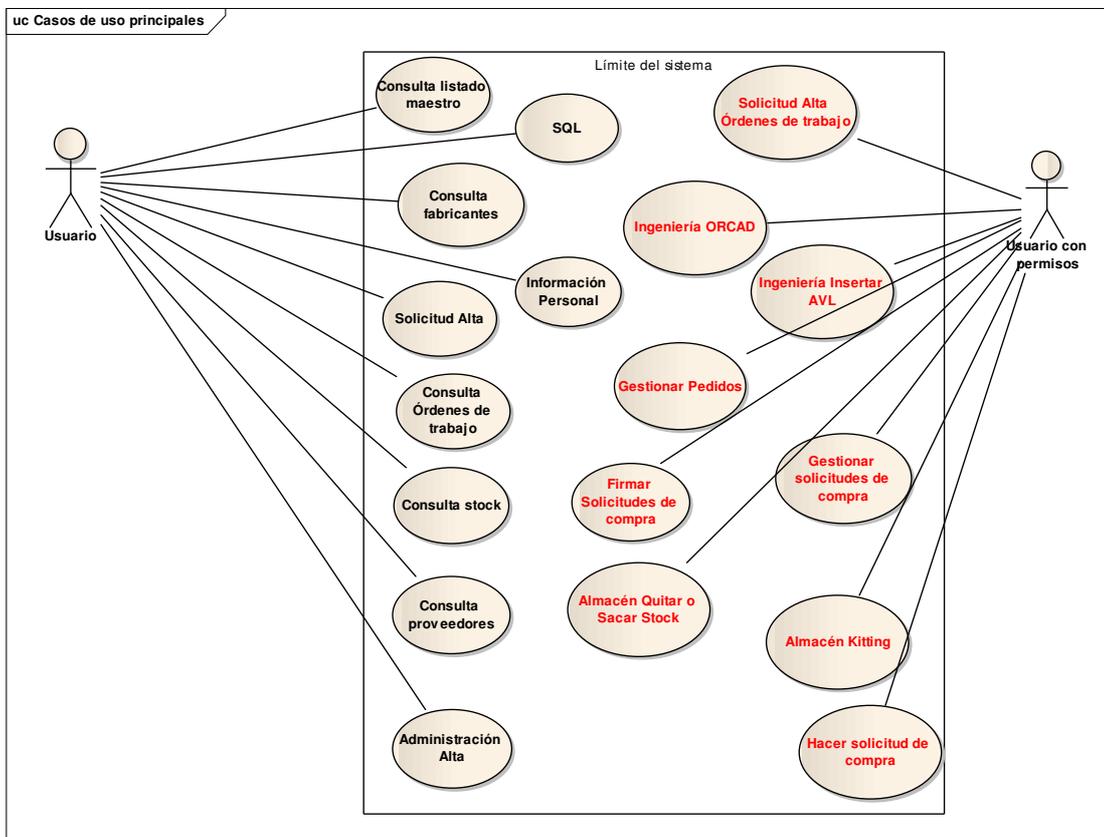


Figura 3.1 – Casos de Uso

<b>Caso de uso</b>	Búsqueda de información de un componente	
<b>Descripción</b>	Búsqueda de información sobre fabricantes, proveedores, datasheet, ubicaciones, stocks, ofertas, costes de planificación y donde se está usando	
<b>Actor iniciador</b>	Usuario	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	Un usuario accede a la aplicación para buscar información sobre un componente. Introduce la(s) palabra(s) a buscar en un formulario y ordena la búsqueda. El sistema debe buscar coincidencias de componentes con las palabras que el usuario ha introducido y devolverle el listado	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú INVENTARIADO y submenú LISTADO MAESTRO	
<b>Postcondiciones</b>	--	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. El usuario introduce los términos de la búsqueda en la caja de texto del formulario y pulsa el botón de "Buscar".</p> <p>5. El navegador recibe la página con los resultados de la búsqueda y se la muestra al usuario.</p>	<p>2. El navegador envía el contenido del formulario al servidor, para que este realice una búsqueda en la base de datos.</p> <p>3. El servidor web se conecta con la base de datos y realiza una consulta basándose en los datos proporcionados por el usuario.</p> <p>4. Con la respuesta de la base de datos, el servidor web construye una página con un listado de los componentes que coincidieron y la envía al navegador.</p>
<b>Extensiones síncronas</b>	Si en 3. La base de datos no devuelve ningún resultado se envía un mensaje informativo comunicando de la inexistencia de coincidencias y el usuario podrá intentarlo de nuevo.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Búsqueda de fabricantes de un componente	
<b>Descripción</b>	Búsqueda de información sobre componentes a partir de fabricantes. Se obtendrá información sobre proveedores, datasheet, ubicaciones, stocks, ofertas, costes de planificación y donde se está usando	
<b>Actor iniciador</b>	Usuario	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	Un usuario accede a la aplicación para buscar información sobre un componente a partir de fabricantes. Introduce la(s) palabra(s) a buscar en un formulario y ordena la búsqueda. El sistema debe buscar coincidencias de componentes con las palabras que el usuario ha introducido y devolverle el listado	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú INVENTARIADO y submenú CONSULTA FABRICANTES	
<b>Postcondiciones</b>	--	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. El usuario introduce los términos de la búsqueda en la caja de texto del formulario y pulsa el botón de "Buscar".</p> <p>5. El navegador recibe la página con los resultados de la búsqueda y se la muestra al usuario.</p>	<p>2. El navegador envía el contenido del formulario al servidor, para que este realice una búsqueda en la base de datos.</p> <p>3. El servidor web se conecta con la base de datos y realiza una consulta basándose en los datos proporcionados por el usuario.</p> <p>4. Con la respuesta de la base de datos, el servidor web construye una página con un listado de los componentes que coincidieron y la envía al navegador.</p>
<b>Extensiones síncronas</b>	Si en 3. La base de datos no devuelve ningún resultado se envía un mensaje informativo comunicando de la inexistencia de coincidencias y el usuario podrá intentarlo de nuevo.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Búsqueda de proveedores de un componente	
<b>Descripción</b>	Búsqueda de información sobre componentes a partir de proveedores. Se obtendrá información sobre fabricantes, datasheet, ubicaciones, stocks, ofertas, costes de planificación y donde se está usando	
<b>Actor iniciador</b>	Usuario	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	Un usuario accede a la aplicación para buscar información sobre un componente a partir de proveedores. Introduce la(s) palabra(s) a buscar en un formulario y ordena la búsqueda. El sistema debe buscar coincidencias de componentes con las palabras que el usuario ha introducido y devolverle el listado	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú INVENTARIADO y submenú CONSULTA PROVEEDORES	
<b>Postcondiciones</b>	--	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. El usuario introduce los términos de la búsqueda en la caja de texto del formulario y pulsa el botón de "Buscar".</p> <p>5. El navegador recibe la página con los resultados de la búsqueda y se la muestra al usuario.</p>	<p>2. El navegador envía el contenido del formulario al servidor, para que este realice una búsqueda en la base de datos.</p> <p>3. El servidor web se conecta con la base de datos y realiza una consulta basándose en los datos proporcionados por el usuario.</p> <p>4. Con la respuesta de la base de datos, el servidor web construye una página con un listado de los componentes que coincidieron y la envía al navegador.</p>
<b>Extensiones síncronas</b>	Si en 3. La base de datos no devuelve ningún resultado se envía un mensaje informativo comunicando de la inexistencia de coincidencias y el usuario podrá intentarlo de nuevo.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Búsqueda de información sobre stock de un componente	
<b>Descripción</b>	Búsqueda de información sobre el stock de los componentes a partir de una lista de componentes y stock que quiere saber.	
<b>Actor iniciador</b>	Usuario	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	Un usuario accede a la aplicación para buscar información sobre el stock de varios componentes a partir de una lista. Introduce un listado de componentes y stock de los cuales se quiere obtener información y ordena la búsqueda. El sistema debe buscar coincidencias de componentes con el listado que el usuario ha introducido y devolvérselo cumplimentado.	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú INVENTARIADO y submenú CONSULTA STOCK	
<b>Postcondiciones</b>	--	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. El usuario introduce los términos de la búsqueda en la lista de texto del formulario y pulsa el botón de "Buscar".</p> <p>5. El navegador recibe la página con los resultados de la búsqueda y se la muestra al usuario.</p>	<p>2. El navegador envía el contenido del formulario al servidor, para que este realice una búsqueda en la base de datos.</p> <p>3. El servidor web se conecta con la base de datos y realiza una consulta basándose en los datos proporcionados por el usuario.</p> <p>4. Con la respuesta de la base de datos, el servidor web construye una página con un listado de los componentes y stock que coincidieron y la envía al navegador.</p>
<b>Extensiones síncronas</b>	Si en 3. La base de datos no devuelve ningún resultado se envía un mensaje informativo comunicando de la inexistencia de coincidencias y el usuario podrá intentarlo de nuevo.	
<b>Extensiones asíncronas</b>	--	

## Memoria del Proyecto Final de Carrera

Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

<b>Caso de uso</b>	Agregar componentes mediante un fichero	
<b>Descripción</b>	Agregar componentes desde un fichero en formato .csv y con los campos obligatorios.	
<b>Actor iniciador</b>	Usuario administrador	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	Un usuario administrador accede a la aplicación para insertar desde un fichero todos los componentes que tengan bien cumplimentados los datos obligatorios. El sistema debe buscar coincidencias de componentes con el listado que el usuario administrador ha introducido y no agregará componentes repetidos ni con los datos erróneos	
<b>Precondiciones</b>	El usuario administrador ha accedido a la aplicación, ha entrado en el menú INVENTARIADO y submenú ALTAS MASIVAS y tiene permisos de administrador.	
<b>Postcondiciones</b>	Los nuevos componentes quedan registrados en el sistema	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. El usuario introduce el fichero con los componentes en campo del formulario y pulsa el botón de "Insertar".</p> <p>4. El navegador recibe el mensaje de información errónea o no.</p>	<p>2. El navegador envía el contenido del formulario al servidor, para que este realice una búsqueda en la base de datos.</p> <p>3. El servidor web se conecta con la base de datos y realiza una inserción de cada uno de los componentes basándose en los datos proporcionados por el usuario y que no sean repetidos o erróneos. Cada uno de ellos tendrán un ID propio.</p>
<b>Extensiones síncronas</b>	Si en 4. En la base de datos existe registros repetidos o erróneos se envía un mensaje informativo comunicando de las líneas que han producido el fallo o los errores que tiene el fichero al navegador y el usuario podrá intentarlo de nuevo.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Hacer informes con consultas SQL	
<b>Descripción</b>	Se hacen todo tipo de informes mediante consultas SQL y será posible la exportación de los mismos en Excel	
<b>Actor iniciador</b>	Usuario	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	Un usuario accede a la aplicación para hacer sus propios informes a partir de consultas SQL para posteriormente exportarlos en EXCEL. El sistema debe buscar coincidencias con el tipo de consulta que se introdujo y devolver la información en un listado.	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú SQL	
<b>Postcondiciones</b>	--	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. El usuario introduce la consulta para buscar la información deseada y pulsa el botón de "SQL".</p> <p>5. El navegador recibe la página con los resultados de la búsqueda y se la muestra al usuario.</p> <p>6. Si el usuario quisiera exportarlo en formato EXCEL le daría al botón "EXCEL".</p> <p>8. El usuario acepta el archivo.</p>	<p>2. El navegador envía el contenido del formulario al servidor, para que este realice una búsqueda en la base de datos.</p> <p>3. El servidor web se conecta con la base de datos y realiza una consulta basándose en la consulta proporcionada por el usuario.</p> <p>4. Con la respuesta de la base de datos, el servidor web construye una página con un listado y la envía al navegador.</p> <p>7. Con la respuesta el servidor web construye un archivo EXCEL y la envía al navegador.</p> <p>9. Con la respuesta el servidor web permite el acceso al fichero.</p>
<b>Extensiones síncronas</b>	Si en 8. El usuario cancelase su descarga no ocurriría el punto 9.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Modificar o mostrar la información personal.	
<b>Descripción</b>	Modificar u obtener información sobre los datos personales dentro de la aplicación.	
<b>Actor iniciador</b>	Usuario	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá obtener y modificar información personal útil para la aplicación.	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú INFORMACIÓN PERSONAL.	
<b>Postcondiciones</b>	Los cambios de información quedan registrados en el sistema	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>3. El usuario comprueba los datos y modifica lo que desea. Posteriormente pulsa el botón "Modificar"</p> <p>5. El navegador recibe la página con los resultados de la modificación y se la muestra al usuario.</p>	<p>1. El servidor web se conecta con la base de datos y realiza una consulta basándose en el usuario que ha accedido.</p> <p>2. Con la respuesta de la base de datos, el servidor web construye una página con los datos y la envía al navegador.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y registra los cambios en la base de datos. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	En 3 el usuario puede salir sin modificar ningún dato.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Mostrar o modificar la información sobre una huella de un componente	
<b>Descripción</b>	Modificar u obtener información sobre los datos de una huella de un componente.	
<b>Actor iniciador</b>	Usuario administrador con permiso de Ingeniería	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá obtener y modificar información sobre los datos de una huella de un componente	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú INGENIERÍA y submenú ORCAD. Tendrá que tener el permiso de Ingeniería	
<b>Postcondiciones</b>	Los cambios de información quedan registrados en el sistema	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario selecciona un componente que quiere modificar o mostrar mediante una búsqueda relativa introduciendo para ello información de búsqueda.</p> <p>3. El usuario modifica los datos y presiona el botón "Modificar".</p> <p>5. El navegador recibe la página con los resultados de la modificación y se la muestra al usuario.</p>	<p>2. El sistema recoge los datos de la base de datos y genera un formulario con la información.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y registra los cambios en la base de datos. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	En 3 el usuario puede salir sin modificar ningún dato.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Agregar una huella nueva de un componente	
<b>Descripción</b>	Agregar una huella nueva de un componente.	
<b>Actor iniciador</b>	Usuario administrador con permiso de Ingeniería	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá agregar una nueva huella a un componente	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú INGENIERÍA y submenú ORCAD. Tendrá que tener el permiso de Ingeniería	
<b>Postcondiciones</b>	Los cambios de información quedan registrados en el sistema	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario selecciona un componente al que quiere agregar una nueva huella mediante una búsqueda relativa introduciendo para ello información de búsqueda.</p> <p>3. El usuario introduce los datos y presiona el botón "Introducir".</p> <p>5. El navegador recibe la página con la nueva alta y se la muestra al usuario.</p>	<p>2. El sistema recoge los datos de la base de datos y genera un formulario con la información.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y agrega el registro en la base de datos con un ID propio. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	En 3 el usuario puede salir sin introducir ninguna huella a ese componente.	
<b>Extensiones asíncronas</b>	--	

## Memoria del Proyecto Final de Carrera

Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

<b>Caso de uso</b>	Mostrar o modificar la información sobre un fabricante de un componente	
<b>Descripción</b>	Modificar u obtener información sobre los datos de un fabricante de un componente.	
<b>Actor iniciador</b>	Usuario administrador con permiso de Ingeniería	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá obtener y modificar información sobre los datos un fabricante de un componente	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú INGENIERÍA y submenú AVL. Tendrá que tener el permiso de Ingeniería	
<b>Postcondiciones</b>	Los cambios de información quedan registrados en el sistema	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario selecciona un componente que quiere modificar o mostrar mediante una búsqueda relativa introduciendo para ello información de búsqueda.</p> <p>3. El usuario modifica los datos y presiona el botón “Modificar”.</p> <p>5. El navegador recibe la página con los resultados de la modificación y se la muestra al usuario.</p>	<p>2. El sistema recoge los datos de la base de datos y genera un formulario con la información.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y registra los cambios en la base de datos. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	En 3 el usuario puede salir sin modificar ningún dato.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Agregar un fabricante nuevo a un componente	
<b>Descripción</b>	Agregar un fabricante nuevo a un componente	
<b>Actor iniciador</b>	Usuario administrador con permiso de Ingeniería	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá agregar un nuevo fabricante a un componente	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú INGENIERÍA y submenú AVL. Tendrá que tener el permiso de Ingeniería	
<b>Postcondiciones</b>	Los cambios de información quedan registrados en el sistema	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario selecciona un componente al que quiere agregar un nuevo fabricante mediante una búsqueda relativa introduciendo para ello información de búsqueda.</p> <p>3. El usuario introduce los datos y presiona el botón "Introducir".</p> <p>5. El navegador recibe la página con la nueva alta y se la muestra al usuario.</p>	<p>2. El sistema recoge los datos de la base de datos y genera un formulario con la información.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y agrega el registro en la base de datos con un ID propio. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	En 3 el usuario puede salir sin introducir ningún fabricante.	
<b>Extensiones asíncronas</b>	--	



<b>Caso de uso</b>	Modificar la información sobre una orden de trabajo	
<b>Descripción</b>	Modificar información sobre los datos de una orden de trabajo	
<b>Actor iniciador</b>	Usuario administrador con permiso de Producción	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá modificar información sobre los datos de una orden de trabajo	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú PRODUCCIÓN y submenú ÓRDENES DE TRABAJO. Tendrá que tener el permiso de Producción	
<b>Postcondiciones</b>	Los cambios de información quedan registrados en el sistema	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario selecciona una orden de trabajo que quiere modificar.</p> <p>3. El usuario modifica los datos y presiona el botón "Modificar".</p> <p>5. El navegador recibe la página con los resultados de la modificación y se la muestra al usuario.</p>	<p>2. El sistema recoge los datos de la base de datos y genera un formulario con la información.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y registra los cambios en la base de datos. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	En 3 el usuario puede salir sin modificar ningún dato.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Agregar una nueva orden de fabricación	
<b>Descripción</b>	Agregar una nueva orden de fabricación	
<b>Actor iniciador</b>	Usuario administrador con permiso de Producción	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá agregar una nueva orden de fabricación	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú PRODUCCIÓN y submenú SOLICITUD ALTA. Tendrá que tener el permiso de Producción	
<b>Postcondiciones</b>	La información queda registrada en el sistema	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario presiona el botón "SOLICITUD ALTA".</p> <p>3. El usuario introduce los datos y presiona el botón "Introducir".</p> <p>5. El navegador recibe la página con la nueva alta y se la muestra al usuario.</p>	<p>2. El sistema recoge los datos mediante un formulario.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y agrega el registro en la base de datos con un ID propio. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	<p>En 3 no se rellena los campos obligatorios o no tienen el formato correcto, el sistema informa del error y vuelve a solicitarlos.</p> <p>En 3 puede cancelar el alta.</p>	
<b>Extensiones asíncronas</b>	--	

## Memoria del Proyecto Final de Carrera

Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

<b>Caso de uso</b>	Agregar un nuevo componente al sistema.	
<b>Descripción</b>	Agregar un nuevo componente al sistema a expensas de que un administrador lo confirme para darlo de alta definitivamente.	
<b>Actor iniciador</b>	Usuario	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá agregar un nuevo componente en el sistema de forma parcial hasta que un administrador corrobore los datos y se ingrese en el sistema de manera definitiva	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú INVENTARIADO y submenú SOLICITUD ALTA.	
<b>Postcondiciones</b>	La información queda registrada en el sistema.	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario presiona el botón "SOLICITUD ALTA".</p> <p>3. El usuario introduce los datos y presiona el botón "Introducir".</p> <p>5. El navegador recibe la página con la nueva alta y se la muestra al usuario.</p>	<p>2. El sistema recoge los datos mediante un formulario.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y agrega el registro en la base de datos con un ID propio. Posteriormente envía la información al navegador y envía un mensaje de correo para el administrador.</p>
<b>Extensiones síncronas</b>	<p>En 3 no se rellena los campos obligatorios o no tienen el formato correcto, el sistema informa del error y vuelve a solicitarlos.</p> <p>En 3 puede cancelar el alta.</p>	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Modificar la información sobre una alta parcial de un componente	
<b>Descripción</b>	Modificar la información sobre una alta parcial de un componente	
<b>Actor iniciador</b>	Usuario	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá modificar información sobre un alta parcial de un componente.	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú INVENTARIADO y submenú ALTAS PROVISIONALES.	
<b>Postcondiciones</b>	Los cambios de información quedan registrados en el sistema	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario selecciona un componente que quiere modificar de los que había dado de alta parcialmente.</p> <p>3. El usuario modifica los datos y presiona el botón "Modificar".</p> <p>5. El navegador recibe la página con los resultados de la modificación y se la muestra al usuario.</p>	<p>2. El sistema recoge los datos de la base de datos y genera un formulario con la información.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y registra los cambios en la base de datos. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	<p>En 3 el usuario puede salir sin modificar ningún dato.</p> <p>En 5 el usuario puede reenviar el componente parcial de nuevo para ser evaluado de nuevo por los administradores. En este caso iría acompañado de un mensaje de reenvío.</p>	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Eliminar la información sobre una alta parcial de un componente	
<b>Descripción</b>	Eliminar la información sobre una alta parcial de un componente	
<b>Actor iniciador</b>	Usuario	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá Eliminar información sobre un alta parcial de un componente.	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú INVENTARIADO y submenú ALTAS PROVISIONALES.	
<b>Postcondiciones</b>	Los cambios de información quedan registrados en el sistema	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario selecciona un componente que quiere eliminar de los que había dado de alta parcialmente.</p> <p>3. El usuario selecciona el componente y presiona el botón "Eliminar".</p> <p>5. El navegador recibe la página con los resultados de la eliminación y se la muestra al usuario.</p>	<p>2. El sistema recoge los datos de la base de datos y genera un formulario con la información.</p> <p>4. El sistema lo elimina y registra los cambios en la base de datos. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	En 3 el usuario puede salir sin eliminar ningún dato.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Agregar un nuevo componente al sistema.	
<b>Descripción</b>	Agregar un nuevo componente al sistema revisando algún componente dado de alta provisional.	
<b>Actor iniciador</b>	Usuario administrador	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá agregar un nuevo componente en el sistema cuando compruebe que todos los datos parciales son correctos	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú HERRAMIENTAS DW y submenú ALTAS ADMINISTRADOR.	
<b>Postcondiciones</b>	La información queda registrada en el sistema.	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario presiona el botón "ALTAS ADMINISTRADOR".</p> <p>3. El usuario comprueba los datos introducidos anteriormente por el usuario que solicitó el alta y presiona el botón "Introducir".</p> <p>5. El navegador recibe la página con la nueva alta y se la muestra al usuario.</p>	<p>2. El sistema recoge los datos mediante un formulario.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y agrega el registro en la base de datos con un ID propio. Posteriormente envía la información al navegador y envía un mensaje de correo para el solicitante para que sepa que ha sido correcta y otro correo informativo para los demás usuarios con el ID definitivo.</p>
<b>Extensiones síncronas</b>	<p>En 3 no se rellena los campos obligatorios o no tienen el formato correcto, el sistema informa del error y vuelve a solicitarlos.</p> <p>En 3 puede cancelar o rechazar el alta. En este último caso se envía un correo al solicitante para que sepa el motivo y pueda subsanarlo.</p>	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Agregar o quitar stock a un componente del sistema.	
<b>Descripción</b>	Agregar o quitar stock a un componente del sistema. Esta acción se puede hacer introduciendo previamente el ID del componente o por la ubicación donde se encuentre.	
<b>Actor iniciador</b>	Usuario con permisos de ALMACÉN	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá agregar o quitar stock de un componente	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú ALMACÉN y submenú OPERACIONES POR PART NUMBER.	
<b>Postcondiciones</b>	La información queda registrada en el sistema.	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario presiona el botón "OPERACIONES POR PART NUMBER". Selecciona un componente al que quiere agregar o quitar stock mediante una búsqueda relativa introduciendo para ello información de búsqueda.</p> <p>3. El usuario agrega o extrae las unidades del componente que quiera y presiona el botón correspondiente.</p>	<p>2 El sistema recoge los datos de la base de datos y genera un formulario con la información</p> <p>4. El sistema comprueba que no existen datos erróneos y cambia el registro en la base de datos. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	<p>En 3 no se rellena los campos obligatorios o no tienen el formato correcto, el sistema informa del error y vuelve a solicitarlos.</p> <p>En 3 puede cancelar la acción.</p>	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Agregar o quitar stock a uno o varios componentes del sistema mediante un kitting.	
<b>Descripción</b>	Agregar o quitar stock a uno o varios componentes del sistema. El usuario lo podrá hacer mediante un listado donde podrá elegir desde que ubicaciones se extraen las unidades. Ésta acción se podrá hacer de forma parcial o completa. Ésta acción se podrá también hacer con un listado de componentes o un ID de un componente del cual cuelga una estructura entera de componentes que forman un producto.	
<b>Actor iniciador</b>	Usuario con permisos de ALMACÉN	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	Agregar o quitar stock a uno o varios componentes del sistema mediante un kitting.	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú ALMACÉN y submenú KITTING NUEVO.	
<b>Postcondiciones</b>	La información queda registrada en el sistema.	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario rellena el formulario donde deposita la lista de componentes y las unidades que quiere extraer. Seguidamente presiona el botón "KITTING"</p> <p>3. El usuario extrae las unidades de las ubicaciones que desee de cada uno de los componentes que introdujo en la lista y presiona el botón "EJECUTAR KITTING".</p>	<p>2 El sistema recoge los datos de la base de datos y genera un formulario con la información.</p> <p>4. El sistema comprueba que no existen datos erróneos y cambia el registro en la base de datos. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	<p>En 3 no se rellena los campos obligatorios o no tienen el formato correcto, el sistema informa del error y vuelve a solicitarlos.</p> <p>En 3 puede cancelar la acción.</p>	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Agregar una solicitud de compra.	
<b>Descripción</b>	El usuario podrá solicitar la compra de diferentes componentes eligiendo entre diversas referencias y proveedores.	
<b>Actor iniciador</b>	Usuario con permisos de COMPRAS	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá solicitar la compra de diferentes componentes eligiendo entre diversas referencias y proveedores.	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú COMPRAS y submenú SOLICITUD DE COMPRA.	
<b>Postcondiciones</b>	La información queda registrada en el sistema.	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario rellena el formulario donde primero rellena los datos de la solicitud y luego deposita la lista de componentes y la elección de las referencias. Seguidamente presiona el botón "SOLICITUD COMPRA".</p> <p>3. El usuario acepta el archivo que se construye con la información introducida por el usuario.</p>	<p>2 El sistema recoge los datos de la base de datos y genera un formulario con la información.</p> <p>4. El sistema comprueba que no existen datos erróneos y cambia el registro en la base de datos. Posteriormente envía la información al navegador y la muestra en pantalla en formato pdf.</p>
<b>Extensiones síncronas</b>	<p>En 1 no se rellena los campos obligatorios o no tienen el formato correcto, el sistema informa del error y vuelve a solicitarlos.</p> <p>En 3 puede cancelar la acción.</p>	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Mostrar o modificar la información sobre las solicitudes de compra.	
<b>Descripción</b>	Modificar u obtener información sobre las solicitudes de compra.	
<b>Actor iniciador</b>	Usuario con permisos de COMPRAS	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá obtener y modificar información sobre las solicitudes de compra.	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú COMPRAS y submenú SEGUIMIENTO SOLICITUDES. Tendrá que tener el permiso de Compras	
<b>Postcondiciones</b>	Los cambios de información quedan registrados en el sistema	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario selecciona una solicitud de compra que quiere modificar o mostrar mediante una búsqueda relativa introduciendo para ello información de búsqueda.</p> <p>3. El usuario modifica los datos y presiona el botón "Modificar".</p>	<p>2. El sistema recoge los datos de la base de datos y genera un formulario con la información.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y registra los cambios en la base de datos. Posteriormente envía la información al navegador y la muestra en pantalla en formato pdf.</p>
<b>Extensiones síncronas</b>	En 3 el usuario puede salir sin firmar ninguna solicitud.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Firmar las solicitudes de compra.	
<b>Descripción</b>	Firmar las solicitudes de compra. Acción requerida para su posterior pedido.	
<b>Actor iniciador</b>	Usuario administrador con permisos de COMPRAS	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario después de revisar la correcta solicitud de compra podrá firmarla para crear el pedido y así gestionarlo.	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú COMPRAS y submenú FIRMAS SOLICITUDES. Tendrá que tener el permiso de Compras	
<b>Postcondiciones</b>	Los cambios de información quedan registrados en el sistema	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario selecciona una solicitud de compra que quiere firmar mediante una búsqueda relativa introduciendo para ello información de búsqueda.</p> <p>3. El usuario pone la fecha de firma.</p> <p>5. El navegador recibe la página con la solicitud de compra convertida en pedido y se muestra al usuario.</p>	<p>2. El sistema recoge los datos de la base de datos y genera un formulario con la información.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y registra los cambios en la base de datos. Posteriormente envía la información al navegador y la muestra en pantalla en formato pdf. En ese instante el sistema hace que la solicitud de compra se convierta en pedido cambiándole el estado.</p>
<b>Extensiones síncronas</b>	En 3 el usuario puede salir sin modificar ningún dato.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Mostrar la información sobre los pedidos.	
<b>Descripción</b>	Obtener información sobre los pedidos.	
<b>Actor iniciador</b>	Usuario con permisos de COMPRAS	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá obtener información sobre los pedidos.	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú COMPRAS y submenú SEGUIMIENTO PEDIDOS. Tendrá que tener el permiso de Compras	
<b>Postcondiciones</b>	--	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario selecciona un pedido que quiere mostrar mediante una búsqueda relativa introduciendo para ello información de búsqueda.</p> <p>5. El navegador recibe la página con los resultados de la búsqueda y se la muestra al usuario.</p>	<p>2. El navegador envía el contenido del formulario al servidor, para que este realice una búsqueda en la base de datos.</p> <p>3. El servidor web se conecta con la base de datos y realiza una consulta basándose en los datos proporcionados por el usuario.</p> <p>4. Con la respuesta de la base de datos, el servidor web construye una página con un listado de los pedidos que coincidieron y la envía al navegador.</p>
<b>Extensiones síncronas</b>	Si en 3. La base de datos no devuelve ningún resultado se envía un mensaje informativo comunicando de la inexistencia de coincidencias y el usuario podrá intentarlo de nuevo.	
<b>Extensiones asíncronas</b>		

<b>Caso de uso</b>	Mostrar la información sobre proveedores.	
<b>Descripción</b>	Obtener información sobre proveedores.	
<b>Actor iniciador</b>	Usuario con permisos de COMPRAS	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá obtener información sobre los pedidos.	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú COMPRAS y submenú CONSULTA PROVEEDORES. Tendrá que tener el permiso de Compras	
<b>Postcondiciones</b>	--	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario selecciona proveedores que quiere mostrar mediante una búsqueda relativa introduciendo para ello información de búsqueda.</p> <p>5. El navegador recibe la página con los resultados de la búsqueda y se la muestra al usuario.</p>	<p>2. El navegador envía el contenido del formulario al servidor, para que este realice una búsqueda en la base de datos.</p> <p>3. El servidor web se conecta con la base de datos y realiza una consulta basándose en los datos proporcionados por el usuario.</p> <p>4. Con la respuesta de la base de datos, el servidor web construye una página con un listado de proveedores que coincidieron y la envía al navegador.</p>
<b>Extensiones síncronas</b>	Si en 3. La base de datos no devuelve ningún resultado se envía un mensaje informativo comunicando de la inexistencia de coincidencias y el usuario podrá intentarlo de nuevo.	
<b>Extensiones asíncronas</b>		

<b>Caso de uso</b>	Agregar un nuevo proveedor.	
<b>Descripción</b>	Agregar un nuevo proveedor.	
<b>Actor iniciador</b>	Usuario administrador con permisos de COMPRAS	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá agregar un nuevo proveedor en el sistema cuando compruebe que todos los datos son correctos	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú COMPRAS y submenú ALTA PROVEEDOR.	
<b>Postcondiciones</b>	La información queda registrada en el sistema.	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario presiona el botón "ALTA PROVEEDOR".</p> <p>3. El usuario introduce todos los datos obligatorios y presiona el botón "Introducir".</p> <p>5. El navegador recibe la página con la nueva alta y se la muestra al usuario.</p>	<p>2. El sistema recoge los datos mediante un formulario.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y agrega el registro en la base de datos con un ID propio. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	En 3 no se rellena los campos obligatorios o no tienen el formato correcto, el sistema informa del error y vuelve a solicitarlos.	
<b>Extensiones asíncronas</b>	--	

<b>Caso de uso</b>	Agregar una nueva oferta.	
<b>Descripción</b>	Agregar una nueva oferta.	
<b>Actor iniciador</b>	Usuario administrador con permisos de COMPRAS	
<b>Actores secundarios</b>	--	
<b>Resumen</b>	El usuario podrá agregar una nueva oferta en el sistema cuando compruebe que todos los datos son correctos	
<b>Precondiciones</b>	El usuario ha accedido a la aplicación y ha entrado en el menú COMPRAS y submenú ALTA OFERTA.	
<b>Postcondiciones</b>	La información queda registrada en el sistema.	
<b>Flujos de eventos</b>	<b>Interacciones del usuario</b>	<b>Obligaciones del sistema</b>
	<p>1. Se inicia cuando el usuario presiona el botón "ALTA OFERTA".</p> <p>3. El usuario introduce todos los datos obligatorios y presiona el botón "Introducir".</p> <p>5. El navegador recibe la página con la nueva alta y se la muestra al usuario.</p>	<p>2. El sistema recoge los datos mediante un formulario.</p> <p>4. El sistema comprueba que no existen repeticiones ni datos erróneos y agrega el registro en la base de datos con un ID propio. Posteriormente envía la información al navegador.</p>
<b>Extensiones síncronas</b>	En 3 no se rellena los campos obligatorios o no tienen el formato correcto, el sistema informa del error y vuelve a solicitarlos.	
<b>Extensiones asíncronas</b>	--	

## 3.2 DIAGRAMAS DE SECUENCIA

En este apartado informaré con diversos diagramas de secuencia como actúa la aplicación y qué pasos ejecuta para ciertas interacciones del usuario.

En general se va a tratar en este apartado los diagramas más frecuentes dentro del sistema como son las consultas para que un usuario obtenga la información deseada de cualquier dato del sistema según sus privilegios dentro de la aplicación y las inserciones para agregar nuevos registros al sistema.

En este caso he elegido una búsqueda de las múltiples que se pueden hacer en el sistema como es la búsqueda de un componente y una inserción de datos como es el alta de un usuario.

Como se observa en la Figura 3.2 para cualquier búsqueda dentro del sistema se establece una conexión hacia la base de datos para comprobar que los datos insertados por el usuario en la interfaz gráfica, en este caso el formulario pertinente, son correctos y poder así, ofrecer al usuario el resultado de la misma forma que introdujo los datos. Básicamente todas las consultas que se quieran hacer por parte de cualquier usuario en el sistema se harán de la forma indicada anteriormente. El sistema, por su parte, las resolverá siguiendo unas directrices como son la conexión a la base de datos, comprobación de errores y por último búsqueda de la información solicitada.

En la figura 3.3 observamos la inserción de un nuevo usuario dentro del sistema pero se puede ampliar a cualquier inserción dentro del sistema. En este caso, en el formulario correspondiente, el usuario pertinente según sus privilegios seleccionará la opción dar de alta un usuario nuevo. Posteriormente introducirá los datos necesarios y el sistema se encargará de conectar con la base de datos para la verificación de los mismos. En caso erróneo el sistema le mostrará los fallos ocurridos y la posible solución para subsanarlos y, en caso contrario, mostrará un mensaje confirmando la operación realizada y los nuevos registros introducidos. En general el usuario actuará de forma similar en todas las posibles inserciones que se quieran realizar dentro del sistema teniendo en cuenta las posibles restricciones. El sistema, en este caso, resolverá de la misma forma que se ha resuelto el caso que se muestra en la figura 3.3, primero pedirá los datos necesarios para la inserción, luego comprobará que son correctos y posteriormente informará al usuario si ha sido posible la inserción y en caso contrario intentar ofrecerle una solución vía mensaje.

✓ **Búsqueda de un componente dentro del sistema**

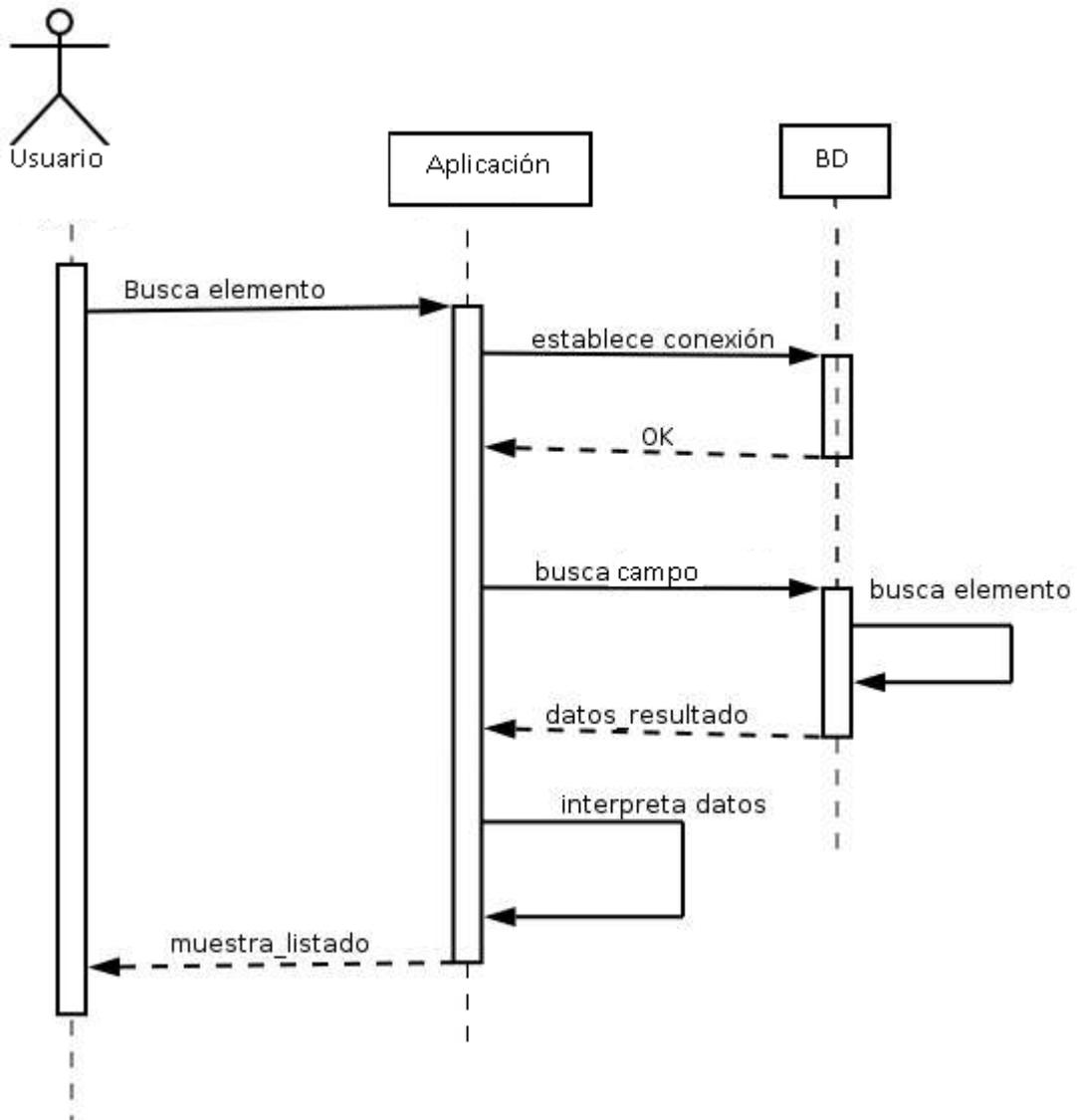


Figura 3.2 – Búsqueda de un componente dentro del sistema

- ✓ Insertar un nuevo usuario dentro del sistema por parte de un administrador

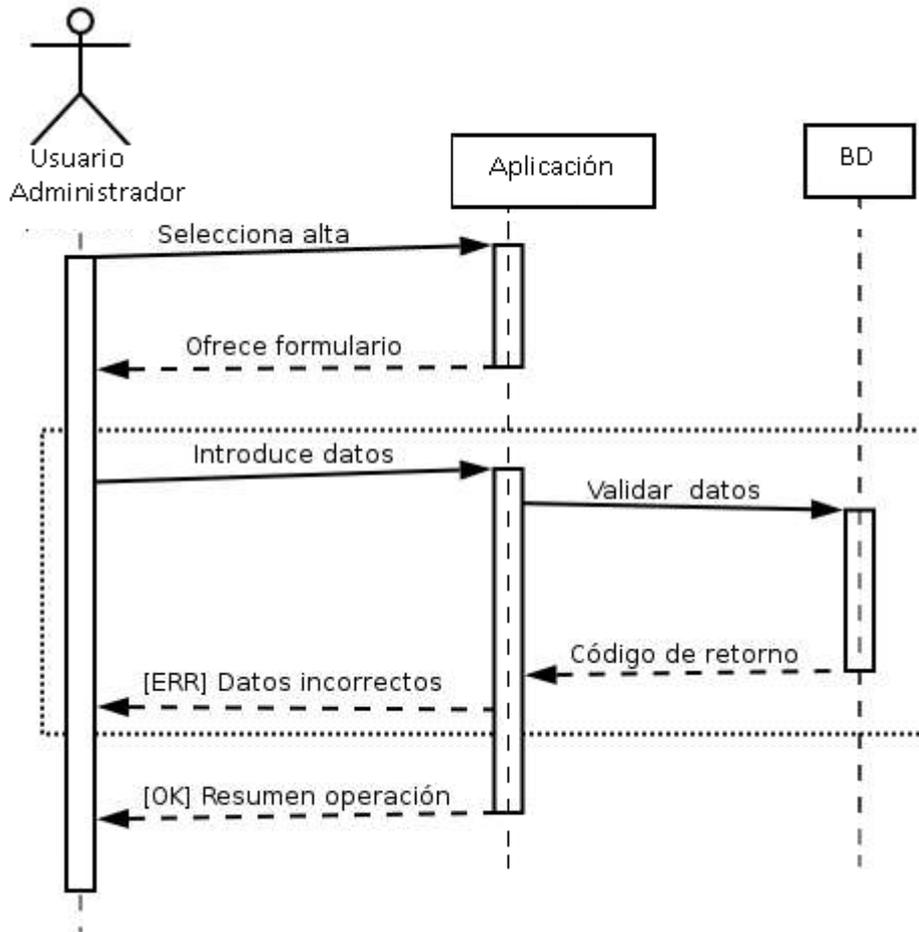


Figura 3.3 – Insertar un nuevo usuario dentro del sistema

## 3.3 DIAGRAMA UML

El diagrama UML de la Figura 3.4 ayuda a entender de una forma más fácil las diferentes funciones que se pueden realizar, que tipo de usuario las puede hacer, como se pueden hacer y qué tipos de elementos están relacionados entre sí y con qué restricciones lo están.

Principalmente, hay una clase de elementos fundamentales alrededor de los cuales se realizan las principales funciones de la aplicación. Éstos son itemmaster, ubica, avl, ofertas, usuarios, órdenes de fabricación y proveedores. Serán los encargados de interactuar con todos los demás elementos de la aplicación.

Ahora explicaré cada uno de los elementos de la aplicación con sus posibles acciones sobre ellos y relaciones con los demás elementos.

**Usuarios** identifica a cada uno de los miembros que podrán utilizar la aplicación. Según los permisos que tenga pertenecientes a cada uno de los campos establecidos en la tabla podrá hacer operaciones como dar de alta componentes como órdenes de trabajo, administrar tablas, modificar o eliminar elementos u órdenes de trabajo, administrar los proveedores, quitar o extraer stock, hacer kitting o hacer solicitudes de compra.

**Itemmaster, AVL, Ubica** identifica cada uno de las diferentes informaciones que se necesitan sobre un componente dentro del sistema. Todos estos elementos están relacionados entre sí por una clave primaria denominada Part Number.

**Ofertas, proveedores** identifica la oferta con un proveedor mediante una relación la cual está asociada a un fabricante de la tabla AVL. Sólo se podrán dar de alta si el usuario tiene el permiso pertinente.

**Órdenes de fabricación** identifica la información que se necesita para cada una de las tareas de fabricación. Está relacionado con la tabla de solicitudes que según su campo estado pasarían a ser pedidos y los cuales tendrán una referencia a su orden de fabricación. Según los privilegios del usuario se podrán cambiar el estado de la orden, dar de alta o administrar la tabla.

**Huellas** identifica la información necesaria para producción y estará relacionada con la tabla itemmaster ya que tiene que existir el componente para poder tener huella. Según los permisos del usuario podrá dar de alta o administrar la tabla que las contiene.

**Kitting y líneas de kitting** contendrán toda la información necesaria sobre el kitting y los componentes a los cuales se les extraerán unidades del stock. La relación de estas dos tablas estará compuesta por el identificador del kitting y el identificador del componente.

**Solicitud compra y líneas solicitud compras** contendrán toda la información necesaria sobre la solicitud de compra y los componentes. La relación de estas dos tablas estará compuesta por el identificador de la solicitud de compra y el identificador de cada uno de los componentes.

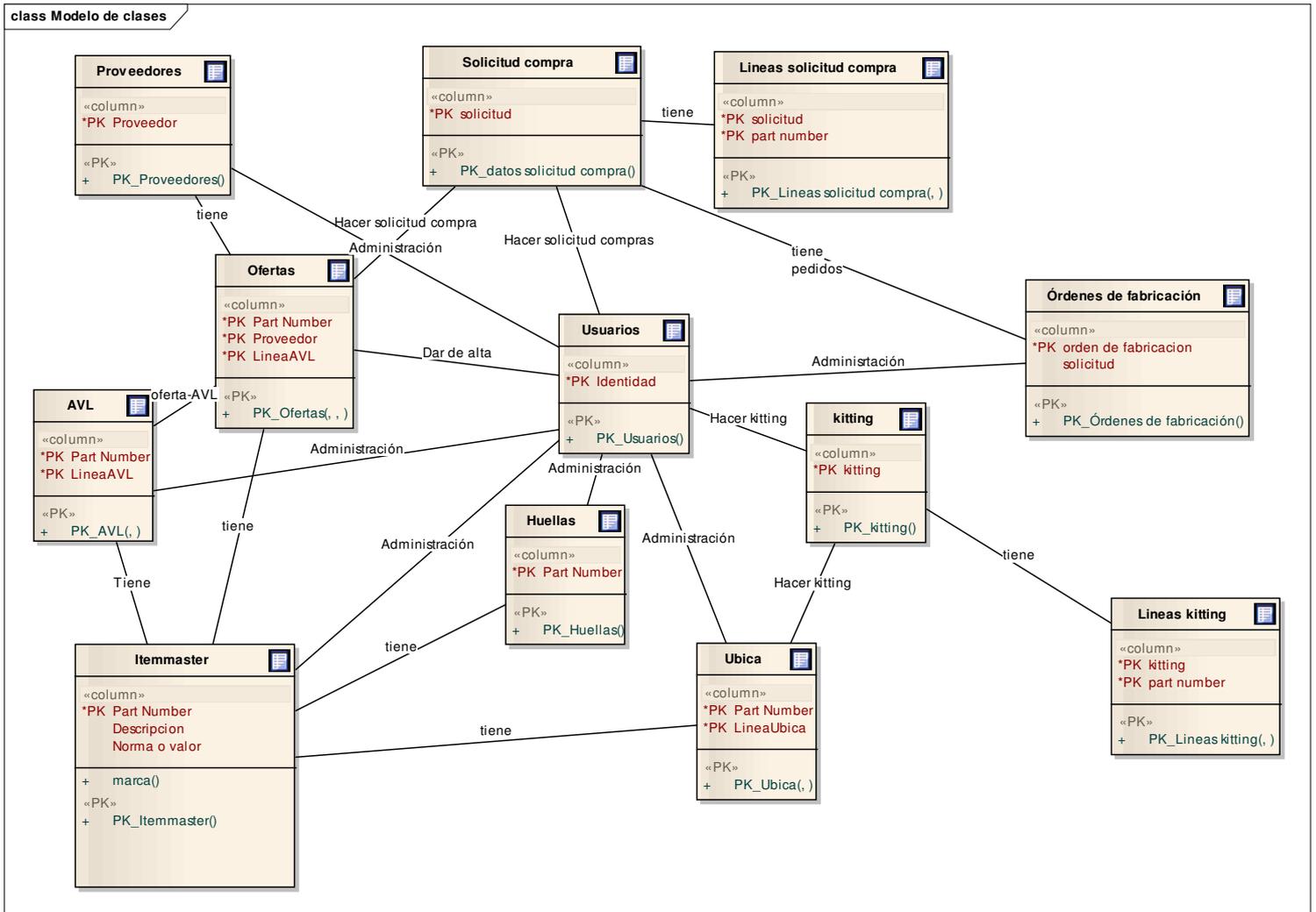


Figura 3.4 – Diagrama UML

### 3.4 DISEÑO ENTIDAD-RELACIÓN

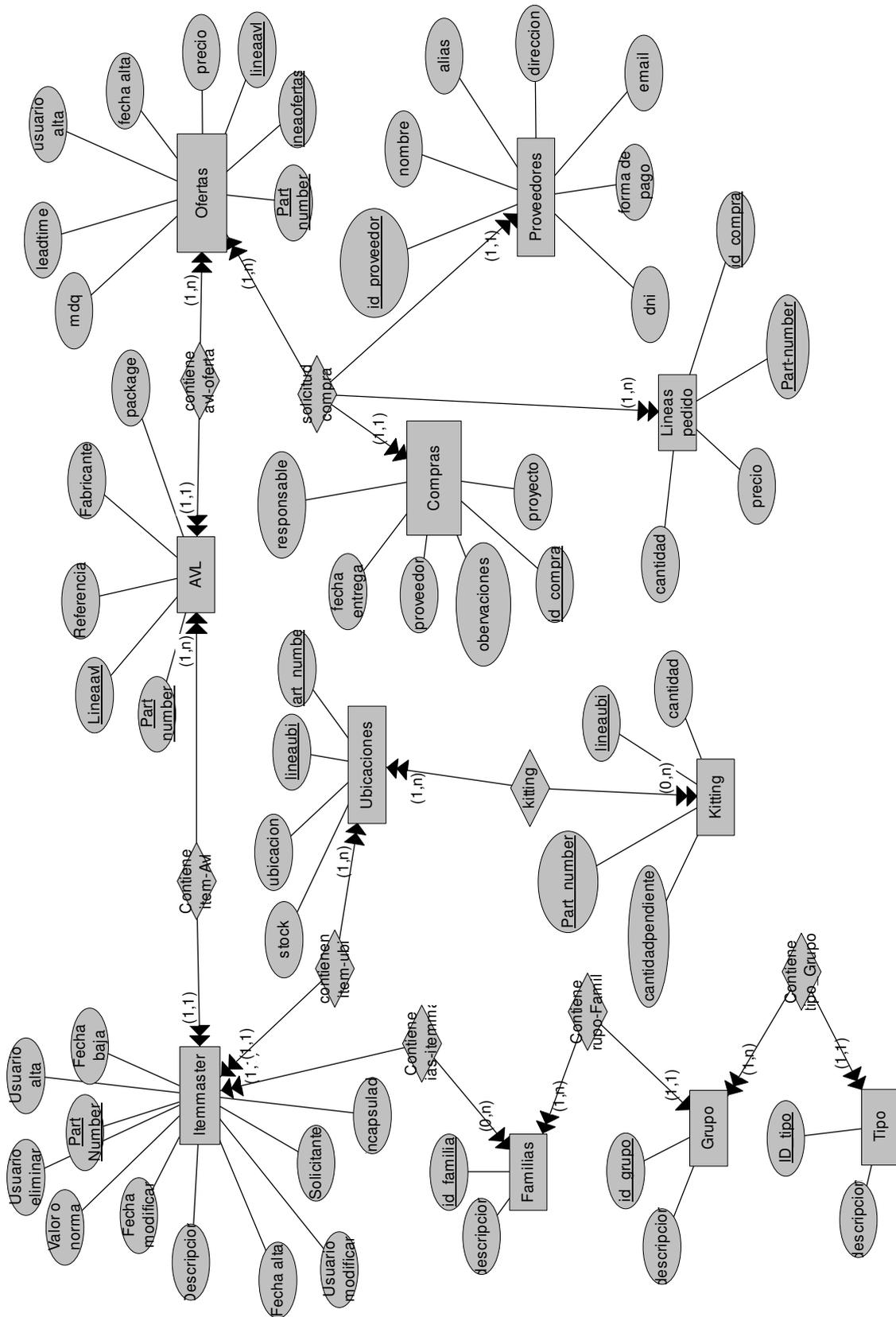


Figura 3.5 – Diagrama entidad-relación

## 4 DISEÑO

El diseño de la aplicación web se ha basado en una arquitectura multicapa de tres capas lógicas como son (Figura 4.1):

- 1) Nivel de interfaz de usuario o presentación.
- 2) Nivel de aplicación o lógico.
- 3) Nivel de persistencia de los datos.

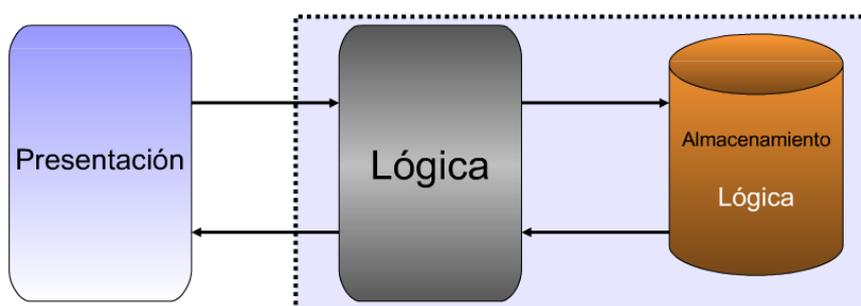


Figura 4.1 – Diseño de aplicación

El nivel de interfaz de usuario o presentación está basado en todos los documentos que se envía al servidor por parte de la aplicación mediante el usuario y que el servidor, de manera recíproca, le proporciona a éste toda la información que solicitó a través de formularios y/o enlaces.

El nivel de aplicación o lógico está constituido por todas las librerías que contienen clases para facilitar la operacional de la aplicación. En nuestro caso podemos mencionar la utilización de JQuery, fpdf y calendario.

El nivel de persistencia de los datos está formado por la base de datos y el sistema de gestión de la misma. Este nivel es el encargado de almacenar toda la información y controlar el acceso a la aplicación.

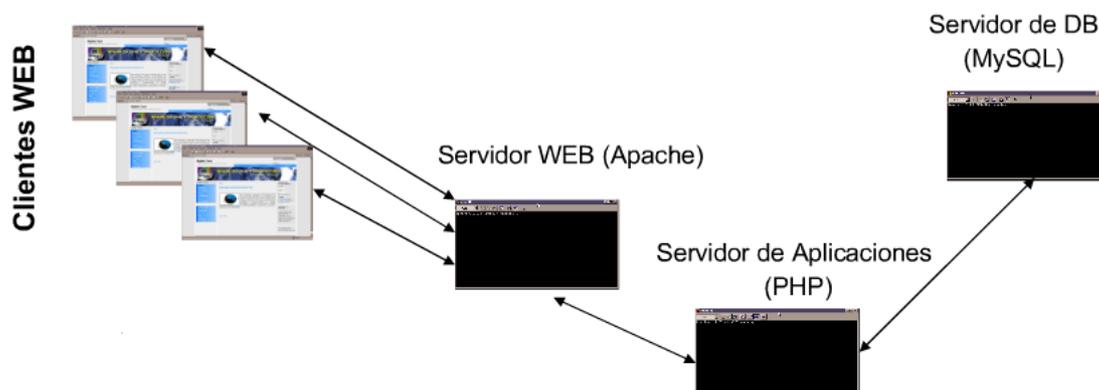


Figura 4.2 – Arquitectura multicapa

### 4.1 NIVEL DE INTERFAZ

Se ha diseñado una interfaz de usuario lo más amigable posible basándome en aspectos fundamentales como la facilidad de navegación por toda la aplicación, un manejo nada complejo y un aprendizaje sobre la misma rápido y conciso. Se hizo un pequeño patrón de diseño sobre la interfaz que cumplía con la imagen corporativa de la empresa. Dicha estructura puede resumirse en:

- Una cabecera con el logo de la empresa, nombre del menú donde se encuentra y nombre del usuario que está utilizando la aplicación.
- El cuerpo central según donde se encuentre el usuario puede ser un menú de enlaces hacia otros submenús, formularios basados en la información introducida por el usuario o información proporcionada por la aplicación. En el caso de las búsquedas se presenta la información con un título donde comunica al usuario el tipo de búsqueda que es, logo de la empresa , número de resultados encontrados y la tabla con los mismos donde en el margen derecho están las posibles acciones que se puedan realizar con esa búsqueda.

Como existe gran diversidad de páginas dentro de la aplicación a continuación detallaré los cambios existentes entre ellas para diferenciarlas y no exista ningún tipo de duda por parte del usuario.

#### ❖ Login

En la página de login (Figura 4.3) se puede apreciar en el cuerpo los dos campos que se necesitan para empezar a utilizar la aplicación. Como se observa la cabecera es la misma que la mencionada anteriormente con su logo de empresa, nombre de menú donde se encuentra y el nombre de usuario con la palabra usuario, ya que en este punto no se sabe quien se loguea.

**DAS**  
Photonics

**DATA WAREHOUSE**  
AUTENTIFICACION

**USUARIO**

Nombre:

Contraseña:

LOGIN

Introduzca usuario y contraseña en la casilla correspondiente.  
Cada usuario tendrá acceso privado según sus privilegios.  
Contactar con el administrador en caso de fallo/alta o baja en el sistema.

[Salir](#)

Figura 4.3 – Login

#### ❖ **Página principal**

La página principal (Figura 4.4) tiene una cabecera con el logo de la empresa, nombre de la aplicación, nombre del usuario y nombre del menú principal. En la parte del cuerpo se encuentran todos los menús por los cuales se puede navegar. Este cuerpo puede ser diferente al que muestro en este documento debido a los diferentes permisos y privilegios que tenga el usuario (es posible que no tengan acceso a los menús de compras, almacén, administración e ingeniería). En la parte inferior se encuentra el cierre de sesión de la aplicación al salir de la misma y volver al menú de login.

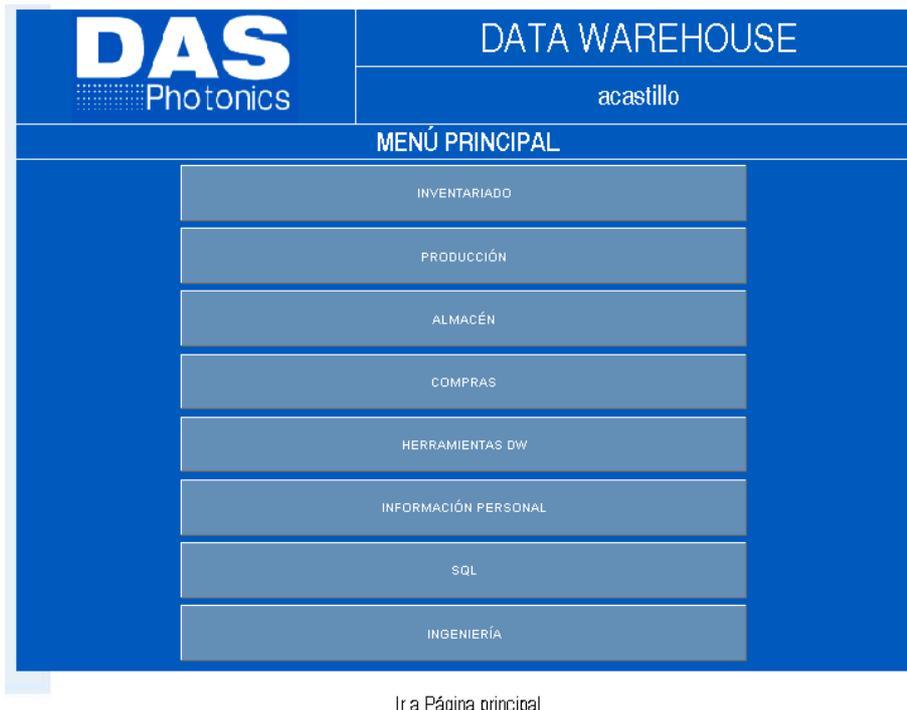


Figura 4.4 – Página Principal

#### ❖ **Enlaces**

Cualquiera de los menús de la página principal llevará al usuario a otros submenús como el que se muestra a continuación (Figura 4.5). En ellos se muestra, en el cuerpo, los diferentes enlaces que tiene acceso el usuario. Puede variar cada submenú según los permisos y privilegios que tenga. En la cabecera se aprecia el logo de la empresa, nombre del menú y nombre del usuario.



[Ir a DATA WAREHOUSE](#)

Figura 4.5 – Enlaces

### ❖ Formularios

El aspecto de los formularios es como el mostrado en la Figura 4.6. Tiene la misma cabecera que cualquier página de la aplicación, con su logo de la empresa, nombre del menú donde se encuentra y el nombre del usuario. En el cuerpo se encontrará, como cualquier de los formularios que compone la aplicación, los diversos datos y campos que se tendrán que rellenar para obtener o introducir la información en la base de datos.

Figura 4.6 – Formularios

❖ **Búsquedas con tablas**

En los formularios de búsqueda los resultados de la misma serán representados como se muestran en la Figura 4.7. Se compone de una cabecera donde se encuentra la ruta de enlaces a los que se ha accedido, el logo de la empresa y número de resultados encontrados. En la parte lateral derecha estará las posibles acciones que se podrán hacer con la búsqueda y en la parte inferior se situará el menú de botones para proceder a acciones de enlaces.

**LISTADO MAESTRO - CONSULTA PART NUMBER**

Nº	PN	Div	Descripción	Encapsulado/norma	Valor	DATA_SH	AVL	ALMACEN	COSTES PLANIFICACION	OFERTAS	DISPONIBILIDA
1	11010001	A	R 1K3 5% 0603 REEL	0603	1K3	○	●	●	●	●	●
2	11010002	A	R ZEROHM 5% 0603	0603	0R	○	●	●	●	●	●
3	11010003	A	R 51R 5% 0603 REEL	0603	51R	○	●	●	●	●	●
4	11010004	A	R 4K7 5% 0603 REEL	0603	4K7	○	●	●	●	●	●
5	11010005	A	R 10R 5% 0603 REEL	0603	10R	○	●	●	●	●	●
6	11010006	A	R 390R 5% 0603 REEL	0603	390R	○	●	●	●	●	●
7	11010007	A	R 3K9 5% 0603 REEL	0603	3K9	○	●	●	●	●	●
8	11010008	A	R 3R9 5% 0603 REEL	0603	3R9	○	●	●	●	●	●
9	11010009	A	R 39R 5% 0603 REEL	0603	39R	○	●	●	●	●	●
10	11010010	A	R 47R 5% 0603 REEL	0603	47R	○	●	●	●	●	●
11	11010011	A	R 560R 5% 0603 REEL	0603	560R	○	●	●	●	●	●
12	11010012	A	RSISTOR 10R 0402	0402	10R	●	●	●	●	●	●
13	11010013	A	RESISTOR 680R 0402	0402	680R	●	●	●	●	●	●
14	11010014	A	R 68R 5% 0603 REEL	0603	68R	○	●	●	●	●	●

Figura 4.7 – Búsqueda con tablas



4.1.1.2 USUARIO CON PERMISO DE ADMINISTRACIÓN EN ÓRDENES DE TRABAJO

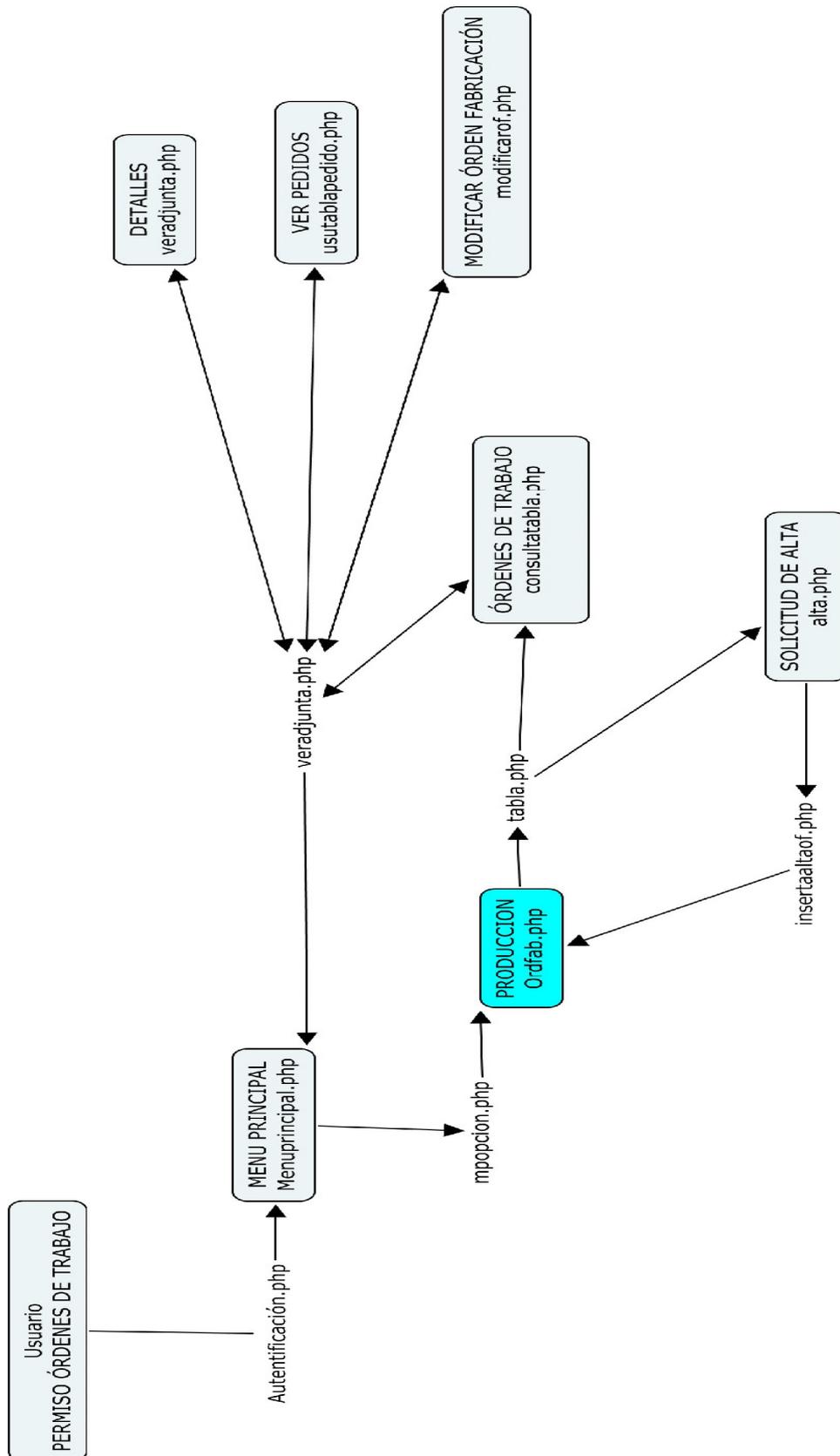


Figura 4.9 – Usuario con permisos de administración en órdenes de trabajo

4.1.1.3 USUARIO CON PERMISO ALMACÉN

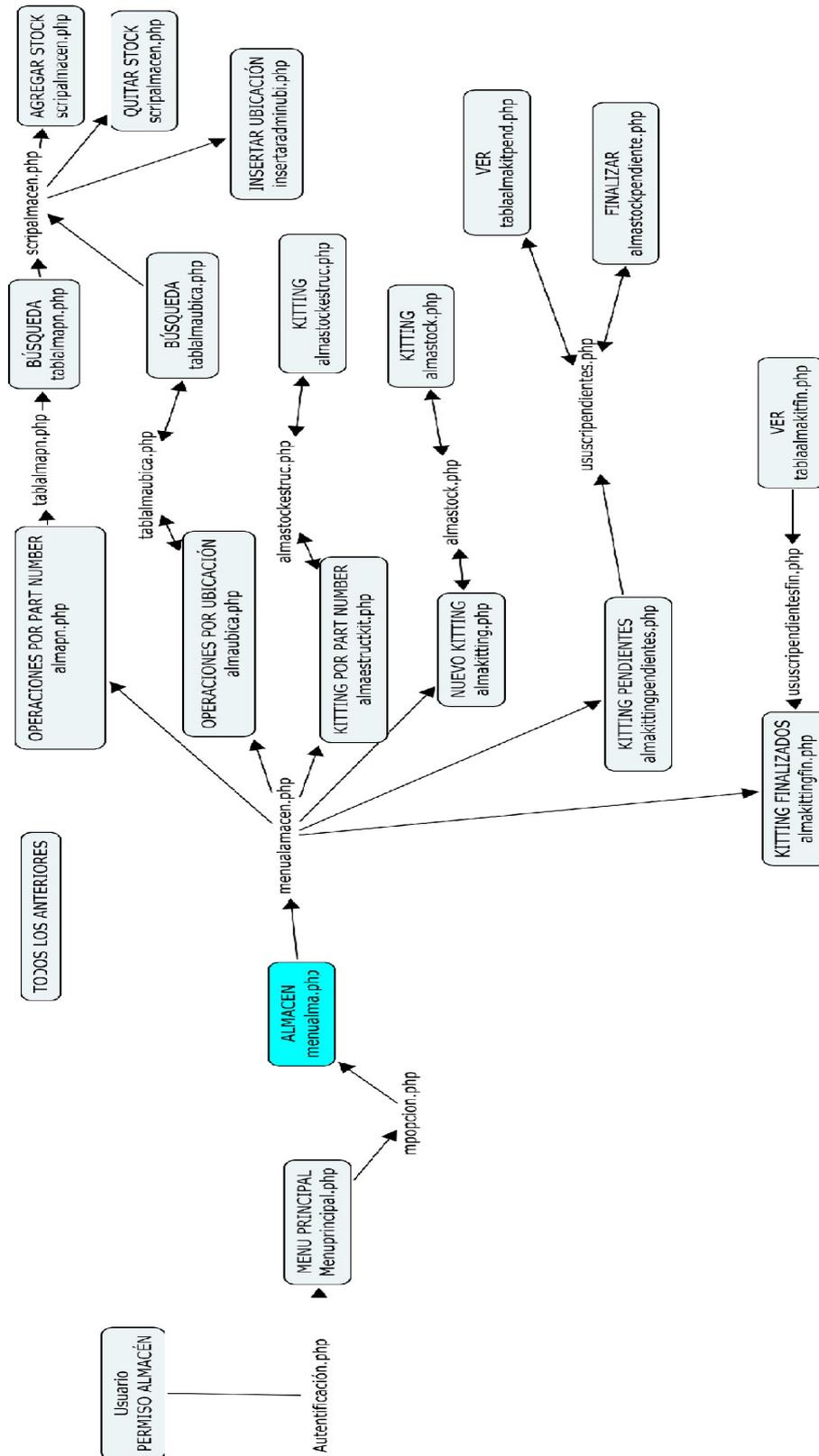


Figura 4.10 – Usuario con permisos de almacén

4.1.1.4 USUARIO CON PERMISO DE INGENIERÍA

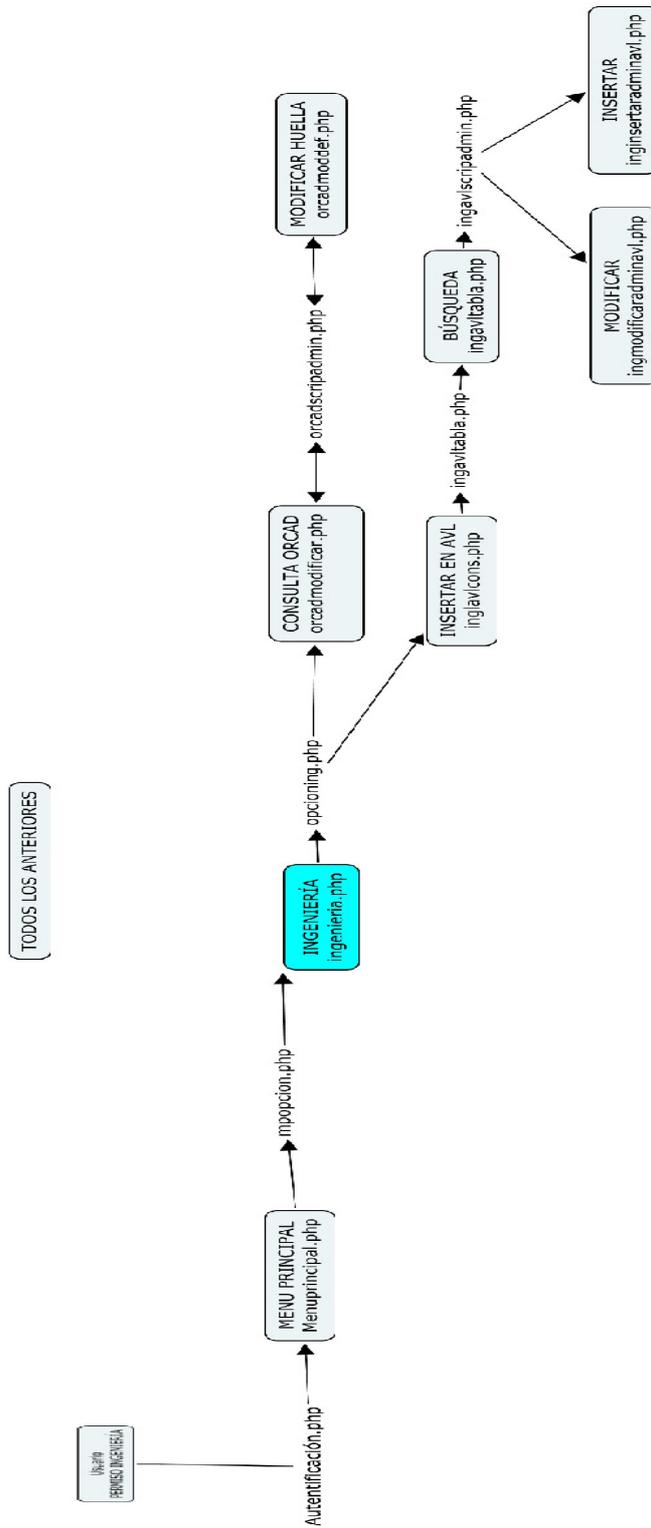


Figura 4.11 – Usuario con permisos de ingeniería

4.1.1.5 USUARIO CON PERMISO DE COMPRAS

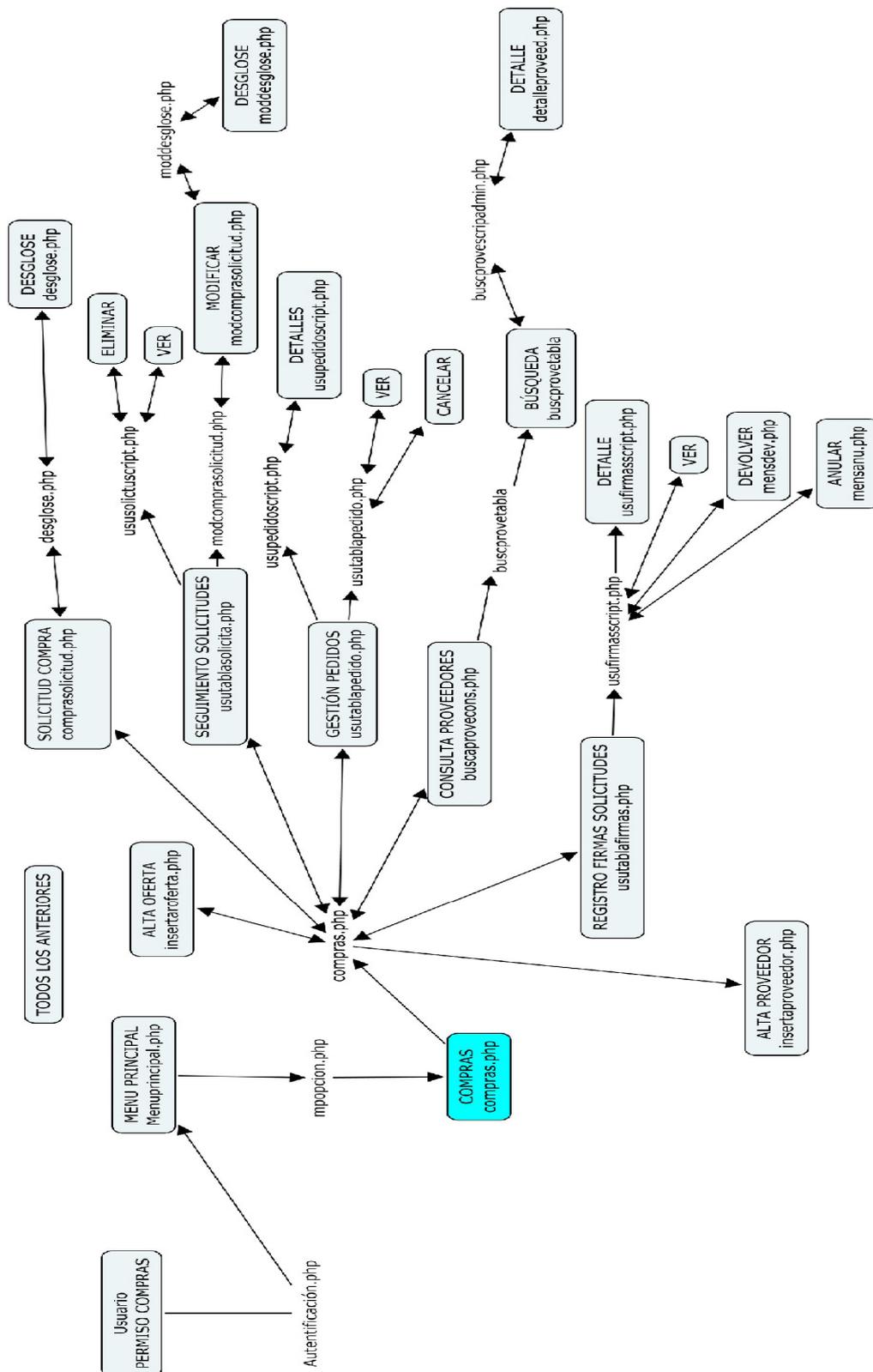


Figura 4.12 – Usuario con permiso compras

4.1.1.6 USUARIO ADMINISTRADOR

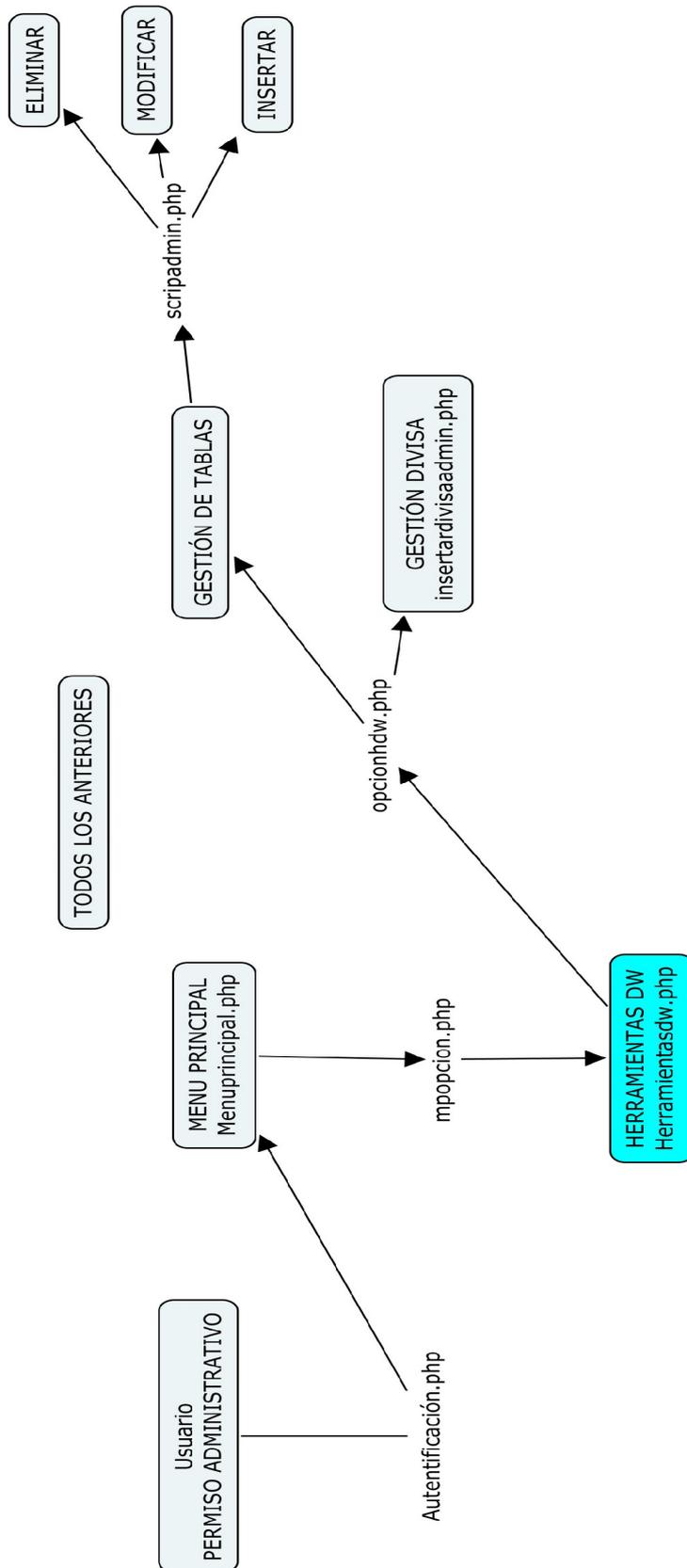


Figura 4.13 – Usuario administrador

## 4.2 NIVEL LÓGICO

El nivel lógico de la aplicación se forma por un conjunto de librerías creadas que incluyen las clases necesarias para el funcionamiento de la misma. Estas librerías ejecutan diferentes funciones relacionadas con operaciones de cálculo, transformación y validación de datos,...

Así mismo, este nivel permite diferenciar entre el nivel de interfaz y el nivel de persistencia, pudiendo así realizar cambios en dicho nivel, cambiar el SGBD o sustituir la base de datos por otro sistema de almacenamiento, sin afectar al nivel de interfaz y sólo ejecutando cambios en el nivel lógico para adaptarlo al nivel inferior.

En este nivel se intenta seguir las pautas de la programación orientada a objetos, en el que funcionalidades relacionadas se engloban dentro de clases, y con relaciones de herencia entre ellas. Por ello, en el nivel de interfaz, se aplicarán las clases para crear los objetos necesarios para realizar las operaciones.

Las librerías relacionadas con operaciones sobre base de datos están formadas por diversos componentes, los cuales tiene relación de herencia con una clase padre proporcionándoles las herramientas de conexión a base de datos, y así el resto sólo debe ocuparse de gestionar operaciones sobre la misma. Existen otros componentes que realizan otras tareas:

- Generación de fragmentos de página a toda la aplicación para el nivel de interfaz.
- Conexiones a otras páginas para gestionar búsquedas externas.
- Funciones varias de ámbito general usadas en el nivel de interfaz y sin relación entre sí.

## 4.3 NIVEL DE PERSISTENCIA

La aplicación utiliza una base de datos relacional para el nivel de persistencia, donde almacenar la información referida a elementos del catálogo, usuarios, noticias,...

En base al propósito de la aplicación, es básico, que la base de datos esté correctamente estructurada y diseñada con diversas entidades y sus correspondientes atributos relacionados entre sí, de forma coherente y eficiente.

Se diseñaron las diferentes entidades necesarias para la base de datos. Posteriormente fueron necesarias otras para poder relacionar entre sí varias entidades haciendo que éstas nuevas fuesen débiles donde a parte de su clave primaria necesitábamos de la clave primaria de la que dependían.

Respecto a los atributos de las entidades, todas disponían de una clave primaria pero ciertas entidades necesitaron de más atributos para componerla, debido a que al pasar a la tercera forma normal algunas relaciones desaparecerían, dieron lugar a este tipo de modificaciones. Posteriormente, la tercera forma normal fue la versión definitiva de la base de datos siendo la que dotase de orden estructurado y comportamiento y/o almacenamiento de los datos dentro de la misma.

### 4.3.1 ESQUEMA DE LA BASE DE DATOS

**Itemmaster** (Part Number:char(8), Descripción:char(64), Valor o norma:char(64), Encapsulado:char(64), Solicitante:char(64), Fecha alta:date(), Fecha baja:date(), Fecha modificar:date(), Usuario alta:char(64), Usuario eliminar: char(64), Usuario modificar:char(64))

**AVL** (Part number:char(8), Lineaavl:int(100), Referencia: char(64), Fabricante:char(64), package: char(64))

**Contiene item-Avl** (Part Number, Part number, Lineaavl)

Claves ajenas:

- Part Number referencia a Itemmaster.

**Ubicaciones** (Part number:char(8), lineaubi: int(100), stock:int(1000), ubicacion:int(8))

**Ofertas** (Part number:char(8), lineaofertas: int(100), precio: int(1000), mdq: int(100), leadtime, usuario alta:char(64), fecha alta:date(), lineaavl: int(100))

**Kitting** (Part number:char(8), lineaubi: int(100), cantidad: int(100), cantidadpendiente: int(100))

**Compras** (id\_compra:int(10), proyecto:char(8), responsable:char(8), fecha entrega:date(), proveedor:int(5), observaciones:char(8))

**Familias** (id\_familia: int(10), descripción:char(8), id\_grupo: int(100))

Claves ajenas:

- id\_grupo referencia a Grupo.

**Grupo** (id\_grupo: int(10), descripción:char(8))

**Tipo** (id\_tipo: int(10), descripción:char(8))

**Contiene tipo\_Grupo** (id\_tipo, id\_grupo)

Claves ajenas:

- Id\_tipo referencia a Tipo.

- id\_grupo referencia a Grupo.

**Contiene familias-itemmaster** (id\_familia, Part Number)

Claves ajenas:

- id\_familia referencia a Familias.

- Part Number referencia a Itemmaster.

**Contiene avl-oferta** (Part number, Lineaavl, Part number\_Ofertas, lineaofertas, lineaavl)

Claves ajenas:

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

- Part number referencia a AVL.
- Lineaavl referencia a AVL.
- Part number\_Ofertas referencia a Ofertas.
- lineaofertas referencia a Ofertas.
- lineaavl referencia a Ofertas.

**Solicitud compra** (id\_compra:int(8), Part number:char(8), lineaofertas:int(100), lineaavl:int(100), id\_proveedor:int(5), id\_compra\_Lineas pedido:int(100))

Claves ajenas:

- id\_compra referencia a Compras.
- Part number referencia a Ofertas.
- lineaofertas referencia a Ofertas.
- lineaavl referencia a Ofertas.
- id\_proveedor referencia a Proveedores.
- id\_compra\_Lineas pedido referencia a Lineas pedido.
- Part-number referencia a Lineas pedido.

**Proveedores** (id\_proveedor:int(5), dni:int(8), nombre:var(64), alias:var(64), dirección:var(64), email:var(64), forma de pago:var(64))

**Lineas pedido** (id\_compra:int(5), Part-number:var(8), cantidad:int(100), precio:int(100))

**Contienen item-ubi** (Part Number, part\_number, lineaubi)

Claves ajenas:

- #Part Number referencia a Itemmaster.
- #part\_number referencia a Ubicaciones.
- #lineaubi referencia a Ubicaciones.

**kitting** (part\_number, lineaubi, Part\_number, lineaubi\_Kitting)

Claves ajenas:

- part\_number referencia a Ubicaciones.
- lineaubi referencia a Ubicaciones.
- Part\_number referencia a Kitting.
- lineaubi\_Kitting referencia a Kitting.

# 5 IMPLEMENTACIÓN E INTEGRACIÓN

En este apartado se tratarán temas como las tecnologías que se han decidido utilizar para la implementación de la aplicación así como de cuáles son los aspectos más importantes en la evolución de la misma dentro de su integración. Es un punto importante dentro de toda la documentación para validar y aceptar el software, obtener información de la integración de cada una de las partes aquí mencionadas dentro del software y posteriormente cumplir con las fases de testeo en el punto 6.

## 5.1 TECNOLOGÍAS

La aplicación se ha desarrollado como una solución LAMP, creando una aplicación Web mediante esta combinación de tecnologías:

- . Linux, sistema operativo
- . Apache, servidor Web
- . MySQL, servidor de base de datos
- . PHP, lenguaje de programación

He optado por estas tecnologías ya que reúnen unas características básicas:

- La gran cantidad y calidad de documentación existente que permite un aprendizaje rápido y profundo de la materia.
- Buena interacción de estas tecnologías entre sí.
- Son tecnologías libres y gratuitas mantenidas por la comunidad de software libre.
- Se utilizan a nivel profesional y demuestran alta calidad y productividad.

Las tecnologías Linux, Apache y MySQL son componentes externos a la aplicación, explicado al detalle en el punto de integración.

### PHP

Es un lenguaje para programar scripts del lado del servidor incluido dentro del código HTML, siendo gratuito y multiplataforma.

El uso de PHP es fundamental en la aplicación. Las páginas se generan de forma dinámica en el servidor mediante los scripts en PHP procesados. He desarrollado un conjunto de librerías en PHP situados en el nivel de acceso a datos y que permiten independencia de la aplicación. Las páginas dinámicas permiten:

- Listados de todo el inventariado y sus búsquedas. Estas páginas cambian cuando se va actualizando el inventariado y las búsquedas permiten realizar listados personalizados.

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

- Modificaciones sobre todo el inventariado (fabricantes, proveedores, ofertas, ubicaciones, etc.) reflejados de inmediato en los listados y búsquedas mencionadas.
- Control de acceso a las diferentes secciones de la aplicación
- Validación de entrada de datos por los usuarios en formularios
- Verificar las diversas restricciones que se pueden realizar en cuanto a las operaciones.
- Redirecciones de forma dinámica a páginas distintas según condiciones del usuario.
- Envío, según la identificación del navegador cliente, de distintos elementos
- Acceso a la base de datos donde están los datos de la aplicación y sus modificaciones
- Realización de tareas automatizadas.
- Realización de estadísticas en base a las ofertas realizadas.

En la ejecución del proyecto he usado la versión PHP 4.3. de forma limitada, que ofrece una programación orientada a objetos, implementando parte de las librerías con relación de herencia entre ellas.

### CSS

Las hojas de estilo son un lenguaje formal que se utiliza para definir la presentación de un documento estructurado escrito en HTML, XML o XHTML. Éstas pueden insertarse directas en el código HTML en un archivo separado de hoja de estilos enlazando al documento principal. Este método, en el cual los cambios se limitan a un único archivo y además el tamaño de los documentos XHTML se reduce al no contener ya información referente al estilo, va ganando en eficiencia. Otra ventaja radica en que un cambio en la hoja de estilos derive en un cambio de estilo general para la aplicación, de otra manera habría que revisar uno a uno todos los documentos aplicándoles los estilos correspondientes

Las hojas de estilo en la aplicación se han utilizado para presentar las páginas web generadas por el servidor de manera estructurada, aplicando diseño a la aplicación para que los navegadores muestren al usuario de forma más agradable.

## 5.2 HERRAMIENTAS

Anteriormente hemos visto la aplicación como una presentación Web formada por una base de datos relacional en MySQL y por un portal Web en PHP, conectadas entre sí y ubicadas en una misma máquina con sistema operativo Linux y un servidor Web( Apache).

Los documentos HTML y scripts en PHP se han realizado usando DreamWeaver, especializado en los lenguajes HTML, XHTML y CSS. También permite trabajar con PHP.

Cabe destacar que DreamWeaver permite utilizar dentro de la propia interfaz, completos manuales de referencia de HTML y CSS, y el administrador de archivos incorporado deja trabajar con varios documentos a la vez.

Para facilitar el trabajo con la base de datos he utilizado phpMyAdmin que es un administrador Web de bases de datos MySQL y permite realizar las tareas de administración necesarias (gestionar tablas, eliminar campos,...)

Otra herramienta es la aplicación de retoque fotográfico The Gimp para la creación o modificación del logotipo, los iconos, toda la aplicación de ubicaciones y demás gráficos utilizados en la aplicación, siguiendo un criterio general que clarifica, simplifica el uso de la aplicación. Además de HTML y PHP, en algún caso he utilizado el lenguaje de scripting JavaScript.

## 5.3 DETALLES DE LA IMPLEMENTACIÓN

### 5.3.1 PERFILES DE USUARIO

Existen diversos tipos de usuario según los permisos que obtengan o se le atribuyan en cada momento. La aplicación dependiendo del tipo de usuario que la esté utilizando, adapta automáticamente la interfaz con la funcionalidad adecuada al nivel de permisos de dicho usuario. Por lo tanto a cualquier usuario no identificado que acceda a la aplicación se le redirigirá a la página principal. A continuación se relatarán diversos tipos de usuarios:

Uno de ellos será el usuario básico el cual podrá consultar cualquier información sobre cualquier componente de la base de datos. Obtendrá información sobre fabricantes, proveedores, ofertas, ubicaciones, estructuras donde se está utilizando, stock. Otra funcionalidad será el solicitar dar de alta un componente nuevo y mirar en qué estado se encuentra su petición de alta. También podrá modificar datos personales y hacer cualquier tipo de consulta sobre cualquier orden de trabajo dada de alta (responsable, componentes, pedidos, fechas de orden, proyecto, horas dedicadas y dinero dedicado).

Otro usuario sería el que obtiene permisos de almacén que teniendo todas las acciones del usuario básico además tiene funcionalidades como la adición y extracción de stock de cualquier componente, la creación de nuevas ubicaciones para componentes ayudado de una aplicación de codificación de ubicaciones sobre un mapa de la empresa, modificación y eliminación de ubicaciones y la posibilidad de hacer kitting completos o parciales (parecido a un carrito de la compra).

Otro tipo de usuario es el que obtiene los permisos de compras con los que podrá hacer solicitudes de compra y verificar el estado en el que se encuentran y donde cabe destacar otro usuario del tipo administrador que se encargará de firmar esas solicitudes para que posteriormente se conviertan en pedidos. Este último usuario aparte de tener las funcionalidades del usuario de compras básico también se encarga de la actualización del valor de las divisas.

Otro tipo de usuario es el encargado de la administración de las órdenes de fabricación las cuales podrá dar de alta en cualquier momento sin supervisores por el medio. Tendrá permisos de modificación de datos sensibles de la orden así como de la eliminación de la misma. Este tipo de usuario podrá obtener información de todos los pedidos que tenga una orden de fabricación a diferencia del usuario básico que solo podrá ver los pedidos hechos por él en cada una de ellas.

Por último existe un usuario administrador del sistema que se encargará de todo lo anterior y aparte tendrá acceso a la parte de administración de permisos para cada uno de los usuarios así como el dar de alta a un usuario nuevo dentro del sistema. Este usuario también se responsabilizará de dar de alta una lista de componentes mediante altas masivas si el fichero que las contiene está correctamente creado bajo la estructura acordada.

#### 5.3.2 AUTENTIFICACIÓN DE USUARIOS

La aplicación utiliza un mecanismo de autenticación basado en el seguimiento de sesiones que posee PHP y consultas a la base de datos almacenando los identificadores de usuario y su contraseña cifrada en md5.

Al acceder el usuario a la página de identificación, introduce su identificador y contraseña. En un futuro el envío de esta información se producirá mediante el protocolo HTTP seguro o HTTPS, de esta forma ambos viajarán cifrados por la red. En este momento el sistema comprueba si los datos introducidos son válidos y si el usuario existe a través de la tabla accesos de la base de datos. Si todo es válido establecemos las variables de sesión indicando los permisos del usuario y se cargará la página de inicio personalizada para dicho usuario.

La sesión permanecerá abierta hasta que el usuario finalice mediante la opción cerrar sesión o cerrando el navegador Web.

Las páginas restringidas a usuarios identificados comprueban el tipo de usuario que es y si dicho usuario tiene permiso necesario para visualizar dicha página. Si no tiene acceso, se expulsará de la aplicación y se enviará a la página de autenticación y así controlaremos el acceso a la funcionalidad de la aplicación.

#### 5.3.3 ASPECTOS GENERALES

La información mostrada por la aplicación está formateada en HTML y generada mediante el código en PHP, el cual, dota de funcionalidad a la aplicación.

Por ello, la aplicación es un conjunto de páginas HTML con scripts en PHP que ejecuta el servidor.

Las diversas clases utilizadas en los scripts que componen la aplicación se clasifican en tres tipos:

- . Relacionadas con la interfaz de usuario, que muestran y obtienen información de los mismos
- . Realizan la lógica de la aplicación
- . Encargados del nivel de persistencia (recuperar y salvar información de base de datos)

En base a esta estructuración por capas se consigue cierta independencia entre las diferentes partes de la aplicación; podemos refinar el funcionamiento de las funciones de acceso a datos o cambiar el SGBD utilizado sin preocuparnos en el nivel de presentación o la lógica del programa. He incluido una serie de librerías en PHP encargadas del nivel de persistencia o de acceso a los datos, de forma que el nivel lógico de la aplicación accede a éstas y no a la propia base de datos.

La información introducida por los usuarios al sistema se recoge a través de formularios HTML.

Los campos de los formularios se envían entre páginas mediante el método POST codificando los datos a enviar en HTML.

Resaltaremos el empleo de campos ocultos en los formularios HTML, para que la aplicación traspase información propia de una página a otra.

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

La funcionalidad de la aplicación se basa en consultas al sistema de gestión de bases de datos MySQL. Ésta interpreta los resultados obtenidos y actúa en consecuencia..

Referente al diseño, la aplicación Web se crea siguiendo las especificaciones de HTML y de estilos CSS, de forma que se visualice correctamente en cada navegador sin perder ninguna de sus funcionalidades.

## 5.4 INTEGRACIÓN

Los componentes de origen externo utilizados en la aplicación son el servidor Web y el Gestor de Bases de Datos.

El servidor Web que he elegido es Apache, instalado sobre el S.O. GNU/Linux. Se ha utilizado con sus correspondientes módulos de PHP, para que sea capaz de interpretar scripts en el lado del servidor.

El módulo de PHP debe estar compilado con las siguientes directivas adicionales:

- *--enable-trans-sid*, con el que las URIs relativas se modifican de forma que contienen el *session id* automáticamente (por ejemplo en los enlaces o formularios), con lo que se consigue mantener la sesión incluso sin el uso de *cookies*.

El SGBD elegido ha sido MySQL con el que he implementado el nivel de consistencia de la aplicación, por las evidentes ventajas de almacenar la información utilizando una base de datos frente a ficheros.

Cabe destacar que PHP se combina a la perfección con MySQL, por lo que incorporarlo en el sistema no ha sido una tarea costosa.

Sin embargo, de cara a futuras funcionalidades autónomas (como puede ser pedidos automáticos, emails de pedidos, alarmas de stock por debajo de su nivel mínimo), he encontrado problemas de implementación. Debido a que MySQL es una base de datos sin integridad referencial y sin soporte para disparadores, su velocidad es mucho mayor frente a otras soluciones, mucho más completas como el PostgreSQL, MS SQL Server, Oracle, etc. Por ello se decidió debido a su mayor expansión en internet y principalmente por su rendimiento, pero pueden llegar a producir situaciones no deseadas.

Debido a, la complejidad alta y al coste operacional, lo hacía inviable por las operaciones inútiles que se llevarían a cabo.

Se decidió para este tipo de funcionalidades futuras delegar esta tarea en un programa externo: el programador estándar de tareas de UNIX: *crontab*, que se encargará diariamente de éstas acciones.

Generalmente, *crontab* utiliza un “demonio”, *crond*, que se ejecuta continuamente en segundo plano y comprueba cada minuto si alguna de las tareas programadas necesita ser ejecutada. De ser así, las ejecuta.

Evidentemente esta es una solución alternativa, quizá lo más correcto sería la utilización de un SGBD que contemplase los disparadores como puede ser PostgreSQL, que provee soporte para lenguajes SQL92/SQL99, transacciones, integridad referencial, procedimientos almacenados y extensibilidad de tipos, con lo que se evitaría el uso de una solución dependiente del S.O. como es el uso del programador de tareas.

## Memoria del Proyecto Final de Carrera

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

Dada la arquitectura que se ha desarrollado en la aplicación, el SGBD podría fácilmente ser intercambiado por otro. Los cambios a realizar entonces serían en la capa de aplicación, ya que se encuentra por encima del nivel de persistencia donde se encuentra el SGBD y es la que se comunica con éste. Estos cambios además se realizarían únicamente en las llamadas a funciones dependientes del SGBD utilizado, ya que el resto de funciones son generación de consultas en SQL u otras no relacionadas con el nivel de persistencia, sino con el de presentación.

# 6 EVALUACIÓN Y PRUEBAS

## 6.1 EVALUACIÓN

He realizado una serie de pruebas y análisis de la usabilidad. La usabilidad se centra en las necesidades del usuario:

- Pensar en qué modo el usuario desearía encontrar la información.
- Definir un árbol de navegación intuitivo como referencia de navegabilidad sin excesivos enlaces, y mostrándolos a medida que el usuario se interesa por ellos.
- Definir un diseño visual que transmita justo lo que se quiere sin que éste produzca confusión.
- Tener en cuenta las limitaciones de los recursos del usuario a nivel de ancho de banda, capacidad de procesamiento y resolución.

En las fases de diseño e implementación, con el diseño de la interfaz, el análisis etc. se ha tenido en cuenta este aspecto importante. Por lo tanto la usabilidad se ha ido incorporando de un modo interactivo a la vez que se diseñaba e implementaba la aplicación, intentando en todo momento que esta fuese lo más fácil, cómoda e intuitiva posible para el usuario final.

## 6.2 PRUEBAS

En este apartado se ha forzado la aplicación a situaciones extremas en diversas funcionalidades y en el caso de que no cumpliesen con los requisitos establecidos se ha tenido que tomar decisiones importantes como rediseño, reprogramación y reimplementación. También se ha sometido a pruebas del tipo formatos, resoluciones y tiempo de cómputo a la hora de hacer búsquedas dentro del sistema. Se ha finalizado con una aplicación estable y compacta con el éxito de todos los objetivos planteados y preparada para futuras mejoras o ampliación de funcionalidades.

Seguidamente se mostrarán diversos casos de uso que se han ido mencionando durante el documento.

### 6.2.1 PRUEBA SOLICITUD DE ALTA DE UN NUEVO COMPONENTE

1. Se rellena el formulario.

**DAS** Photonics

### Solicitud Nueva Alta Part Number

Usuario: acastillo

Solicitante:acastillo Fecha Solicitud:27-06-2011 Familia: Division: (Elija opción)

#### Datos Básicos(Obligatorios)

Division  Grupo  Familia  Tipo

Descripción

Encapsulado/Norma  Valor

Comentario Componente

#### Lista de Fabricantes Aprobados [AVL] (Obligatorios)

Fabricante  Referencia Fabricante  Package

#### Oferta

Proveedor  Fecha de oferta

Referencia Proveedor

MDQ  Plazo de entrega (en semanas)

Precio Unitario  Factor Lote

DataSheet  Examinar...

Comentario para supervisor

Alta Part Number  Menu Principal

Figura 6.2.1.1 – Formulario

2. Se agrega si se quisiese un archivo en formato pdf que irá adjunto al componente con la codificación interna que se le asignó al componente.

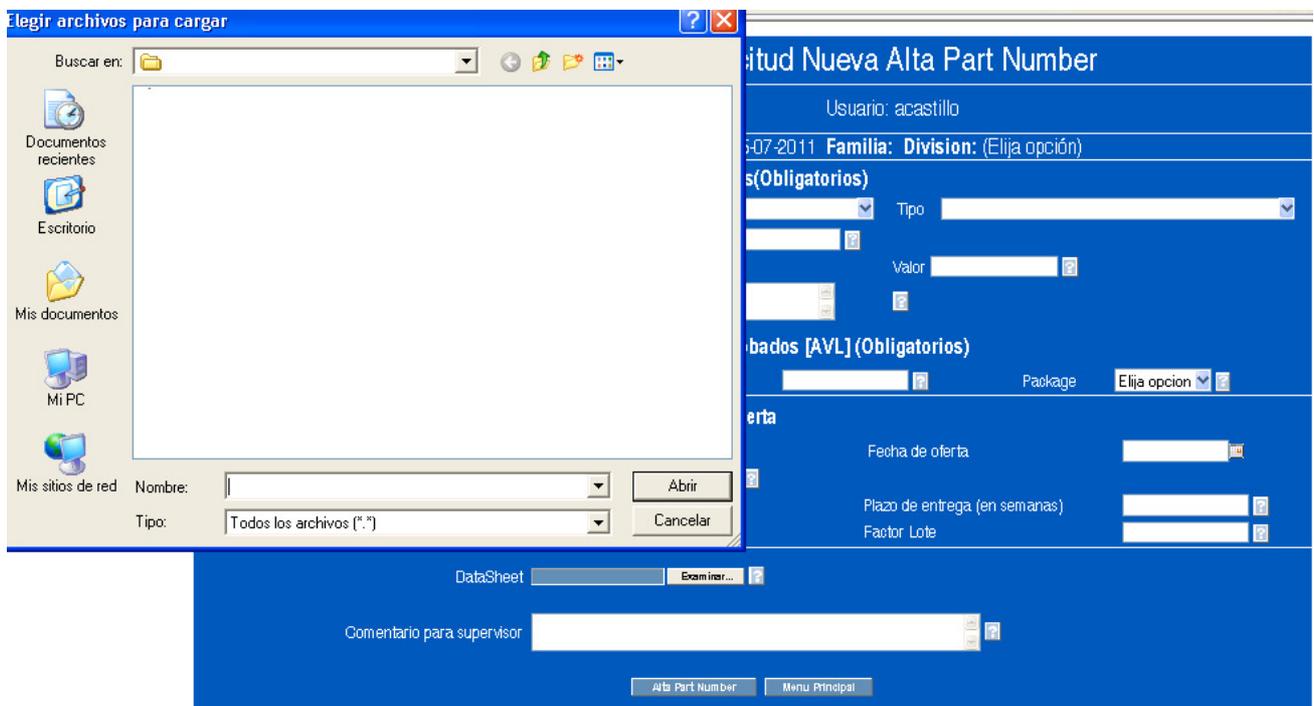


Figura 6.2.1.2 –Adjuntar archivo

Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

3. El componente está dado de alta parcialmente hasta que un administrador lo compruebe y lo agregue al sistema.
4. El usuario puede modificar el alta mientras no esté dada de alta definitiva. También puede reenviar la solicitud en cuanto corrija los errores que pueden haber surgido.

## 6.2.2 PRUEBA CONSULTA SOBRE INFORMACIÓN DE CUALQUIER COMPONENTE

1. Se rellena el formulario con los datos que se quieren hacer coincidir con la búsqueda.

Figura 6.2.2.1 –Consulta PART NUMBERS

2. Se muestra una tabla con todas las coincidencias, En su parte derecha existe las diferentes opciones que tiene el usuario para poder buscar la información deseada.

### LISTADO MAESTRO - CONSULTA PART NUMBER

PN	Div	Descripción	Encapsulado/norma	Valor	DATA	SH	AVL	ALMACEN	COSTES PLANIFICACION	OFERTAS	DISPONIBILIDAD	WI
11C10003	A	R 51R 5% 0603 REEL	0603	51R	○	●	●	●	●	●	●	●
11C10004	A	R 4K7 5% 0603 REEL	0603	4K7	○	●	●	●	●	●	●	●
11C10005	A	R 10R 5% 0603 REEL	0603	10R	○	●	●	●	●	●	●	●
11C10006	A	R 390R 5% 0603 REEL	0603	390R	○	●	●	●	●	●	●	●
11C10007	A	R 3K9 5% 0603 REEL	0603	3K9	○	●	●	●	●	●	●	●
11C10008	A	R 3R9 5% 0603 REEL	0603	3R9	○	●	●	●	●	●	●	●
11C10009	A	R 39R 5% 0603 REEL	0603	39R	○	●	●	●	●	●	●	●
11C10010	A	R 47R 5% 0603 REEL	0603	47R	○	●	●	●	●	●	●	●
11C10011	A	R 560R 5% 0603 REEL	0603	560R	○	●	●	●	●	●	●	●
11C10012	A	RESISTOR 10R 0402	0402	10R	●	●	●	●	●	●	●	●
11C10013	A	RESISTOR 680R 0402	0402	680R	●	●	●	●	●	●	●	●
11C10014	A	R 68R 5% 0603 REEL	0603	68R	●	●	●	●	●	●	●	●
11C10015	A	RESISTOR 100R 1206	1206	100R	●	●	●	●	●	●	●	●

Figura 6.2.2.2 –Resultado búsqueda

- Se podrá ver sus fabricantes.

### LISTADO MAESTRO - CONSULTA PART NUMBER - AVL



PN	Div	Descripción	Fabricante	Ref.Fabricante	Package
11010003	A	R 51R 5% 0803 REEL	BIOHMSA	--	Reel

Figura 6.2.2.3 –Información AVL

- Se podrá ver sus ofertas.

### LISTADO MAESTRO - CONSULTA PART NUMBER - OFERTAS



PN	Descripción	Fabricante	Ref.Fabricante	Proveedor	Ref.Proveedor	Precio	MDQ	Factor Lote	Leadtimeweek	Fecha ofertada	PRIORIDAD Proveedor	PRIORIDAD C
11010004	R 4K7 5% 0803 REEL	BIOHMSA	--	BIHOMSA	CH0874701J	0	5000	1	99	14-05-2007	NO	NO

Figura 6.2.2.4 –Información OFERTAS

- Se podrá ver sus ubicaciones.

### LISTADO MAESTRO - CONSULTA PART NUMBER - ALMACEN



PN	Div	Descripción	Stock	Ubica	Datecode	Lotepedido	Comubica
11010004	A	R 4K7 5% 0803 REEL	4999	1A22000	s/r	s/r	Reel

Figura 6.2.2.5 –Información ALMACEN o DISPONIBILIDAD

6. Se podrá ver su pdf asociado.

### LISTADO MAESTRO - CONSULTA PART NUMBER

10 279 COMPONENTES

Descripción	EN	COSTES PLANIFICACION	OFERTAS	DISPONIBILIDA
R 51R 5% 0603 REEL		•	•	•
R 4K7 5% 0603 REEL		•	•	•
R 10R 5% 0603 REEL		•	•	•
R 390R 5% 0603 REEL		•	•	•
R 3K9 5% 0603 REEL		•	•	•
R 3R9 5% 0603 REEL	0603	•	•	•
R 39R 5% 0603 REEL	0603	•	•	•
R 47R 5% 0603 REEL	0603	•	•	•
R 560R 5% 0603 REEL	0603	•	•	•
RESISTOR 10R 0402	0402	•	•	•
RESISTOR 680R 0402	0402	•	•	•
R 68R 5% 0603 REEL	0603	•	•	•
RESISTOR 100R 1206	1206	•	•	•

**Descarga de archivos**

¿Desea abrir o guardar este archivo?

Nombre: 11010012.pdf  
 Tipo: Adobe Acrobat Document  
 De: 10.20.1.220

Abrir Guardar Cancelar

Aunque los archivos procedentes de Internet pueden ser útiles, algunos archivos pueden llegar a dañar el equipo. Si no confía en el origen, no abra ni guarde este archivo. [¿Cuál es el riesgo?](#)

Figura 6.2.2.6 –Información DATASHEET

7. Se podrá ver en donde puede estar usado y la cantidad que se necesita. Está parte es muy útil para saber por parte del usuario cuantas unidades necesita de ese componente dentro una estructura que conforman un producto.



Nivel	PN	Descripción	Div	Cantidad
0	11010004	R 4K7 5% 0603 REEL	A	

Nº	WHERE IS USED	Descripción	CANTIDAD	WI
1	C1201410	PCBA Control	1	•
2	C2502110	PCBA Módulo Tx	1	•
3	C2502130	PCBA RF Input	1	•

Figura 6.2.2.7 – Información WI

### 6.2.3 PRUEBA CONSULTA SOBRE INFORMACIÓN DE CUALQUIER ÓRDEN DE FABRICACIÓN

1. Se presiona el botón de órdenes de fabricación

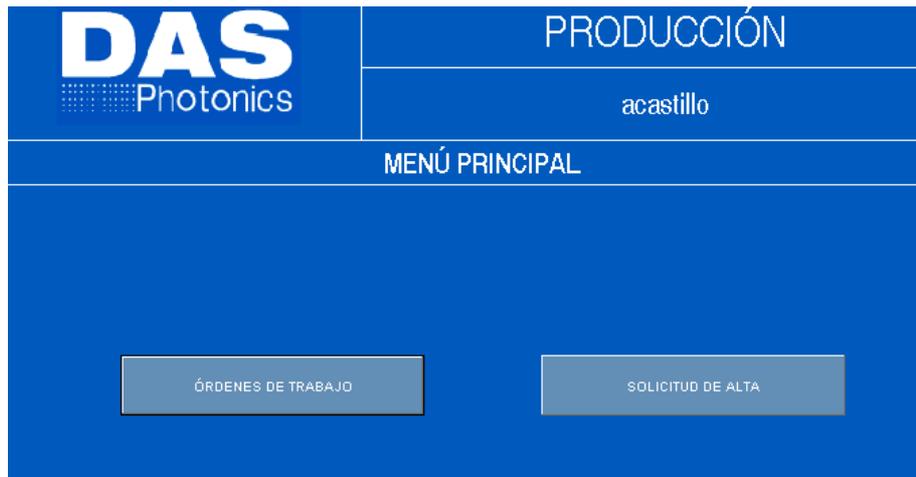


Figura 6.2.3.1 –Presionar botón “ÓRDENES DE FABRICACIÓN”

- Se muestra una tabla con todas las órdenes de fabricación existentes y que coincidan con los filtros que se encuentran en la parte superior de la misma.

Nº OT	Cliente	Lanzamiento	Proyecto	Producto	PN	Denominación	Estado	Detalles	Modificar	Ver Pedidos
DF100001	DAS Photonics	10-03-2010	12 DF PADC	PhotoLumia ADC	01200000	PADC 2 antenas	En proceso	●	●	●
DF100004	DAS Photonics	09-06-2010	20 DF LAMDA	Link Analógico Modulación Directa	52001000	Link Óptico (TX + RX)	En proceso	●	●	●
DF100007	DAS Photonics	22-07-2010	24 OT LO FOTONICO	Laser Pulsante	52400010	Prototipo LO fotónico	En proceso	●	●	●
DF100000	DAS Photonics	22-07-2010	21 OT L3AP	Link Óptico Altas Prestaciones	52'000000	Link Altas prestaciones 1 transceiver	En proceso	●	●	●
UH100009	DAS Photonics	22-07-2010	22 U1 ID	Líneas Retardo de RF	52200000	Línea de retardo de RF	En proceso	●	●	●
DF1000'0	ESA	23-09-2010	L0 AS LOD	Distribuidor DL	--	Diseño distallado	En proceso	●	●	●
DF1000'1	DAS Photonics	25-10-2010	19 DF COA	Conector Óptico Activo	51903000	COA proto Tx+Rx	En proceso	●	●	●
DF110001	DAS Photonics	14-01-2011	00 OT PCBs	PCBs Comunes	n/a	Equipos Dem u Ministerio de Defensa	En proceso	●	●	●
UH110002	DAS Photonics	20-01-2011	U1 -- LAB	Equipamiento Laboratorio	n/a	Equipamiento Laboratorio	En proceso	●	●	●
DF110003	DAS Photonics	26-01-2011	25 DF LINK OPTICO	Links Analógicos	52502000	Links 18 GHz VME	En proceso	●	●	●
DF110004	DAS Photonics	26-01-2011	26 DF LINK OPTICO	Links Analógicos	pendien:	Lineación Link	En proceso	●	●	●
DF110005	DAS Photonics	28-01-2011	L1 -- LAB	Tools y Consumibles	n/a	Compras de tools comerciales y consumibles de laboratorio	En proceso	●	●	●
DF110006	DAS Photonics	10-02-2011	00 OT PCBs	PCBs Comunes	varios	Paneles PCBs Mixdos	En proceso	●	●	●
DF110008	DAS Photonics	11-02-2011	10 OT PCBs	PCBs Comunes	varios	Paneles PCBs Mixdos	En proceso	●	●	●

Figura 6.2.3.2 –Órdenes de fabricación

- Se podrá ver detalles de cualquier orden de fabricación.

ORDEN DE TRABAJO - Detalle

Nº de Orden: OF100004      Project Manager: --      Estado: En proceso

**Datos del Cliente**

Nombre	DAS Photonics	Cód. Cliente	0000	División	DF
Proyecto DAB	2D DF LAMBDA - Link Analógico Modulación Directa	Pedido	Interno	Fecha pedido	09-09-2010

**Datos de Fabricación**

PN DAB	52001000	Autorización Compra	SI
Denominación	Link Óptico (TX + RX)		
Cantidad	1 de cada		
Fecha de entrega	01-09-2010		

**Datos Económicos**

Facturación	0 (EUR)
-------------	---------

**Datos de Costes**

	PREVIOS	TOTAL	DIFERENCIA
Materia	0	0	0
TOTAL EB	0	0	0
Compra	0	0	0
Inventarios	0	0	0
Subcontrataciones	0	0	0
Costos Variables	0	0	0
TOTAL EB	0	0	0

**Observaciones**

Prototipo de evaluación  
 Incluye al PN 52002010

Valor Uds OTC
Menu Principal Producción

Figura 6.2.3.3 – Ver detalles

- Se podrá ver pedidos de cualquier orden de fabricación.

### Gestión de Pedidos



N Pedido	Proyecto	OF	Tipo de compra	Disponibilidad	Proveedor	Total	Divisa	Urgente	Fecha según pedido	Fecha comprometida	Fecha llegada	Fecha cierre	Estado	DETALLES	PDF	CANCELAR
CSSE20110613-1	25 DF LINK OPTICO	CF110003	Aprovisionamiento	18.960,1600	FARNELL	32,300	EUR	SI	20/06/2011		22/06/2011		Recepcionado	<a href="#">●</a>	<a href="#">●</a>	<a href="#">●</a>
OT20110627-1	25 DF LINK OPTICO	CF110003	Aprovisionamiento	18.960,1600	INFRACTIVE	3.290	EUR	SI	05/08/2011				En curso	<a href="#">●</a>	<a href="#">●</a>	<a href="#">●</a>
CT20110627-2	25 DF LINK OPTICO	CF110003	Aprovisionamiento	18.960,1600	INFRACTIVE	4.490	EUR	SI	05/08/2011				En curso	<a href="#">●</a>	<a href="#">●</a>	<a href="#">●</a>
OMI20110607-1	25 DF LINK OPTICO	CF110003	Aprovisionamiento	18.960,1600	UMESAL	328,6000	EUR	SI	15/06/2011	16/06/2011	16/06/2011	16/06/2011	Cerrado	<a href="#">●</a>	<a href="#">●</a>	<a href="#">●</a>
OMI2011062-1	25 DF LINK OPTICO	CF110003	Aprovisionamiento	18.960,1600	JURGA	105	EUR	NO	05/07/2011		01/07/2011		Recepcionado	<a href="#">●</a>	<a href="#">●</a>	<a href="#">●</a>
JEX20110214-1	25 DF LINK OPTICO	CF110003	Aprovisionamiento	18.960,1600	TECHDRIBIA	101,5400	EUR	NO	23/02/2011	23/02/2011	23/02/2011	23/02/2011	Cerrado	<a href="#">●</a>	<a href="#">●</a>	<a href="#">●</a>
JEX20110303-1	25 DF LINK OPTICO	CF110003	Aprovisionamiento	18.960,1600	NEYPC	73	EUR	SI	09/03/2011	08/03/2011	08/03/2011	08/03/2011	Cerrado	<a href="#">●</a>	<a href="#">●</a>	<a href="#">●</a>
PG20110625-1	25 DF LINK OPTICO	CF110003	Aprovisionamiento	18.960,1600	Agitron Inc.	225	USD	NO	24/08/2011				En curso	<a href="#">●</a>	<a href="#">●</a>	<a href="#">●</a>

Figura 6.2.3.4 – Ver pedidos

- En caso de ser administrador puede modificar y cambiar datos relevantes como pudieran ser los datos económicos.

### 6.2.4 PRUEBA EXTRAER O QUITAR STOCK DE UN COMPONENTE EN UNA UBICACIÓN DETERMINADA

1. En el formulario existe una pequeña aplicación de ubicaciones por si el usuario no supiese la ubicación exacta de la misma.

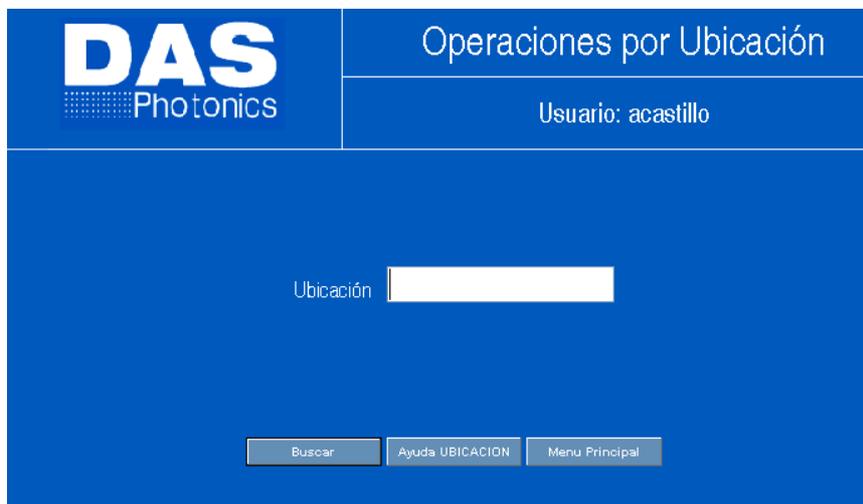


Figura 6.2.4.1 –Operaciones por ubicaciones

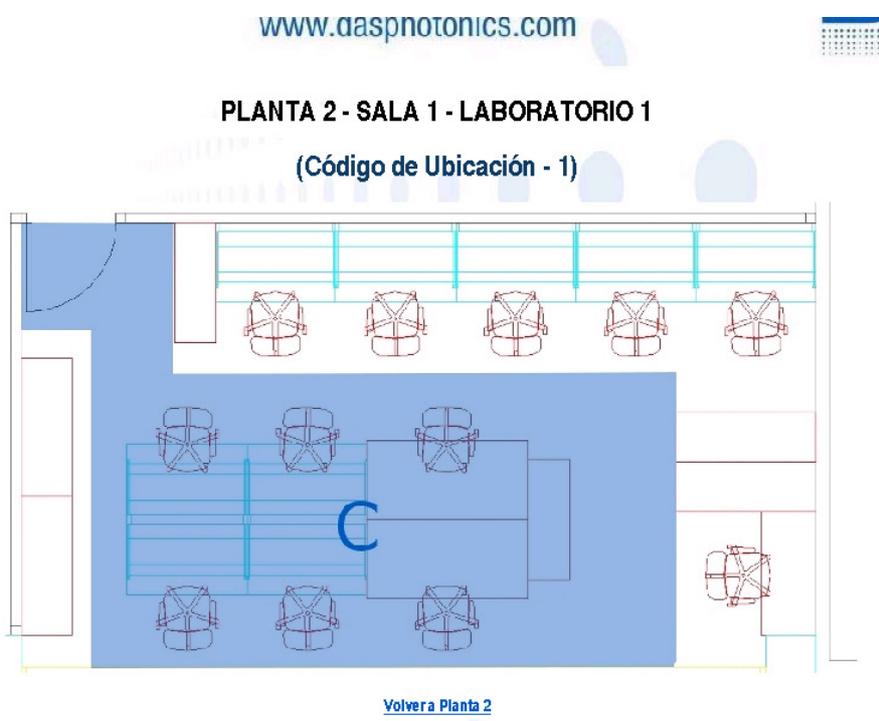


Figura 6.2.4.2 –Aplicación para codificar ubicación

2. Una vez introducida se muestra una tabla con todos los componentes de esa ubicación.



Figura 6.2.4.3 –Ubicación

3. Se podrá extraer o quitar stock pinchando en el elegido.

**Menu ALMACEN - Operaciones por Ubicación**

Nº	PN	Div	Descripción	Stock	Ubica	Datecode	Lote/Pedido	Comentario	AGREGAR STOCK	EXTRAER STOCK
1	14011001	A	CABLE BLUE BOBINA 100M 32/0,2MM	0	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
2	14011002	A	CABLE BOBINA DE 100M 30 AWG NARANJA	1	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
3	14011003	A	HILO BOBINA DE 100M 26AWG RED	1	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
4	14011004	A	HILO BOBINA DE 100M 26 AWG WHITE	1	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
5	14011005	A	HILO BOBINA DE 100M 26AWG NEGRO	1	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
6	14011006	A	CABLE BOBINA DE 30M 26AWG ROJO	1	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
7	14011007	A	CABLE BOBINA DE 30M 18AWG AZUL	1	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
8	14011008	A	CABLE BOBINA DE 30M 18AWG RED	1	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
9	14011009	A	CABLE BOBINA DE 30M 18AWG BLACK	1	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
10	14011010	A	CABLE KIT AWG 24 ROJO	2	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
11	14011011	A	CABLE KIT AWG NEGRO	2	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
12	14011013	A	cable 30awg rojo (rollo de 100m)	1	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
13	14011014	A	cable 30awg azul (rollo de 100m)	1	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>
14	14011015	A	cable 30awg negro (rollo de 100m)	1	1C00000	s/r	s/r		<input type="radio"/>	<input type="radio"/>

Figura 6.2.4.4 – Componentes de dicha ubicación

4. Se tendrá que introducir la cantidad deseada a extraer o quitar.

Menu ALMACEN - Operaciones por Ubicación



N ENCONTRADO 19 COMPONENTES

PN	Div	Descripción	Stock	Ubica	Datecode	Lote/Pedido	Comentario	AGREGAR STOCK	EXTRAER STOCK
011001	A	CABLE BLUE BOBINA 100M 32/0,2MM	0	1C00000	s/r	s/r		•	•
011002	A	CABLE BOBINA DE 100M 30 AWG NARANJA	1	1C00000	s/r	s/r		•	•
011003	A	HILO BOBINA DE 100M 26AWG RED						•	•
011004	A	HILO BOBINA DE 100M 26 AWG WHITE						•	•
011005	A	HILO BOBINA DE 100M 26AWG NEGRO						•	•
011006	A	CABLE BOBINA DE 30M 26AWG ROJO						•	•
011007	A	CABLE BOBINA DE 30M 18AWG AZUL						•	•
011008	A	CABLE BOBINA DE 30M 18AWG RED	1	1C00000	s/r	s/r		•	•
011009	A	CABLE BOBINA DE 30M 18AWG BLACK	1	1C00000	s/r	s/r		•	•
011010	A	CABLE KIT AWG 24 ROJO	2	1C00000	s/r	s/r		•	•

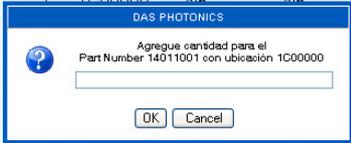


Figura 6.2.4.5 –Extraer o agregar stock

6.2.5 PRUEBA HACER UN KITTING

1. En el formulario se introducirá la lista de componentes que se desea extraer con sus respectivas cantidades.

Figura 6.2.5.1 – Nuevo kitting

2. Se mostrará una tabla con todas las ubicaciones posibles de todos los componentes introducidos en la lista para seleccionar entre ellas de cual se quiere extraer. Existirán componentes inicialmente que no se podrán extraer y otros que a elección del usuario puede dejarlos en pendiente por cualquier motivo.

### Kitting Nuevo



OF: OF10004  
 PART NUMBER: AS3FMRI Y Eulista  
 REFERENCIA KITTING: page3  
 NÚMERO DE COMPONENTES: 1

Nº	PN	Div	Descripción	Ubicación	Datecode	Lot#Pedido	Seleccionar Ubicación	Stock	Deag,base Pedido	Stock restante	Total pedido	Falta seleccionar	Kitting Pendiente
1	1411185	A	CONECTOR HE14 PCB RECIO 4H4	IB12400	sl-	sl-	<input checked="" type="checkbox"/>	12	2	10	2	0	<input type="checkbox"/>
2	1601102	A	TORNILLO ALLEN DIN 912 M2x4	IB11100	sl-	sl-	<input checked="" type="checkbox"/>	30	5	44	6	0	<input type="checkbox"/>
3	1601104	A	TORNILLO ALLEN DIN 912 M2x4	IDI1100	sl-	sl-	<input checked="" type="checkbox"/>	100	4	96	4	0	<input type="checkbox"/>
4	1601105	A	TORNILLO ALLEN DIN 912 M2x4	IB11100	sl-	sl-	<input checked="" type="checkbox"/>	91	12	78	12	0	<input type="checkbox"/>
5	110.0131	A	RESISTOR 1DR 0602	1A1211.	sl-	sl-	<input type="checkbox"/>	20	0	20	25	35	<input checked="" type="checkbox"/>
6	1601106	A	TORNILLO ALLEN DIN 912 M2x4	IB11100	sl-	sl-	<input checked="" type="checkbox"/>	30	1	49	1	0	<input type="checkbox"/>
7	1601107	A	TORNILLO ALLEN DIN 912 M2x4	IDI1100	sl-	sl-	<input checked="" type="checkbox"/>	100	5	95	5	0	<input type="checkbox"/>
8	1601108	A	TORNILLO ALLEN DIN 912 M2x4	IB11100	sl-	sl-	<input checked="" type="checkbox"/>	200	7	207	205	205	<input type="checkbox"/>
9	1601109	A	TORNILLO ALLEN DIN 912 M2x4	IB11100	sl-	sl-	<input checked="" type="checkbox"/>	107	5	101	6	0	<input type="checkbox"/>
10	1100006	A	resistor de carbono 5A	IB12400	sl-	sl-	<input type="checkbox"/>	10	1	10	51	51	<input type="checkbox"/>
11	1601101	A	TORNILLO ALLEN DIN 912 M2x4	IB11200	sl-	sl-	<input checked="" type="checkbox"/>	100	16	84	15	0	<input type="checkbox"/>
12	1601107	A	TORNILLO ALLEN DIN 912 M2x4	IDI1100	sl-	sl-	<input checked="" type="checkbox"/>	180	1	179	1	0	<input type="checkbox"/>
13	11920112	A	sensor de temperatura SPI	1A1213R	sl-	sl-	<input checked="" type="checkbox"/>	2	1	1	1	0	<input type="checkbox"/>
14	612.410	-	PCB control v1.0	NO CONSTA	NO CONSTA	NO CONSTA	<input type="checkbox"/>	0	0	0	28	28	<input type="checkbox"/>
15	61201410	-	PCBA Control	NO CONSTA	NO CONSTA	NO CONSTA	<input type="checkbox"/>	0	0	0	5	5	<input checked="" type="checkbox"/>

Figura 6.2.5.2 – Selección de ubicaciones

3. Se podrán hacer kittings parciales y volver a retomarlos en la siguiente pantalla.

### Kitting Pendientes



Nº	Kitting	Referencia Kitting	OF	Assembly Quantity	Part Number Assembly	Fecha	VER	FINALIZAR
1	30000001	61201102	OF10007	1	KITTING	01-05-2011	<input type="button" value="v"/>	<input type="button" value="v"/>

Figura 6.2.5.3 –Kittings pendientes

4. Seleccionamos el kitting correspondiente y actuamos de la misma forma que en el punto 2.

### Kitting Nuevo



OF: OF10004  
 PART NUMBER: AS3FMRI Y Eulista  
 REFERENCIA KITTING: page3  
 NÚMERO DE COMPONENTES: 1

Nº	PN	Div	Descripción	Ubicación	Datecode	Lot#Pedido	Seleccionar Ubicación	Stock	Deag,base Pedido	Stock restante	Total pedido	Falta seleccionar	Kitting Pendiente
1	1411185	A	CONECTOR HE14 PCB RECIO 4H4	IB12400	sl-	sl-	<input checked="" type="checkbox"/>	12	2	10	2	0	<input type="checkbox"/>
2	1601102	A	TORNILLO ALLEN DIN 912 M2x4	IB11100	sl-	sl-	<input checked="" type="checkbox"/>	30	5	44	6	0	<input type="checkbox"/>
3	1601104	A	TORNILLO ALLEN DIN 912 M2x4	IDI1100	sl-	sl-	<input checked="" type="checkbox"/>	100	4	96	4	0	<input type="checkbox"/>
4	1601105	A	TORNILLO ALLEN DIN 912 M2x4	IB11100	sl-	sl-	<input checked="" type="checkbox"/>	91	12	78	12	0	<input type="checkbox"/>
5	110.0131	A	RESISTOR 1DR 0602	1A1211.	sl-	sl-	<input type="checkbox"/>	20	0	20	25	35	<input checked="" type="checkbox"/>
6	1601106	A	TORNILLO ALLEN DIN 912 M2x4	IB11100	sl-	sl-	<input checked="" type="checkbox"/>	30	1	49	1	0	<input type="checkbox"/>
7	1601107	A	TORNILLO ALLEN DIN 912 M2x4	IDI1100	sl-	sl-	<input checked="" type="checkbox"/>	100	5	95	5	0	<input type="checkbox"/>
8	1601108	A	TORNILLO ALLEN DIN 912 M2x4	IB11100	sl-	sl-	<input checked="" type="checkbox"/>	200	7	207	205	205	<input type="checkbox"/>
9	1601109	A	TORNILLO ALLEN DIN 912 M2x4	IB11100	sl-	sl-	<input checked="" type="checkbox"/>	107	5	101	6	0	<input type="checkbox"/>
10	1100006	A	resistor de carbono 5A	IB12400	sl-	sl-	<input type="checkbox"/>	10	1	10	51	51	<input type="checkbox"/>
11	1601101	A	TORNILLO ALLEN DIN 912 M2x4	IB11200	sl-	sl-	<input checked="" type="checkbox"/>	100	16	84	15	0	<input type="checkbox"/>
12	1601107	A	TORNILLO ALLEN DIN 912 M2x4	IDI1100	sl-	sl-	<input checked="" type="checkbox"/>	180	1	179	1	0	<input type="checkbox"/>
13	11920112	A	sensor de temperatura SPI	1A1213R	sl-	sl-	<input checked="" type="checkbox"/>	2	1	1	1	0	<input type="checkbox"/>
14	612.410	-	PCB control v1.0	NO CONSTA	NO CONSTA	NO CONSTA	<input type="checkbox"/>	0	0	0	28	28	<input type="checkbox"/>
15	61201410	-	PCBA Control	NO CONSTA	NO CONSTA	NO CONSTA	<input type="checkbox"/>	0	0	0	5	5	<input checked="" type="checkbox"/>

Figura 6.2.5.4 –Kitting por finalizar

### 6.2.6 PRUEBA SOLICITUD DE COMPRA

1. Rellenamos la cabecera de nuestra solicitud. Pudiendo agregar cargos adicionales a la misma que consisten en componentes o cualquier cargo no contemplado ni codificado en la base de datos del sistema.

**DAS Photonics** Solicitud Compra  
Usuario: acastillo

Solicitante: acastillo Fecha Solicitud: 05-07-2011

Proveedor: BIHOMSA  N. 10029

A/a: Ventas Divisa: EUR

Fecha de entrega: 06/07/2011 Condiciones: DDP

Tipo de compra: Elija opcion

OF: OF110031 MFDC (Artes 5.1 Optical Multi-Frequency Conversion Unit)

Concepto: Artes 5.1 Optical Multi-Frequency Conversion Unit

Proyecto: MFAS MFDC

Observaciones:

Urgente  IPC  O.T.A.N.

Cargos adicionales

Cargo	Cantidad	Precio	Eliminar
<input type="text"/>	<input type="text"/>	<input type="text"/> €	<input type="checkbox"/>

Figura 6.2.6.1 – Cabecera de la solicitud

2. Posteriormente se introduce la codificación del componente y se selecciona que oferta se desea.

Solicitud Compra - Desglose



Nº	DAS Part Number	Descripcion	Referencia	Oferta
1	<input type="text" value="11010001"/>	R 1K8 5% 0603 REEL		<input checked="" type="radio"/>

Figura 6.2.6.2 – Introducción del componente

- Se pincha la oferta deseada y el sistema cargará los datos en la solicitud de compra. Si en algún caso los datos de la oferta no son los deseados se le avisará al usuario en el campo observaciones y éste podrá tomar la decisión de continuar o de modificar esos datos.



PN	Descripcion	Fabricante	Ref.Fabricante	Proveedor	Ref.Proveedor	Precio	MDQ	Factor	Lote	Leadtimeweek	Fecha ofertada	PRIORIDAD Proveedor	PRIORIDAD Oferta	Selección
11010001	R 1K8 5% 0603 REEL	BIOHMSA	--	BIHOMSA	CH0671801J	0	5000	1		99	14-05-2007	NO	NO	<input checked="" type="radio"/>

Figura 6.2.6.3 – Oferta seleccionada

Solicitud Compra - Desglose



DAS Part Number	Descripcion	Referencia	Oferta	Cantidad	Precio Unidad	Total sin IVA	Observaciones	Eliminar
<input type="text" value="11010001"/>	R 1K8 5% 0603 REEL	CH0671801J	<input checked="" type="radio"/>	<input type="text" value="50"/>	<input type="text" value="12"/> €	600€	"Cantidad minima(5000) y leadtime(99semanas) incorrectos"	<input checked="" type="radio"/>

Figura 6.2.6.4 –Fin de solicitud de compra

Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

4. Al finalizar el sistema hará un pdf con todos los datos introducidos y el usuario seleccionará en hacer la solicitud definitiva o parcial. Si es parcial podrá modificarla o reenviarla de nuevo para que sea definitiva y así poder firmarla o si es definitiva sólo faltarán las consiguientes firmas para poder ser un pedido estable y, en ese momento, por parte del sistema crear el pedido.



Ciudad Politécnica de la Innovación - Camino de Vera s/n Edificio 8F 2ª Planta 46022 Valencia - SPAIN - +34 963 556 150 fax +34 963 562 581 - www.dasphotonics.com

### Solicitud de Compra: CSA20110705-1

Proveedor:	BIOHMSA COMP.ELECTRÓNICOS	Nº 10028
A/a:	Ventas	
Fecha entrega:	06/07/2011	Condiciones de entrega: DDP

Solicitante:	Adrián Castillo	Fecha Solicitud:	Martes 05 de Julio de 2011		
Concepto:	Artes 5.1 Optical Multi-Frequency Conversion Unit		Orden de fabricación:	OF110031	
Urgente:	SI	IPC:	SI	Tipo de compra:	Inversión

#### Detalle del pedido

DAS PN	Descripción	Referencia	Cantidad	Precio unidad	Total
11010001	R 1K8 5% 0603 REEL	CH0671801J	50	12 €	600 €
11010003	R 51R 5% 0603 REEL	CH0675109J	5.000	2 €	10.000 €
--	pruebas	CARGO ADICIONAL	12	23 €	276 €
<b>Total</b>					<b>10.876 €</b>

Observaciones
---------------

Firmas (segun DASQ-GST-040)		
Solicitante	Director de compras	Aprobado por
Fdo:Adrián Castillo	Fdo: Sebastian Pantoja	Fdo: Javier Martí
Fecha:05/07/2011	Fecha:	Fecha:

Figura 6.2.6.5 –Archivo pdf

# 7 CONCLUSIONES

Para concluir puedo decir que dicho proyecto me ha ayudado mucho para tener más experiencia en aspectos importantes a la hora de afrontar un trabajo de este nivel de desarrollo. Tareas como la planificación, diseño, codificación, integración, pruebas y aspectos intuitivos y llamativos para garantizar el éxito de la aplicación. Ha sido un avance sobre la vida laboral después de la culminación de una carrera por la universidad con el colofón de este proyecto.

Otros aspectos más informáticos que he aprendido han sido a la organización de código y a una programación limpia para una mejor administración del código. Esto provocó una mejor y fácil resolución de problemas en la aplicación o adaptar de forma más rápida alguna mejora.

Dicha aplicación tiene un gran futuro a nivel empresarial puesto que durante todo el tiempo que ha transcurrido se ha ido haciendo cada vez más potente. Por ello como ya he indicado existen diversos puntos futuros que pueden hacer la aplicación aún si cabe más exitosa. Pienso también que la elección del diseño y las herramientas utilizadas han sido las correctas y las más concretas dentro del marco en el que se quería hacer este tipo de trabajo.

El resultado final como se puede apreciar me ha producido una satisfacción plena debido a que tiene un gran futuro de desarrollo y porque viendo la evolución de la aplicación en el tiempo se ha ido alejando un poco de los primeros conceptos establecidos para convertirse en un proyecto de mayor impacto dentro de la empresa donde he podido hacer el proyecto.

He aprendido a desenvolverme en un ámbito laboral, a tomar decisiones importantes con las personas responsables sobre mi trabajo y a tomar iniciativas propias con ideas nuevas. Agradezco a la empresa DAS PHOTONICS S.L por la oportunidad que me ha brindado y al grupo humano que trabaja en ella. Espero que se siga trabajando sobre un proyecto que se apostó fuerte y que no sea esto un punto final sino un punto y seguido.

# BIBLIOGRAFÍA

## • ERS:

- [1] Wikipedia en español: <http://es.wikipedia.org>
- [2] Manual de PHP: <http://www.php.net>
- [3] Especificaciones de HTML 1.0: <http://www.w3schools.com/html/default.asp>
- [4] Documentación de Apache: <http://httpd.apache.org>

## • HTML:

- [1] Tutorial W3 Schools: [www.w3schools.com/html/](http://www.w3schools.com/html/)
- [2] Manual HTML: [www.webestilo.com/html/](http://www.webestilo.com/html/)
- [3] Especificaciones de HTML 1.0: <http://www.w3schools.com/html/default.asp>

## • CSS:

- [1] Guía breve de CSS: [www.w3c.es/divulgacion/guiasbreves/hojasestilo](http://www.w3c.es/divulgacion/guiasbreves/hojasestilo)
- [2] Tutorial CSS: [www.w3schools.com/css/](http://www.w3schools.com/css/)

## • PHP:

- [1] Manual de PHP: <http://www.php.net/>
- [2] PHP a fondo: <http://www.desarrolloweb.com/php/>

## • MySQL:

- [1] Documentación oficial de MySQL: [www.mysql.com](http://www.mysql.com)
- [2] Manual de MySQL: [www.webestilo.com/mysql/](http://www.webestilo.com/mysql/)

## • Apache:

- [1] Documentación de Apache: <http://httpd.apache.org>.

# ANEXO A. DESCRIPCIÓN DE TECNOLOGÍAS

## A.1. HTTP

Es el protocolo usado en cada transacción de la World Wide Web. HTTP fue desarrollado por el World Wide Web Consortium y la Internet Engineering Task Force, colaboración que culminó en 1999 con la publicación de una serie de RFC, el más importante de ellos es el RFC 2616 que especifica la versión 1.1. HTTP define la sintaxis y la semántica que utilizan los elementos de software de la arquitectura web (clientes, servidores, proxies) para comunicarse. Es un protocolo orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor. Al cliente que efectúa la petición (un navegador web o un spider) se lo conoce como "user agent" (agente del usuario). A la información transmitida se la llama recurso y se la identifica mediante un localizador uniforme de recursos(URL). Los recursos pueden ser archivos, el resultado de la ejecución de un programa, una consulta a una base de datos, la traducción automática de un documento, etc.

HTTP es un protocolo sin estado, es decir, que no guarda ninguna información sobre conexiones anteriores. El desarrollo de aplicaciones web necesita frecuentemente mantener estado. Para esto se usan las cookies, que es información que un servidor puede almacenar en el sistema cliente. Esto le permite a las aplicaciones web instituir la noción de "sesión", y también permite rastrear usuarios ya que las cookies pueden guardarse en el cliente por tiempo indeterminado.

Una transacción HTTP está formada por un encabezado seguido, opcionalmente, por una línea en blanco y algún dato. El encabezado especificará cosas como la acción requerida del servidor, o el tipo de dato retornado, o el código de estado.

El uso de campos de encabezados enviados en las transacciones HTTP le da gran flexibilidad al protocolo. Estos campos permiten que se envíe información descriptiva en la transacción, permitiendo así la autenticación, cifrado e identificación de usuario.

Un encabezado es un bloque de datos que precede a la información propiamente dicha, por lo que muchas veces se hace referencia a él como metadato —porque tiene datos sobre los datos—.

Si se reciben líneas de encabezado del cliente, el servidor las coloca en las variables de ambiente de CGI con el prefijo HTTP\_ seguido del nombre del encabezado. Cualquier carácter guión ( - ) del nombre del encabezado se convierte a caracteres "\_".

El servidor puede excluir cualquier encabezado que ya esté procesado, como Authorization, Content-type y Content-length. El servidor puede elegir excluir alguno o todos los encabezados si incluirlos exceden algún límite del ambiente de sistema. Ejemplos de esto son las variables HTP\_ACPT y HTP\_R\_AEN.

- HT\_AEP'. Los tipos MIME que el cliente aceptará, dado los encabezados HTTP. Otros protocolos quizás necesiten obtener esta información de otro lugar. Los elementos de esta lista deben estar separados por una coma, como lo dice la especificación HTTP: tipo, tipo.
- HTT\_USR\_AET. El navegador que utiliza el cliente para realizar la petición. El formato general para esta variable es: software/versión biblioteca/versión.

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

HTTP ha pasado por múltiples versiones del protocolo, muchas de las cuales son compatibles con las anteriores. El RFC 2145 describe el uso de los números de versión de HTTP. El cliente le dice al servidor al principio de la petición la versión que usa, y el servidor usa la misma o una anterior en su respuesta.

HTTP/0.9 Obsoleta. Soporta sólo un comando, GET, y además no especifica el número de versión HTTP. No soporta cabeceras. Como esta versión no soporta POST, el cliente no puede enviarle mucha información al servidor.

HTTP/1.0 (mayo de 1996) Esta es la primera revisión del protocolo que especifica su versión en las comunicaciones, y todavía se usa ampliamente, sobre todo en servidores proxy.

HTTP/1.1 (junio de 1999) Versión actual; las conexiones persistentes están activadas por defecto y funcionan bien con los proxies. También permite al cliente enviar múltiples peticiones a la vez (pipelining) lo que hace posible eliminar el tiempo de Round-Trip delay por cada petición.

HTTP/1.2 Los primeros borradores de 1995 del documento PEP — an Extension Mechanism for HTTP (el cuál propone el Protocolo de Extensión de Protocolo, abreviado PEP) los hizo el World Wide Web Consortium y se envió al Internet Engineering Task Force. El PEP inicialmente estaba destinado a convertirse en un rango distintivo de HTTP/1.2. En borradores posteriores, sin embargo, se eliminó la referencia a HTTP/1.2. El RFC 2774 (experimental), HTTP Extension Framework, incluye en gran medida a PEP.

## A.2. PHP

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente para la interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

Fue creado originalmente por Rasmus Lerdorf en 1994; sin embargo la implementación principal de PHP es producida ahora por The PHP Group y sirve como el estándar de facto para PHP al no haber una especificación formal. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre.

Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. El lenguaje PHP se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores, el número de sitios en PHP ha compartido algo de su preponderante sitio con otros nuevos lenguajes no tan poderosos desde agosto de 2005. Es también el módulo Apache más popular, el cliente hace una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente. Mediante extensiones es también posible la generación de archivos PDF, Flash, así como imágenes en diferentes formatos.

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite.

XAMPP es un servidor independiente de plataforma, software libre, que consiste principalmente en la base de datos MySQL, el servidor Web Apache y los intérpretes para lenguajes de script: PHP y Perl. El nombre proviene del acrónimo de X (para cualquiera de los diferentes sistemas operativos), Apache, MySQL, PHP, Perl. El programa está liberado bajo la licencia GNU y actúa como un servidor Web libre, fácil de usar y capaz de interpretar páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, GNU/Linux, Solaris, y MacOS X.

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como Unix (y de ese tipo, como Linux o Mac OS X) y Microsoft Windows, y puede interactuar con los servidores de web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.

PHP es una alternativa a las tecnologías de Microsoft ASP y ASP.NET (que utiliza C# y Visual Basic .NET como lenguajes), a ColdFusion de la empresa Adobe, a JSP/Java y a CGI/Perl. Aunque su creación y desarrollo se da en el ámbito de los sistemas libres, bajo la licencia GNU, existe además un entorno de desarrollo integrado comercial llamado Zend Studio. Recientemente, CodeGear (la división de lenguajes de programación de Borland) ha sacado al mercado un entorno de desarrollo integrado para PHP, denominado 'Delphi for PHP. También existen al menos un par de módulos para Eclipse, uno de los entornos más populares.

PHP es un acrónimo recursivo que significa PHP Hypertext Pre-processor (inicialmente PHP Tools, o, Personal Home Page Tools). Fue creado originalmente por Rasmus Lerdorf en 1994; sin embargo la implementación principal de PHP es producida ahora por The PHP Group y sirve como el estándar de facto para PHP al no haber una especificación formal. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre.

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores crear aplicaciones complejas con una curva de aprendizaje muy corta. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones.

Aunque todo en su diseño está orientado a facilitar la creación de sitios webs, es posible crear aplicaciones con una interfaz gráfica para el usuario, utilizando la extensión PHP-Qt o PHP-GTK. También puede ser usado desde la línea de órdenes, de la misma manera como Perl o Python pueden hacerlo; a esta versión de PHP se la llama PHP-CLI (Command Line Interface).

Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite.

#### **Ventajas**

- Es un lenguaje multiplataforma.

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

- Orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
  - El código fuente escrito en PHP es invisible al navegador web y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
  - Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
  - Capacidad de expandir su potencial utilizando módulos (llamados ext's o extensiones).
  - Posee una amplia documentación en su sitio web oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
  - Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
  - Permite aplicar técnicas de programación orientada a objetos.
  - Biblioteca nativa de funciones sumamente amplia e incluida.
  - No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
  - Tiene manejo de excepciones (desde PHP5).
  - Si bien PHP no obliga a quien lo usa a seguir una determinada metodología a la hora de programar (muchos otros lenguajes tampoco lo hacen), aun haciéndolo, el programador puede aplicar en su trabajo cualquier técnica de programación o de desarrollo que le permita escribir código ordenado, estructurado y manejable. Un ejemplo de esto son los desarrollos que en PHP se han hecho del patrón de diseño Modelo Vista Controlador (MVC), que permiten separar el tratamiento y acceso a los datos, la lógica de control y la interfaz de usuario en tres componentes independientes.

#### Inconvenientes

- Como es un lenguaje que se interpreta en ejecución, para ciertos usos puede resultar un inconveniente que el código fuente no pueda ser ocultado. La ofuscación es una técnica que puede dificultar la lectura del código pero no la impide y, en ciertos casos, representa un costo en tiempos de ejecución.

### A.3. CSS

El nombre hojas de estilo en cascada viene del inglés Cascading Style Sheets, del que toma sus siglas. CSS es un lenguaje 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.

La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación

#### Los tres tipos de estilos

CSS proporciona tres caminos diferentes para aplicar las reglas de estilo a una página Web:

1. Una hoja de estilo externa, es una hoja de estilo que está almacenada en un archivo diferente al archivo donde se almacena el código HTML de la página Web. Esta es la manera de programar más potente, porque separa completamente las reglas de formateo para la página HTML de la estructura básica de la página:

2. Una hoja de estilo interna, que es una hoja de estilo que está incrustada dentro de un documento HTML. (Va a la derecha dentro del elemento <head>.) De esta manera se obtiene el beneficio de separar la información del estilo del código HTML propiamente dicho. Se puede optar por copiar la hoja de estilo incrustada de una página a otra (esta posibilidad es difícil de ejecutar si se desea para guardar las copias sincronizadas). En general, la única vez que se usa una hoja de estilo interna, es cuando se quiere proporcionar alguna característica a una página Web en un simple fichero, por ejemplo, si se está enviando algo a la página Web.

3. Un estilo en línea (inline) es un método para insertar el lenguaje de estilo de página directamente dentro de una etiqueta HTML. Esta manera de proceder no es totalmente adecuada. El incrustar la descripción del formateo dentro del documento de la página Web, a nivel de código, se convierte en una manera larga, tediosa y poco elegante de resolver el problema de la programación de la página. Este modo de trabajo se podría usar de manera ocasional si se pretende aplicar un formateo con prisa, al vuelo. No es todo lo claro o estructurado que debería ser, pero funciona. Éste es el método recomendado para maquetar correos electrónicos en HTML.

#### **Ventajas de usar las hojas de estilo**

Las ventajas de utilizar CSS (u otro lenguaje de estilo) son:

- Control centralizado de la presentación de un sitio web completo con lo que se agiliza de forma considerable la actualización del mismo.
- Los navegadores permiten a los usuarios especificar su propia hoja de estilo local, que será aplicada a un sitio web, con lo que aumenta considerablemente la accesibilidad. Por ejemplo, personas con deficiencias visuales pueden configurar su propia hoja de estilo para aumentar el tamaño del texto o remarcar más los enlaces.
- Una página puede disponer de diferentes hojas de estilo según el dispositivo que la muestre o, incluso, a elección del usuario. Por ejemplo, para ser impresa, mostrada en un dispositivo móvil o ser "leída" por un sintetizador de voz.
- El documento HTML en sí mismo es más claro de entender y se consigue reducir considerablemente su tamaño (siempre y cuando no se utilice estilo en línea).

## **A.4. LINUX**

GNU/Linux es uno de los términos empleados para referirse a la combinación del núcleo o kernel libre similar a Unix denominado Linux, que es usado con herramientas de sistema GNU. Su desarrollo es uno de los ejemplos más prominentes de software libre; todo su código fuente puede ser utilizado, modificado y redistribuido libremente por cualquiera bajo los términos de la GPL.

La colección de utilidades para la programación de GNU es con diferencia la familia de compiladores más utilizada en este sistema operativo. Tiene capacidad para compilar C, C++, Java, Ada, entre otros muchos

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

lenguajes. Además soporta diversas arquitecturas mediante la compilación cruzada, lo que hace que sea un entorno adecuado para desarrollos heterogéneos.

Hay varios entornos de desarrollo integrados disponibles para GNU/Linux incluyendo, Anjuta, KDevelop, Ultimate++, Code::Blocks, NetBeans IDE y Eclipse. También existen editores extensibles como Emacs o Vim. GNU/Linux también dispone de capacidades para lenguajes de guión (script), aparte de los clásicos lenguajes de programación de shell, o el de procesamiento de textos por patrones y expresiones regulares llamado awk, la mayoría de las distribuciones tienen instalado Python, Perl, PHP y Ruby.

Las aplicaciones para Linux se distribuyen principalmente en los formatos.deb y.rpm, los cuales fueron creados por los desarrolladores de Debian y Red Hat respectivamente. También existe la posibilidad de instalar aplicaciones a partir de código fuente en todas las distribuciones.

Durante la etapa temprana había pocas aplicaciones de código cerrado para GNU/Linux. Con el tiempo se fueron portando programas no libres al sistema GNU/Linux, entre ellos Adobe Reader, Adobe Flash, Google Picasa, Opera, entre otros.

La creciente popularidad de GNU/Linux se debe, entre otras razones, a su estabilidad, al acceso al código fuente (lo que permite personalizar el funcionamiento y auditar la seguridad y privacidad de los datos tratados), a la independencia de proveedor, a la seguridad, a la rapidez con que incorpora los nuevos adelantos tecnológicos (IPv6, microprocesadores de 64 bits), a la escalabilidad (se pueden crear clusters de cientos de computadoras), a la activa comunidad de desarrollo que hay a su alrededor, a su interoperabilidad y a la abundancia de documentación relativa a los procedimientos. IBM Roadrunner, la supercomputadora más potente de 2008, funciona bajo una distribución GNU/Linux

Hay varias empresas que comercializan soluciones basadas en GNU/Linux: IBM, Novell (SuSE), Red Hat (RHEL), Mandriva (Mandriva Linux), Rxtart, Canonical Ltd. (Ubuntu), así como miles de PYMES que ofrecen productos o servicios basados en esta tecnología.

## A.5. HTML

El HTML, Hyper Text Markup Language (Lenguaje de marcación de Hipertexto) es el lenguaje de marcas de texto utilizado normalmente en la www (World Wide Web). Fue creado en 1986 por el físico nuclear Tim Berners-Lee; el cual tomó dos herramientas preexistentes: El concepto de Hipertexto (Conocido también como link o ancla) el cual permite conectar dos elementos entre sí y el SGML (Lenguaje Estándar de Marcación General) el cual sirve para colocar etiquetas o marcas en un texto que indique como debe verse. HTML no es propiamente un lenguaje de programación como C++, Visual Basic, etc., sino un sistema de etiquetas. HTML no presenta ningún compilador, por lo tanto algún error de sintaxis que se presente éste no lo detectará y se visualizará en la forma como éste lo entienda.

El entorno para trabajar HTML es simplemente un procesador de texto, como el que ofrecen los sistemas operativos Windows (Bloc de notas), UNIX (el editor vi o ed) o el que ofrece MS Office (Word). El conjunto de etiquetas que se creen, se deben guardar con la extensión .htm o .html.

Los elementos son la estructura básica de HTML. Los elementos tienen dos propiedades básicas: atributos y contenido. Cada atributo y contenido tiene ciertas restricciones para que se considere válido al documento HTML. Un elemento generalmente tiene una etiqueta de inicio (p.ej. <nombre-de-elemento>) y una etiqueta de cierre (p.ej. </nombre-de-elemento>).

Los atributos del elemento están contenidos en la etiqueta de inicio y el contenido está ubicado entre las dos etiquetas (p.ej. <nombre-de-elemento atributo="valor">Contenido</nombre-de-elemento>). Algunos elementos, tales como <br>, no tienen contenido ni llevan una etiqueta de cierre. Debajo se listan varios tipos de elementos de marcado usados en HTML.

El marcado estructural describe el propósito del texto. El marcado estructural no define cómo se verá el elemento, pero la mayoría de los navegadores web han estandarizado el formato de los elementos. Un formato específico puede ser aplicado al texto por medio de hojas de estilo en cascada.

El marcado presentacional describe la apariencia del texto, sin importar su función. La mayoría del marcado presentacional ha sido desechada con HTML 4.0, en favor de hojas de estilo en cascada.

El marcado hipertextual se utiliza para enlazar partes del documento con otros documentos o con otras partes del mismo documento. Para crear un enlace es necesario utilizar la etiqueta de ancla <a> junto con el atributo href, que establecerá la dirección URL a la que apunta el enlace.

#### **Atributos**

La mayoría de los atributos de un elemento son pares nombre-valor, separados por un signo de igual «=» y escritos en la etiqueta de comienzo de un elemento, después del nombre de éste. El valor puede estar rodeado por comillas dobles o simples, aunque ciertos tipos de valores pueden estar sin comillas en HTML.

#### **Nociones básicas de HTML**

El lenguaje HTML puede ser creado y editado con cualquier editor de textos básico, como puede ser Gedit en Linux, el Bloc de notas de Windows, o cualquier otro editor que admita texto sin formato como GNU Emacs, Microsoft Wordpad, TextPad, Vim, Notepad++, entre otros.

Existen además, otros editores para la realización de sitios web con características WYSIWYG (What You See Is What You Get, o en español: «lo que ves es lo que obtienes»). Estos editores permiten ver el resultado de lo que se está editando en tiempo real, a medida que se va desarrollando el documento. Ahora bien, esto no significa una manera distinta de realizar sitios web, sino que es una forma un tanto más simple ya que estos programas, además de tener la opción de trabajar con la vista preliminar, tienen su propia sección HTML la cual va generando todo el código a medida que se va trabajando. Algunos ejemplos de editores WYSIWYG son KompoZer, Microsoft FrontPage, o Macromedia Dreamweaver.

Combinar estos dos métodos resulta muy interesante, ya que de alguna manera se ayudan entre sí. Por ejemplo; si se edita todo en HTML y de pronto se olvida algún código o etiqueta, simplemente se dirige al editor visual o WYSIWYG y se continúa ahí la edición, o viceversa, ya que hay casos en que sale más rápido y fácil escribir directamente el código de alguna característica que queramos adherirle al sitio, que buscar la opción en el programa mismo.

Existe otro tipo de editores HTML llamados WYSIWYM que dan más importancia al contenido y al significado que a la apariencia visual. Entre los objetivos que tienen estos editores es la separación del contenido y la presentación, fundamental en el diseño web.

HTML utiliza etiquetas o marcas, que consisten en breves instrucciones de comienzo y final, mediante las cuales determinan la forma en la que debe aparecer en su navegador el texto, así como también las imágenes y los demás elementos, en la pantalla del ordenador.

#### **Accesibilidad web**

El diseño en HTML aparte de cumplir con las especificaciones propias del lenguaje debe respetar unos criterios de accesibilidad web, siguiendo unas pautas, o las normativas y leyes vigentes en los países donde se regule dicho concepto. Se encuentra disponible y desarrollado por el W3C a través de las Pautas de Accesibilidad al Contenido Web 1.0 WCAG

Sin HTML no existiría el Internet que hoy conocemos, si no se hubiera inventado, seguiríamos con el aburrido FTP (Protocolo de Transferencia de Ficheros) mandándonos o bajando archivos que a veces ni sabíamos lo que eran, sin tener la posibilidad de visualizarlo antes. Ahora por medio de una página llena de colores y botones por la cual se puede navegar, se puede tener acceso a bastante información y archivos. El lenguaje de HTML abrió una puerta al mundo permitiéndoles a las personas expresar sus ideas por medio de páginas y mostrárselas a todas las personas de todos los países.

Con el HTML se logró un gran movimiento económico ya que muchísimas empresas publican, venden, y ofrecen sus productos, sus servicios y sus ofertas atrayendo a mayor cantidad de personas. También gracias al HTML nacieron muchas empresas que ofrecen diversos servicios como Yahoo, Altavista, HotMail, Terra, Yupi, Mercadolibre, De Remate, etc.

Infinitas son las posibilidades que te brindan las páginas WEB ya que no solo te dan la posibilidad de pasar el tiempo navegando, sino que también hasta se puede comprar un auto por Internet, solo basta con llenar un formulario con los datos personales y el número de tarjeta de crédito y en ocho días lo tenemos en nuestra casa.

Todo el universo de Internet se lo debemos al HTML, ya que todas las páginas con las que se compone la World Wide Web están hechas con el lenguaje de programación HTML.

## **A.6. SERVIDOR HTTP APACHE**

El servidor HTTP Apache es un servidor web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. Su nombre se debe a que Behelendorf quería que tuviese la connotación de algo que es firme y enérgico pero no agresivo, y la tribu Apache fue la última en rendirse al que pronto se convertiría en gobierno de EEUU, y en esos momentos la preocupación de su grupo

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

era que llegasen las empresas y "civilizasen" el paisaje que habían creado los primeros ingenieros de internet. Además Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. Era, en inglés, a patchy server (un servidor "parcheado").

El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: desde 1996, Apache, es el servidor HTTP más usado. Alcanzó su máxima cuota de mercado en 2005 siendo el servidor empleado en el 70% de los sitios web en el mundo, sin embargo ha sufrido un descenso en su cuota de mercado en los últimos años. (Estadísticas históricas y de uso diario proporcionadas por Netcraft<sup>3</sup>).

La mayoría de las vulnerabilidades de la seguridad descubiertas y resueltas tan sólo pueden ser aprovechadas por usuarios locales y no remotamente. Sin embargo, algunas se pueden accionar remotamente en ciertas situaciones, o explotar por los usuarios locales malévolos en las disposiciones de recibimiento compartidas que utilizan PHP como módulo de Apache.

#### Ventajas

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

#### Uso

Apache 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 web.

Apache es el componente de servidor web en la popular plataforma de aplicaciones LAMP, junto a MySQL y los lenguajes de programación PHP/Perl/Python (y ahora también Ruby).

Este servidor web es redistribuido como parte de varios paquetes propietarios de software, incluyendo la base de datos Oracle y el IBM WebSphere application server. Mac OS X integra apache como parte de su propio servidor web y como soporte de su servidor de aplicaciones WebObjects. Es soportado de alguna manera por Borland en las herramientas de desarrollo Kylix y Delphi. Apache es incluido con Novell NetWare 6.5, donde es el servidor web por defecto, y en muchas distribuciones Linux.

### Diseño y desarrollo de base de datos en MySQL y aplicación Web en PHP con servidor central APACHE

Apache es usado para muchas otras tareas donde el contenido necesita ser puesto a disposición en una forma segura y confiable. Un ejemplo es al momento de compartir archivos desde una computadora personal hacia Internet.

Los programadores de aplicaciones web a veces utilizan una versión local de Apache con el fin de previsualizar y probar código mientras éste es desarrollado.

Microsoft Internet Information Services (IIS) es el principal competidor de Apache, así como Sun Java System Web Server de Sun Microsystems y un anfitrión de otras aplicaciones como Zeus Web Server. Algunos de los más grandes sitios web del mundo están ejecutándose sobre Apache. La capa frontal (front end) del motor de búsqueda.

## A.7. MYSQL

MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con 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 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

### **Aplicaciones**

MySQL es muy utilizado en aplicaciones web, como Drupal o phpBB, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones. Sea cual sea el entorno en el que va a utilizar MySQL, es importante adelantar monitoreos sobre el desempeño para detectar y corregir errores tanto de SQL como de programación.