



Diseño y desarrollo de una base de datos SQL y aplicación web para la gestión.

Yassín Said Esteller

Tutor: José Enrique López Patiño

Trabajo Fin de Grado presentado en la Escuela Técnica Superior de Ingenieros de Telecomunicación de la Universitat Politècnica de València, para la obtención del Título de Graduado en Ingeniería de Tecnologías y Servicios de Telecomunicación

Curso 2015-16

Valencia, 29 de junio de 2016

Agradecimientos

Agradezco a todos aquellos familiares, amigos pareja y profesores que confiaron en mí y me ayudaron a seguir adelante. Con cariño para todos aquellos que alguna vez estuvieron a mi lado y a los que se fueron como a mi admirado abuelo y a mi profesora de física que tanto me enseñó en primero de carrera Elvira Bonet Salom.

Resumen

Este trabajo fin de grado se basa en el desarrollo e implementación de una aplicación web que gestione una base de datos desarrollada para la organización de los servicios que oferta la empresa TRANSOFI S.L. Como marco de trabajo se ha utilizado toda una serie de herramientas de software libre junto con una infraestructura LAMP para el servidor. Todo el desarrollo de la aplicación e instalación de la infraestructura ha sido realizado por el alumno por lo que muchas herramientas desconocidas tuvieron que ser estudiadas para formar la posible solución. La plataforma de desarrollo utilizada se basa en lenguajes de programación como: HTML5, CSS3, JavaScript, PHP5 y SQL.

Distintos aspectos han sido cubiertos a lo largo del proceso de realización del proyecto. Las habilidades en distintas plataformas de programación y desarrollo no fueron lo único a destacar. Se debe resaltar también el trato con los clientes por parte del alumno también fue necesario en las últimas fases del proyecto.

En resumen, la memoria presente, desarrolla paso a paso todos los procesos y toma de decisiones para la realización de este diseño.

Resum

Aquest treball fi de grau està basat en el desenvolupament junt amb la implementació d'una aplicació web per a la gestió d'una base de dades desenvolupada per l'organització dels serveis que oferta l'empresa TRANSOFI S.L. El marc de treball que s'ha emprat consta d'una sèrie de ferramentes de software lliure junt amb la infraestructura LAMP per al servidor. Tot el desenvolupament de l'aplicació i la instal·lació de la infraestructura ha sigut realitzada per l'alumne resultant això en la necessitat estudiar diferents possibles maneres de solucionar el problema. La plataforma de desenvolupament està basada en l'ús de llenguatges de programació com: HTML5, CSS3, JavaScript, PHP5 i SQL.

Diferents aspectes han sigut coberts al llarg del procés de realització del projecte. Les diferents habilitats en diferents plataformes de programació i el desenvolupament tècnic no són l'únic a destacar. Cal remarcar també el tracte amb els clients per part de l'alumne en les fases finals del projecte per adequar l'aplicació web a les necessitats del client final.

En resum la memòria aquí present, desenvolupa pas a pas tots els processos i presa de decisions per a la realització final del disseny.

Abstract

This final project degree is based on the development and the implementation of a web application with an sql data base also developed for the Transofi S.L corporation. The framework used is a list of free software tools with a LAMP infrastructure. All the development and installation of the server infrastructure was performed by the student so therefore a lot of different ways to solve the problem had been studied. The web application was developed in different programming languages as: HTML5, CSS3, JavaScript, PHP5 and SQL

Many aspects have been covered in the process of realization of the project. The different skills and different programming languages and the technical development are not the only aspect to remark. Should also be noted the meetings with clients performed by the student in the final stages of the project to improve the web application.

In summary this project develops step by step all the process and decisions to make the final design.

Índice

Capítulo 1. Introducción.....	5
1.1 Motivación.....	5
1.2 Escenario.....	6
Capítulo 2. Objetivos.....	8
Capítulo 3. Metodología de trabajo.....	9
3.1 Gestión del proyecto.....	9
3.2 Distribución de tareas.....	10
3.3 Diagrama temporal.....	11
Capítulo 4. Instalación del servidor e infraestructura de trabajo.....	12
4.1 Contexto tecnológico.....	12
4.2 Instalación de Debian.....	15
4.3 Instalación de SSH.....	19
4.3.1 SSH.....	19
4.3.2 Configuración de SSH.....	19
4.4 Infraestructura LAMP. Linux Apache, MySQL y PHP.....	20
4.4.1 Apache.....	21
4.4.2 PHP.....	22
4.4.3 MySQL y PhpMyAdmin.....	22
4.5 Instalación del servidor LAMP en Debian.....	24
4.5.1 Instalación de MySQL.....	24
4.5.2 Instalación de Apache.....	24
4.5.3 Instalación de PHP.....	26
Capítulo 5. Instalación de componentes para el diseño.....	28
5.1 Cliente FTP.....	28
5.1.1 FileZilla.....	28
5.2 Cliente SSH de terminal remoto.....	29
5.2.1 Putty.....	29
5.3 Editor de texto.....	30
5.3.1 Sublime Text 2.....	31
5.4 Herramientas para el diseño de bases de datos.....	32
5.4.1 MySQL Workbench.....	32

Capítulo 6. Diseño y desarrollo.....	33
6.1 Diseño de la base de datos.....	33
6.2 Etiquetas.....	33
6.3 Tablas de la base datos.....	35
6.3.1 Empresa, departamento, mobiliario y archivo.....	35
6.3.2 Empresa, departamento, empleado y atributos del empleado.....	36
6.3.3 Archivo, tipos de archivo, caja y trazabilidad.....	38
6.3.4 Configuración de la etiqueta, unión de la etiqueta con el campo, configuración del campo, tipo, valor del campo select y fila.....	39
6.3.5 Orden de trabajo de mobiliario, órdenes de trabajo, asociación de la orden de trabajo y orden de proceso.....	41
6.3.6 Modelo relacional de la base de datos.....	41
6.4 Desarrollo web.....	43
6.4.1 Estructura del árbol de directorios y funcionamiento inicial.....	43
6.4.2 Módulos php.....	46
6.4.3 Configuración del servidor apache.....	63
6.4.4 Migración a un servidor web comercial.....	64
6.4.5 Contratación de un servidor web compartido y migración de la página.....	65
Capítulo 7. Conclusiones y líneas futuras.....	66

Índice de figuras

Figura 1. Interconexión mundial.....	6
Figura 2. Planificación Temporal.....	11
Figura 3. Gráfica uso de distribuciones en servidores Linux.....	12
Figura 4. Debian. (Extraído de mysecureshell.com.....	14
Figura 5. Descargar Debian.....	15
Figura 6. Descarga del sistema operativo Debian.....	16
Figura 7. Menú de instalación de Debian.....	17
Figura 8. Particionado de discos.....	18
Figura 9. Infraestructura LAMP.....	21
Figura 10. Apache software.....	22
Figura 11. Php.....	22
Figura 12. MySQL y PhpMyAdmin.....	23
Figura 13. Pájaro Bici.....	25
Figura 14. Php info.....	27

Figura 15. FileZilla.....	29
Figura 16. Putty.....	30
Figura 17. Sublime Text 2.....	31
Figura 18. MySQL Workbench.....	32
Figura 19. Etiqueta	34
Figura 20. Empresa, departamento, mobiliario y archivo.	35
Figura 21.1.1.1 Empresa, departamento, empleado y atributos del empleado.	36
Figura 22. Tablas archivo, tipos de archivo, caja y trazabilidad.	38
Figura 23. 6.3.4 Configuración de la etiqueta, unión de la etiqueta con el campo, configuración del campo, tipo, valor del campo select y fila.....	39
Figura 24. Orden de trabajo de mobiliario, órdenes de trabajo, asociación de la orden de trabajo y orden de proceso.	41
Figura 25. Diseño de la base de datos.	42
Figura 26. Estructura de directorios.	44
Figura 27. Flujograma de peticiones http.....	45
Figura 28. Directorio public.	46
Figura 29. Inicio de sesión.	47
Figura 30. Logout.....	47
Figura 31. Panel de opciones.....	48
Figura 32. Generación de etiqueta para archivo.....	48
Figura 33. Etiqueta generada correctamente.	49
Figura 34. Creación de la orden de recogida.....	50
Figura 35. Creación de orden de recuperación.....	51
Figura 36. Sugerencias y comentarios.....	52
Figura 37. Menú opciones del usuario supervisor.....	52
Figura 38. Gestión de usuarios.	53
Figura 39. Editar cuenta de usuario.....	53
Figura 40. Gestión de los departamentos.	54
Figura 41. Crear departamento.....	54
Figura 42. Editor de etiquetas.	55
Figura 43. Listado órdenes.....	56
Figura 44. Órdenes libres.	56
Figura 45. Menú de usuario del administrador de TRANSOFI S.L.....	57
Figura 46. Creación de una nueva empresa.....	58
Figura 47. Administrar las empresas clientes.....	58
Figura 48. Panel de opciones durante la administración de empresas.	59
Figura 49. Trazabilidad.	59
Figura 50. Crear caja.	60

Figura 51. Devolver archivo a su caja.....	61
Figura 52. Mover caja.	61
Figura 53. Alta mobiliario.....	62
Figura 54. Destrucción.....	62
Figura 55. Contratación del servidor web.	65

Capítulo 1. Introducción.

En este capítulo se introducirán las motivaciones que han llevado al desarrollo de este proyecto y a la implantación del mismo para fines empresariales.

1.1 Motivación.

Este proyecto nace debido a la necesidad de la empresa Transofi S.L de ofrecer una aplicación de gestión web para ofrecer un servicio más rápido y de calidad a sus clientes. De este modo, gracias a la aplicación se facilita la administración de los recursos ofertados tanto por parte de los clientes como de la empresa que provee el servicio.

El desarrollo de servicios web, tanto como agregado al modelo de negocio o directamente como plataforma para ofertar servicios, ha tomado fuerza desde hace ya un tiempo. Muestra de ello es la gran cantidad de plataformas que han surgido que facilitan que todo tipo de personas puedan ser creadoras de contenido para programación o diseño web. Este ámbito no solo se limita a páginas web donde encontramos negocios online sino que también nos encontramos con gran cantidad de perfiles en redes sociales a fin de realizar marketing hacia el usuario y también conseguir su *feedback*.

Dentro de un mundo interconectado se hace necesario el uso de tecnologías para llegar al mayor número de clientes. El apoyo en este tipo de plataformas se ha extendido de forma y manera que incluso negocios que antes eran impensables se han hecho ahora posibles. Como ejemplo tenemos diversas plataformas de venta que basan su negocio en una aplicación web prescindiendo de grandes almacenes o largas listas de productos en stock. Para cubrir las necesidades de los clientes y ahorrar en espacio se hace en primera instancia el pedido a un tercero, y posteriormente se procede a la entrega del producto al cliente. De este modo teniendo un pequeño almacén y un stock reducido basado en lo que los clientes más demandan. De este modo se facilita el trabajo y se produce un ahorro más que sustancial.

No solo en casos como estos de distribución de productos ha cambiado el modelo de negocio, gracias al desarrollo de aplicaciones web, si no que encontramos infinidad de sectores donde ha influido. Es por esto, que encontramos grandes operaciones de marketing realizadas mediante este tipo de métodos. Además la publicidad y el posicionamiento en los motores de búsqueda es un aspecto muy importante en este contexto. Gracias al uso extendido de las tecnologías de la información las empresas consiguen darse a conocer no solo de manera local, si no también global. Nuevos modelos de negocio han surgido debido a este cambio y por supuesto muchos otros han conseguido, mediante el servicio web, ampliar el número de servicios a ofertar o la apertura hacia nuevos mercados que antes no eran posibles

La oferta de servicios mediante el uso de internet nos abre un mundo hacia nuevos modelos de mercado. Mediante el uso de las mismas se consiguen mejores resultados de adaptación tanto como respuesta a la demanda proveniente de los clientes como a los cambios que puedan surgir en el mercado, para los cuales debemos estar preparados y obrar en consecuencia.



Figura 1. Interconexión mundial. (extraído de www.shutterstock.com)

En el caso que nos ocupa no solo estamos buscando ofrecer un servicio más cercano al cliente, quién además puede gestionar, revisar y consultar los movimientos de sus archivos sino que además permite a la propia empresa la gestión misma de archivos, almacenes, tracking y órdenes de trabajo gracias a la realización de una aplicación web hecha a medida

Dentro de la multitud de opciones que nos encontramos a la hora de generar una página web, se decidió generar todo el contenido desde cero. De este modo resultaba una experiencia más positiva en cuanto a la adquisición de conocimientos para el alumno. Diversas herramientas fueron utilizadas entre las que contamos con: HTML5, CSS, JavaScript, PHP y sql.

1.2 Escenario.

Antes de definir posteriores decisiones y ver como es la solución que nos ocupa, vamos a revisar a que se dedica la empresa que demanda el diseño de la aplicación.

Este proyecto surge ante la necesidad de la empresa Transofi S.L por informatizar sus actividades, de modo que pudieran llevar un control total sobre su negocio. Transofi S.L es una empresa multiservicio dedicada a la construcción, mantenimiento y logística para negocios. Se encuentra localizada en Valencia y pese a estar localizada en esa ciudad sus clientes no solo se limitan a la zona de Valencia sino que encontramos clientes que solicitan el servicio desde distintos puntos de España. La actividad principal de Transofi S.L se centra en la recogida y almacenamiento de archivos de otras empresas. Este es el punto que nos interesa y en el que se centrará nuestro desarrollo, en generar una base de datos que sea capaz de ser lo suficientemente flexible para poder guardar un registro de los archivos almacenados y ordenado por: empresas,

departamentos, fecha, etc junto con el diseño de una aplicación web que permita la gestión de dicha base de datos además de otro tipo de operaciones que se definirán a posteriori.

Es vital entender cómo desarrolla TRANSOFI S.L sus funciones para poder realizar una aplicación que sea acorde a las necesidades de nuestra empresa.

TRANSOFI S.L se encarga principalmente al almacenaje de los archivos que generan las distintas empresas clientes. Estos archivos provienen de todos los departamentos como por ejemplo: logística, jurídico, ventas, gestión, producción...

Ante tal volumen de documentos generados, sobre todo si la empresa cliente es grande, disponer de un espacio de almacenamiento y un sistema de etiquetados o manera de organizar los contenidos se hace más que necesario. Muchas empresas prefieren delegar este tipo de tarea a otras empresas, como por ejemplo la empresa que nos ocupa en este caso.

Una vez definida la actividad central de TRANSOFI S.L y el porqué de la misma veamos como realiza este desempeño. Se definirán tanto el modo de clasificar los archivos, como el modo de almacenar de los archivos y el movimiento que realizan los archivos a través de los almacenes.

La clasificación de los archivos se hace de un modo muy simple. Todos los archivos que pertenecen al departamento jurídico se almacenan en un archivador llamado: archivo Z. El resto de departamentos se almacenan en un tipo de archivador llamado archivo definitivo. Además como condición no se pueden mezclar archivos de departamentos distintos en un mismo archivador ni tampoco se pueden mezclar archivos de diferentes años en un mismo archivador.

Una vez definido el tipo de archivador que usaremos para cada departamento, proseguimos a como se van moviendo los archivos una vez han sido recogidos de la empresa cliente. Cuando se hace la orden de recogida por parte de los clientes, se procede a la recogida de los archivos. Una vez se llega a la empresa cliente se meten los archivos dentro de cajas donde irán almacenados distintos archivadores sin importar de que departamento sean exceptuando al jurídico. Una vez introducidos en la caja, dicha caja se sella delante de los empleados de la empresa cliente y se procede a la retirada. Tras todo esto, la caja se lleva a un almacén situado en la ciudad de Valencia, puesto que debido a que son archivos recientes se considera que pueden ser requeridos por parte de los clientes en algún momento. De este modo los archivos permanecen relativamente cerca debido a la posible necesidad de los mismos. Tras un período de tres años, las cajas contenedoras de archivos se mueven a un segundo almacén. Este está situado a unos 150 km de Valencia, y aunque también pueden ser reclamados. El coste por la solicitud de archivos situados en este segundo almacén se incrementa debido a la distancia. Cabe destacar que en este segundo almacén los archivos serán guardados durante un periodo no menor a 2 años. Normalmente este período se extiende acumulando un total de 7 años en los que la empresa almacena los archivos como sumatorio del tiempo transcurrido en el almacén 1 y en el almacén 2.

Una vez ha transcurrido este tiempo, la empresa ofertante del servicio se pone en contacto con el cliente para la destrucción de los archivos. Sin una autorización explícita por parte de los clientes no se puede proceder a la eliminación de los archivos. Usualmente la mayoría de los archivos son eliminados tras el transcurso de ese período de tiempo, aunque hay excepciones como el caso del departamento jurídico. En el caso del departamento jurídico los tiempos de almacenaje pueden ser ampliamente extensibles puesto que el sistema judicial requiere, a veces, de archivos antiguos y no conviene descartarlos a la ligera.

Capítulo 2. Objetivos.

La finalidad de este proyecto es el diseño de una aplicación web para la gestión de una base de datos que contendrá información de los archivos almacenados. Además se debe proveer de un sistema de código de barras para la correcta identificación de los archivos. Dicho código de barras debe ser intuitivo para el ser humano, puesto que en caso de una caída del sistema la empresa debe ser capaz de proveer el servicio.

Para ello debemos definir una serie de objetivos a cumplir:

- **Diseñar una base datos.** La base de datos debe ser lo suficientemente genérica para que, independientemente de la organización departamental de la empresa, se pueda gestionar el movimiento de archivos a través de la aplicación web.
- **Crear una aplicación web** que cumpla con todas las premisas de diseño y se adapte también a las características de la empresa. Por lo que debe ser configurable ya sea desde un administrador perteneciente a la empresa cliente o desde la propia empresa ofertante de servicios.
- **Dotar de un sistema de permisos a la aplicación web** para que de este modo haya dos administradores. Por un lado, el de la propia empresa cliente que dará los primeros pasos en cuanto a ingreso en la página web, organización departamental, estilos, etiquetas, información contenida en el código de barras, etc. Y, por el otro, el administrador en TRANSOFI S.L por si surgiera alguna incidencia o fallo se pudiera subsanar.
- **Establecer de un sistema de código de barras** para la correcta identificación de los archivos. Para hacer posible este sistema de código de barras será necesario elegir lectores de códigos de barras e impresoras que puedan realizar esa tarea.

Una vez claros todos los objetivos y el modo de funcionamiento de la empresa se puede proceder al planteamiento de posibles soluciones, donde se consigan llevar a cabo los objetivos mencionados expresados en forma de tareas.

Capítulo 3. Metodología de trabajo.

En este capítulo se detalla cómo ha sido llevada la metodología de trabajo, haciendo una explicación de cómo se ha gestionado el proyecto, como se ha dividido en tareas más pequeñas así como un diagrama temporal para observar la evolución del proyecto durante el transcurso del tiempo.

3.1 Gestión del proyecto.

Para la exitosa implementación y cumplimiento de los objetivos se realizó una primera toma de contacto para entender cómo funcionaba la empresa TRANSOFI S.L. De este modo se pudo entender que es lo que realmente esperaban de la aplicación.

Una vez de acuerdo las dos partes, desarrollador y gerente, se procedió a hacer un estudio de la viabilidad del proyecto no en tanto a condiciones económicas sino en plazos de entrega. Se estimó unos 6 meses para el desarrollo completo de la aplicación y base de datos, puesto que el desarrollo del proyecto fue enteramente realizado por una persona.

El marco de trabajo que se decidió para la realización del proyecto fue tan solo de lenguajes de programación. Concretamente PHP, JavaScript, HTML5 y CSS. Aunque si bien es cierto que el uso de plataformas como por ejemplo WordPress hubiera agilizado el proceso se decidió no hacer uso de ellas pues se requería un control total de la aplicación. De este modo sin dejar en manos de terceros, plugins o frameworks de la plataforma, de los cuales desconocemos el contenido real de sus módulos, tenemos no solo un control total sobre nuestra aplicación en cuanto a ejecución sino que también tenemos un control total sobre el diseño y tecnologías adicionales que necesitemos.

Durante las fases de desarrollo se decidió trabajar mediante un servidor Debian desde un entorno local. De este modo una vez se hubiera testado el correcto funcionamiento de la aplicación, se realizarían pruebas de estrés sobre el mismo para comprobar el comportamiento ante un gran número de peticiones de posibles usuarios.

Tras haber testado el diseño, corregido algunos errores y comprobado el correcto funcionamiento durante las pruebas de estrés se procedió a migrar la aplicación a un dominio público.

El seguimiento de la gestión del proyecto por parte de TRANSOFI S.L fue el siguiente:

- Reunión inicial donde se hizo una extensa explicación del funcionamiento de la empresa así como la definición de los objetivos (captura de requisitos)
- Presentación de diferentes plataformas para llevar a cabo la solución y aprobación de una de las mismas.
- Reuniones y presentación de la aplicación con diferentes empresas clientes para poder amoldar a nuestro cliente el diseño.
- Presentación final de la aplicación junto con jornadas de instrucción en el uso de la misma

Debido a que el desarrollo de la aplicación fue en la propia Transofi S.L durante la experiencia en prácticas en empresa las revisiones o verificaciones durante el transcurso del proyecto eran diarias. Cuando se requería de información adicional relativa al funcionamiento de la empresa se realizaban pequeñas reuniones con el equipo de gerencia. Si por otro lado se requería información relativa a las empresas clientes, el departamento de gerencia se ponía en contacto con los clientes para concretar los detalles de la duda. Se tuvo contacto con los clientes por parte del desarrollador en las fases intermedias y finales del desarrollo puesto que gerencia tenía cubierta las expectativas iniciales de los clientes.

3.2 Distribución de tareas.

La distribución de tareas se realizó en 4 fases, en una primera fase donde se realizaba la captura de requisitos y necesidades, una segunda fase donde se decidieron las tecnologías de desarrollo junto con el entorno de trabajo, una tercera fase donde se realizó el diseño, implementación y testeo de la aplicación. Y finalmente, una cuarta fase donde se hizo una elección de la plataforma donde comprar el hosting se migró la aplicación a dicha plataforma y se testeó para acabar de cerciorarnos del correcto funcionamiento de la aplicación.

- **FASE I:** Captura de requisitos tanto por parte de los clientes como de TRANSOFI S.L
- **FASE II:** Elección de las herramientas de desarrollo e instalación de las mismas, elección de dispositivo local para transformarlo en un servidor.
- **FASE III:** Diseño, implementación y testeo de la aplicación web y la base de datos. Podemos dividir el proyecto en las siguientes funcionalidades:
 - 1) Instalación de todos los componentes necesarios: Servidor e infraestructura para el desarrollo web así como las herramientas de desarrollo web y bases de datos.
 - 2) Diseño de una jerarquía de permisos y políticas de registro por parte de las empresas clientes.
 - 3) Diseño e implementación de la base de datos.
 - 4) Diseño e implementación de los módulos que compondrán la aplicación web para la parte de cliente.
 - 5) Diseño específico de los módulos relacionados con la administración de la aplicación web.
 - 6) Elección y compra de las máquinas lectoras de códigos de barras e impresoras de etiquetas.
 - 7) Implantación de medidas de seguridad.
 - 8) Testeo de la aplicación.
- **FASE IV:** Migración del contenido generado a un espacio de hosting y testeo del correcto funcionamiento tras el movimiento.

3.3 Diagrama temporal.

La evolución temporal es un aspecto importante de la planificación a la hora de realizar un proyecto, por lo que en la siguiente figura veremos cómo transcurrió todo el proceso a lo largo del tiempo. Este proyecto se realizó durante unas prácticas en empresa con una duración de 5 meses y un horario laboral de media jornada para que el alumno pudiera compatibilizar sus estudios con el trabajo. En total fueron 480 horas, aunque por supuesto se tuvo que invertir más tiempo debido a aspectos como el aprendizaje personal de los nuevos lenguajes de programación y entornos con los que lidiar.

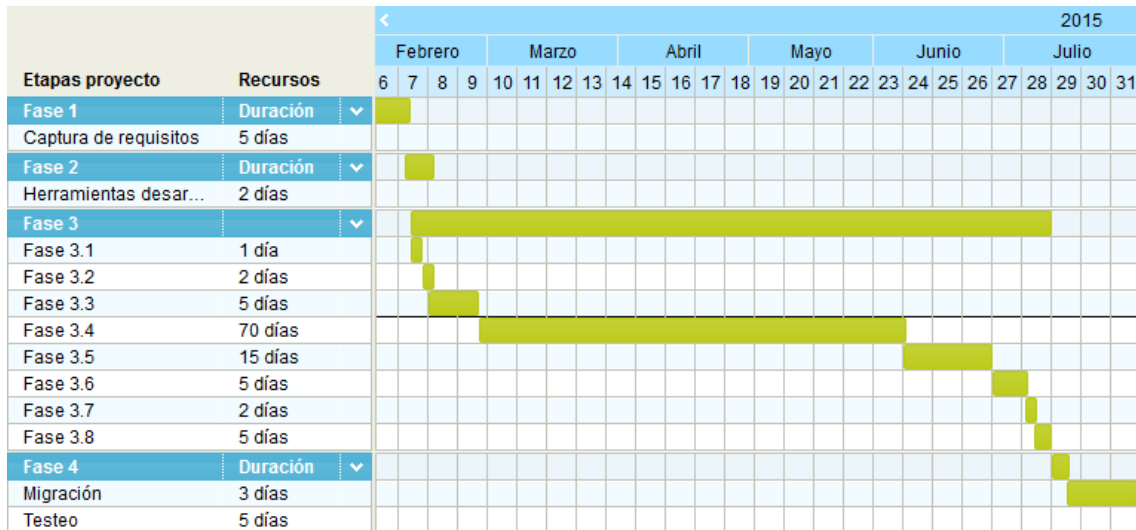


Figura 2. Planificación Temporal.

Capítulo 4. Instalación del servidor e infraestructura de trabajo.

En este capítulo se detallarán el desarrollo y resultados obtenidos así como la exposición de las herramientas utilizadas a lo largo del proceso.

4.1 Contexto tecnológico

El primer paso para la realización de la tarea era disponer de un alojamiento local para el desarrollo de la aplicación. En nuestro caso, se decidió el uso de un ordenador de la compañía que no estaba siendo utilizado. Puesto que se trataba de un entorno transitorio donde desarrollar el trabajo, las prestaciones del mismo no se tuvieron en cuenta, es decir, se trataba a dicha computadora como banco de pruebas donde desarrollar y testear nuestro diseño.

Como decisión inicial se decidió formatear e instalar una distribución de Linux para que hiciera de servidor. La distribución que se decidió instalar fue Debian por diversos motivos que ahora se expondrán. En primer lugar Debian sigue siendo una de las distribuciones más usadas en entornos de servidores según las estadísticas publicadas por la W3techs. Además su cuota de mercado sigue en aumento, cosa que nos indica que los desarrolladores confían cada vez más en este sistema operativo.

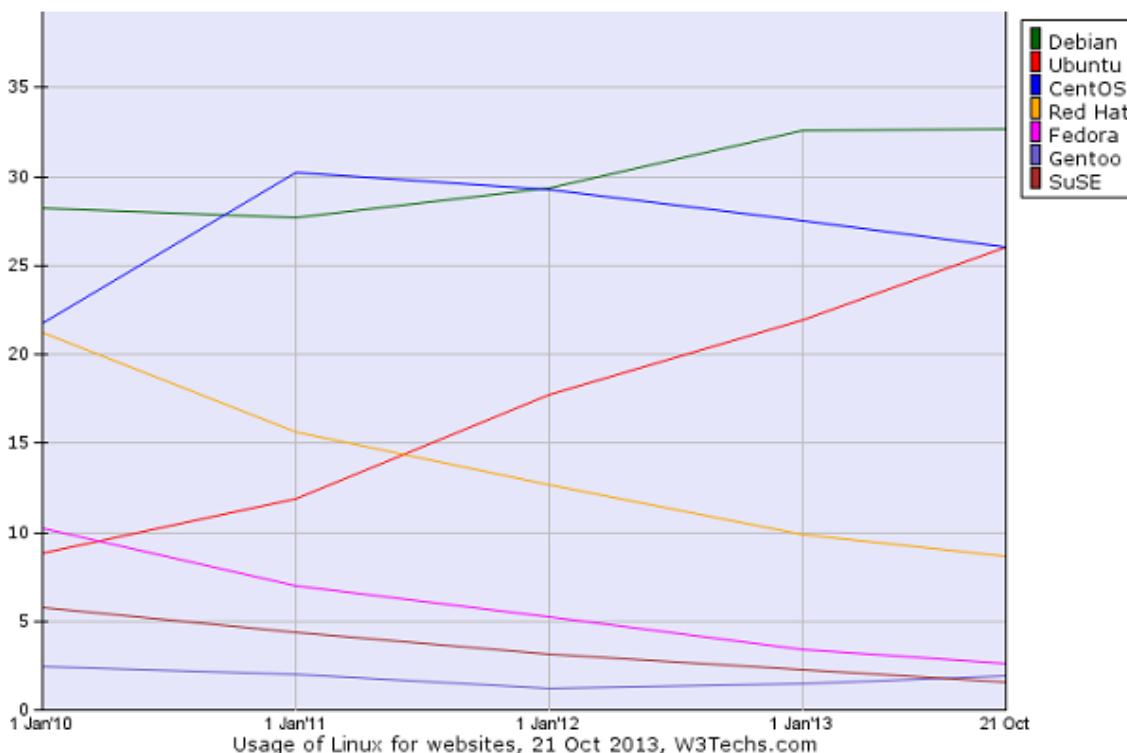


Figura 3. Gráfica uso de distribuciones en servidores Linux. (Extraído de www.w3techs.com)

Además de la gran cantidad de usuarios que descargan esta distribución existen otros puntos positivos que hacen que Debian resulte una distribución adecuada para el desarrollo. A continuación se expondrán una serie de razones por las cuales elegir esta distribución: [1]

- **Está mantenida por sus usuarios.** Cuando surgen problemas o necesidades de mejora la comunidad de usuarios responde de una forma muy prolífera.
- **Soporte ante incidencias.** Ante la posibilidad de aparecer problemas durante el transcurso de uso existe la posibilidad de enviar un correo informando de la problemática. Estos correos son frecuentemente respondidos por los desarrolladores.
- **Un sistema de empaquetamiento de software robusto.** El sistema de empaquetamiento de software de Debian, dpkg es muy sólido y confiable pudiendo confiar las actualizaciones de paquetes o del propio software en él.
- **Fácil instalación.** La instalación es muy sencilla como posteriormente se detallará. Además se puede hacer desde un CD,DVD, Blu-ray, una memoria USB o incluso a través de la red.
- **Extensa cantidad de software.** Debian viene con más de 43000 elementos de software diferentes. Cada bit de éstos es libre. Si se necesita software propietario que corre bajo GNU/Linux o GNU/kFreeBSD, puede usarse (de hecho, puede que incluso exista un instalador en Debian que automáticamente instale y configure todo por nosotros).
- **Integración de paquetes.** Debian sobrepasa a todas las otras distribuciones en lo bien integrados que están sus paquetes. Como todo el software lo empaqueta un grupo coherente, no sólo se pueden encontrar todos los paquetes en un mismo sitio sino que se puede estar seguro de que hemos eliminado todos los problemas al respecto de complejas dependencias. Aunque creemos que el formato .deb tiene algunas ventajas sobre el rpm, es la integración entre paquetes lo que hace a un sistema Debian más robusto.
- **Actualizaciones fáciles.** Actualizarse a una nueva versión de Debian es muy fácil gracias al sistema de empaquetamiento. Sólo hay que ejecutar **apt-get update ; apt-get dist-upgrade (o aptitude update ; aptitude dist-upgrade..**
- **Soporta múltiples arquitecturas y kernels.** Actualmente Debian soporta un impresionante número de arquitecturas CPU: alpha, amd64, armel, hppa, i386, ia64, mips, mipsel, powerpc, s390, y sparc.
- **Estabilidad.** Existen muchos casos de máquinas que trabajan durante más de un año seguido sin reiniciarse. De la misma forma, hay equipos que tan sólo son reiniciados debido a un fallo en el suministro de corriente o a una actualización del hardware.
- **Rápido y ligero en memoria.** Otros sistemas operativos pueden ser rápidos en una o dos áreas, pero, estando basado en GNU/Linux o GNU/kFreeBSD, Debian es ligero y humilde. El software para Windows se ejecuta bajo GNU/Linux usando un emulador a veces más rápido que en su ambiente original.
- **Buena seguridad del sistema.** Debian y la comunidad del software libre son muy sensibles en asegurarse de que los arreglos de problemas de seguridad entren en la distribución rápidamente. Normalmente, los paquetes arreglados se hacen disponibles a los pocos días. La disponibilidad del código fuente permite que la seguridad en Debian se evalúe de forma abierta, lo que evita que se implementen modelos de seguridad pobres. Además, la mayoría de los proyectos de software libre tienen sistemas de revisión por terceras partes, que, como primera medida, evitan que se introduzcan en el sistema problemas de seguridad potenciales.
- **Software de seguridad.** Cualquier cosa enviada por la red puede ser leída por cualquier máquina entre emisor y el receptor. Debian tiene paquetes del famoso software GPG (y PGP) que permite enviar correo entre usuarios preservando su privacidad. Además, ssh permite crear conexiones seguras a otras máquinas que tengan ssh instalado.
- **La adquisición de esta distribución de forma legal es gratuita.**

Por supuesto Debian no es perfecto. Existen áreas en las cuales suelen ser queja común de los usuarios. Estamos hablando de casos como la falta de software comercial popular, no todo el hardware está soportado y que no es trivial la configuración del sistema operativo. Aún con todo, en contrapartida tenemos muchas versiones de programas populares que tienen su réplica en formato de software libre. Por otro lado la configuración del servidor, no confundamos con la instalación del mismo, aunque no sea muy apta para usuarios noveles tiene solución. Como ya se ha comentado la extensa comunidad de Debian es de gran ayuda por lo que si tenemos problemas o dudas podemos ir en busca de ayuda fácilmente. De hecho en muchos casos incluso podemos encontrar scripts realizados por otros usuarios que hacen justamente lo que nosotros necesitamos. Finalmente en cuanto al hardware, si bien es cierto que no todo está soportado estamos hablando de casos puntuales, como por ejemplo hardware muy nuevo o muy antiguo.

Por todo lo acabado de exponer se decidió no solo usar Debian para nuestro entorno de trabajo en local, sino que también se decidió que cuando se migrara la aplicación a un hosting de dominio público se hiciera a un servidor Debian. De esta manera se pueden aprovechar todas las ventajas que este sistema operativo ofrece así como las configuraciones ya realizadas en nuestro entorno de trabajo ahorrando mucho tiempo en cuestiones tales como aprender a configurar otro sistema operativo.



Figura 4. Debian. (Extraído de mysecureshell.com)

4.2 Instalación de Debian.

Como ya se ha comentado anteriormente la instalación de Debian es muy sencilla. De hecho encontramos múltiples páginas web donde se detalla como instalar esta distribución en nuestra máquina. Seguidamente se determinaran los pasos a seguir para poder instalar dicha distribución.

[2] En primer lugar se debe conseguir una copia de Debian. Existen múltiples métodos para la obtención de la misma desde la compra de discos de instalación a realizar la descarga desde la página web de Debian de forma gratuita, legal y libre de malware. La descarga se puede realizar mediante clientes de torrent o directamente mediante nuestro navegador web a través de la descarga directa del contenido.

The screenshot shows the 'Debian en CDs' page. At the top, there is a navigation bar with links: 'Acerca de Debian', 'Cómo obtener Debian', 'Soporte', and 'Rincón de los desarrolladores'. Below the navigation bar, the page title is 'Debian en CDs'. The main content area starts with a paragraph: 'Si quiere obtener Debian en CD o memoria USB, vea abajo las opciones disponibles. En caso de problemas puede ver las [PUF \(FAQs\) sobre los CDs de Debian](#).' Below this, there are two yellow boxes with lightbulb icons. The first box says: 'Si simplemente quiere instalar Debian y tiene una conexión de Internet en el sistema donde va a instalarlo, considere el uso de los medios para la [instalación en red](#) que son descargas más pequeñas.' The second box says: 'En las arquitecturas i386 y amd64, todas las imágenes de CD/DVD también pueden [usarse en memorias USB](#).' Below these boxes, there is a list of download options:

- [Compre CD-ROMs de Debian terminados](#). Son baratos: no perseguimos obtener beneficios con ellos. Si su conexión a Internet se tarifica por minutos, ésta es su única opción. También puede considerar el comprar los CDs si sólo tiene un módem, ya que bajar las imágenes con un módem puede tardar días.
- [Descarga de imágenes de CD/DVD con jigdo](#). El diseño de «jigdo» le permite seleccionar la más rápida de las 300 réplicas de Debian de todo el mundo para su descarga. Proporciona una fácil selección de la réplica y «actualización» de imágenes viejas a la última versión. También es la única manera de descargar imágenes de DVD de Debian para *todas* las arquitecturas.
- [Descarga de imágenes de CD/DVD con BitTorrent](#). El sistema «peer to peer» Bittorrent permite descargar imágenes de forma cooperativa a varios usuarios al mismo tiempo, con una carga mínima para nuestros servidores. Las imágenes de DVD sólo están disponibles para algunas arquitecturas.
- [Descarga de imágenes de CD/DVD usando HTTP o FTP](#). Debido a restricciones de espacio o de ancho de banda, sólo unas pocas réplicas proporcionan directamente enlaces para descarga directa mediante HTTP/FTP. *Estos sitios pueden experimentar escasez de ancho de banda*, y las descargas pueden ser bastante lentas. Por favor, use [jigdo](#) o [bittorrent](#) en su lugar. Las imágenes de DVD sólo están disponibles para algunas arquitecturas y no en todas las réplicas.

 On the right side of the page, there is a sidebar with a list of links: 'Debian en CD', 'PUF', 'Instalación via red', 'Comprar CDs o DVDs', 'Descargar', 'Descargar con jigdo', 'Descargar mediante HTTP/FTP', 'Descargar con Torrent', 'Réplicas rsync', 'Verificar', 'miscelánea', 'Arte', 'Réplicas', and 'Información de imágenes de distribución'.

Figura 5. Descargar Debian.

En nuestro caso se eligió la descarga de CD/DVD usando HTTP o FTP. Una vez en hemos llegado hasta aquí hemos de elegir si elegir una versión estable o una de testing. Las versiones de testing se generan semanalmente por lo que se actualizan frecuentemente. Aun así, se decidió descargar la versión estable para tener un sistema lo más robusto posible. Llegados a este punto queda elegir la arquitectura que deseamos del sistema operativo. En nuestro caso la arquitectura de nuestra computadora era de 64 bits por lo que se decidió elegir la versión amd64 para sacar el máximo partido posible de nuestra máquina.















<u>Name</u>	<u>Last modified</u>	<u>Size</u>
 Parent Directory		-
 MD5SUMS	2016-04-03 12:24	963
 MD5SUMS.sign	2016-04-03 12:27	819
 SHA1SUMS	2016-04-03 12:24	1.1K
 SHA1SUMS.sign	2016-04-03 12:27	819
 SHA256SUMS	2016-04-03 12:24	1.4K
 SHA256SUMS.sign	2016-04-03 12:27	819
 SHA512SUMS	2016-04-03 12:24	2.3K
 SHA512SUMS.sign	2016-04-03 12:27	819
 debian-8.4.0-amd64-DVD-1.iso	2016-04-02 17:59	3.7G
 debian-8.4.0-amd64-DVD-2.iso	2016-04-02 17:59	4.4G
 debian-8.4.0-amd64-DVD-3.iso	2016-04-02 17:59	4.3G
 debian-update-8.4.0-amd64-DVD-1.iso	2016-04-03 11:35	4.2G
 debian-update-8.4.0-amd64-DVD-2.iso	2016-04-03 11:37	3.2G

Figura 6. Descarga del sistema operativo Debian

Nos encontramos en la recta final de la obtención del sistema operativo. Por facilidad y comodidad se decidió descargar la versión .iso del DVD 1. Con solo la parte inicial es suficiente puesto que una vez iniciada la instalación los componentes adicionales que se fueran a necesitar se instalarían automáticamente puesto que la máquina en la que se iba a realizar la instalación estaba conectada a internet. Cabe destacar que en estos directorios no solo encontramos versiones del sistema operativo sino que además vienen firmados. Encontramos estos archivos junto con los .iso a descargar confirmando así la autenticidad del software.

Una vez realizada la descarga se puede proceder a la instalación del sistema operativo en cuestión. Su instalación es sencilla y rápida por lo que, como ahora veremos, en unos sencillos pasos tendremos el sistema operativo instalado y funcionando.

Para poder instalar el sistema operativo necesitamos de un soporte en mediante el que instalaremos la distribución. De un modo muy sencillo podemos preparar una unidad Live USB con cualquier distribución de Linux preparada para la instalación mediante el programa Yumi Multiboot USB Creator. Gracias a este pequeño programa, el cual no requiere de instalación, podemos conseguir un USB para instalar cualquier distribución de Linux. Además Yumi almacenar diversas distribuciones de instaladores de sistemas operativos Linux. De este modo en una única memoria USB podemos tener la posibilidad de instalar o probar varias distribuciones Linux mediante un solo soporte. Otro aspecto importante es que también podemos desinstalar con usando el mismo dispositivo generado con Yumi nuestra distribución. De este modo obtenemos una herramienta con la que de un solo golpe podemos preparar un Live USB que contenga una o varias distribuciones y nos sirva para eliminar aquellas que no necesitamos. Por último decir que este programa se encuentra disponible tanto para las plataformas de Windows como de Linux y es totalmente gratuito.

Una vez tenemos nuestra unidad USB preparada con el sistema operativo a instalar, debemos en primer lugar apagar el ordenador, insertar el USB en alguno de los puertos USB e iniciar. Si todo ha ido correctamente, tras enchufar el ordenador deberíamos encontrarnos con una imagen como esta.

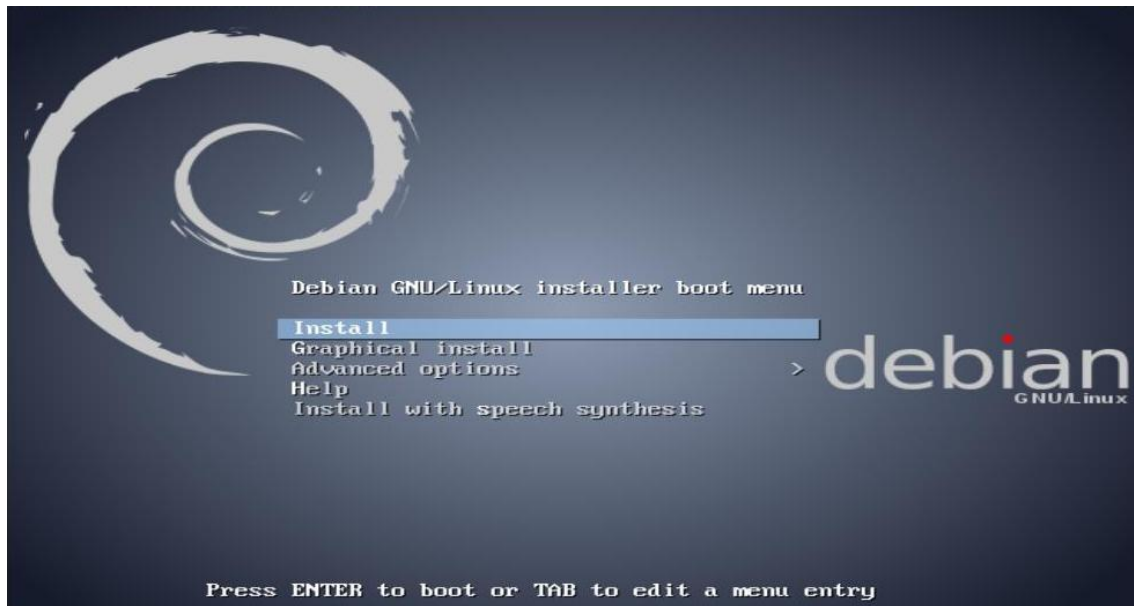


Figura 7. Menú de instalación de Debian.

Como no queremos varios sistemas operativos en el mismo ordenador ni queremos conservar nada la instalación será muy sencilla. Tras aparecer esta imagen seleccionamos la opción de instalar. Tras haber elegido esta opción lo siguiente antes de empezar la instalación será configurar minimamente el sistema.

En primer lugar deberemos seleccionar el idioma, que en nuestro caso será Spanish. Seguidamente nos aparecerá una pantalla donde deberemos introducir nuestro país para luego preguntarnos nuestro tipo de teclado. Tras ello, deberemos seleccionar un nombre para nuestra máquina que en nuestro caso elegimos como nombre Debian.

El siguiente paso en la instalación está relacionado con el usuario administrador, también conocido como root. Linux tiene una jerarquía de usuarios muy interesante, en la que todo está gestionado por el usuario root. Este gestiona los permisos de otros usuarios en cuanto a archivos. En algunos casos quizás sea interesante que ciertos usuarios puedan leer según que archivos pero no modificarlos así como otros modificarlos y leerlos. Para el usuario root todo es posible y además es el que se encarga de permitir la instalación del software tanto del descargado insitu por el propio usuario como del instalado por las actualizaciones del sistema. Por ello en este paso de la instalación piden la asignación de una contraseña para este usuario. De este modo todo usuario que conozca la contraseña será elevado en permisos a usuario root en ese instante y podrá instalar tranquilamente. Una vez introducida la contraseña se vuelve a pedir otra vez la contraseña por si hubiera habido algún tipo de error al teclear. Tras ello debemos darle un nombre al usuario administrador con el cual la máquina se dirigirá al susodicho usuario.

Una vez gestionada la creación del usuario administrador, es el momento de crear nuestro usuario. Sencillamente hay que asignar el nombre que se desee y continuar. Por nuestra seguridad el programa de instalación recomienda que elijamos una contraseña de acceso a la máquina por lo que una vez introducida podemos ir al siguiente paso.

Para acabar de cerciorarse el programa de instalación ahora nos preguntará en que zona vivimos la península, las islas canarias o en Ceuta y Melilla. Este tipo de cuestiones se hacen por el reloj de la máquina así que lo introduciremos correctamente e iremos al siguiente paso.

Una vez recolectados todos estos datos se procede a la instalación del sistema. Pero antes de empezar con ello debemos elegir algunos aspectos. La primera cuestión a resolver es si deseamos realizar una instalación manual o automática. En nuestro caso por simplicidad

elegimos la instalación automática. De este modo el instalador se encargara de organizar el particionado del disco sin ocasionar ninguna molestia en nosotros. Una vez elegidas estas opciones se nos proponen diversas formas de particionar el disco. Para no complicarnos elegimos la opción que viene marcada como ideal para usuarios novatos.

Finalmente como veremos en esta imagen, el programa de instalación nos dice como quedará el disco.

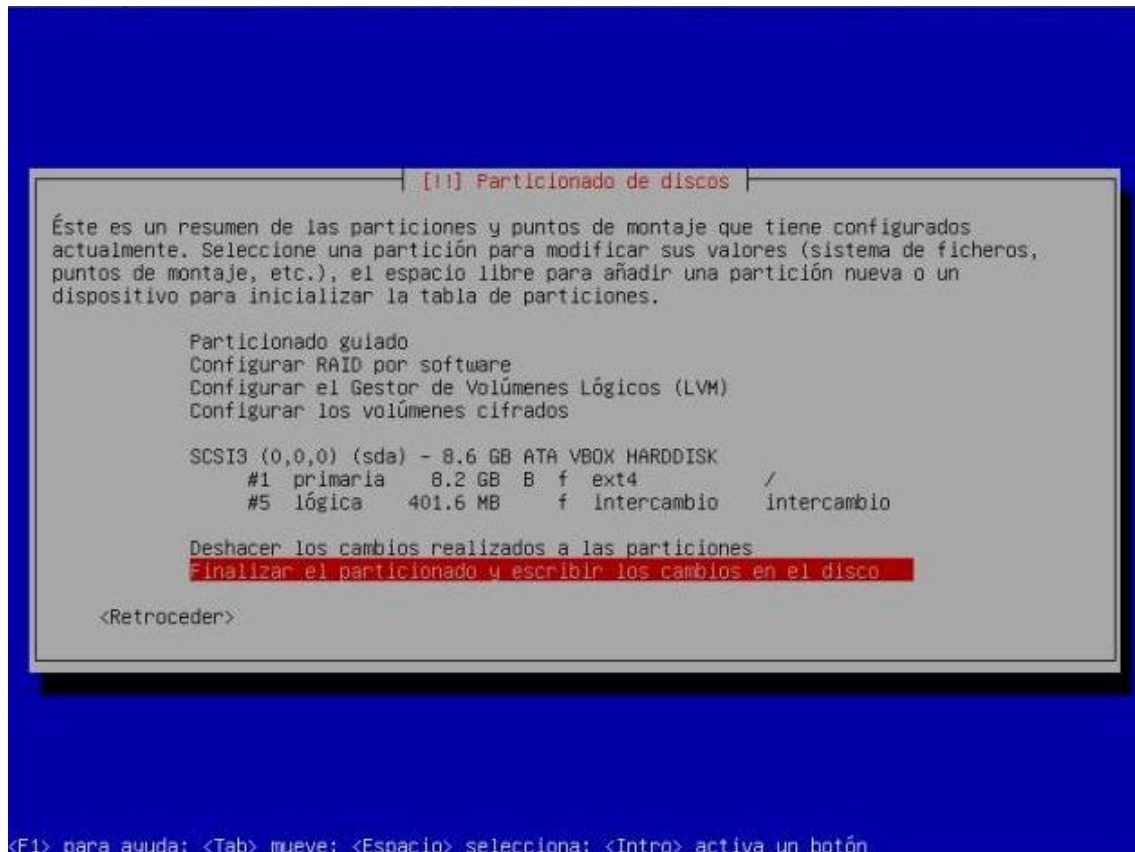


Figura 8. Particionado de discos.

Antes de proceder a la instalación el instalador del sistema operativo volverá a pedir nuestra confirmación para la instalación la cual por supuesto vamos a aceptar. Tras esto la instalación va a empezar y como ya se mencionó anteriormente al solo haber descargado uno de los DVD vamos a necesitar descargar el resto del contenido desde internet.

Para terminar la instalación desde la red y agilizar el trámite el propio instalador nos preguntará desde que país queremos realizar la descarga. Se aconseja descargar desde el país de donde se esté realizando la instalación puesto que así cada usuario descarga desde su zona y no se satura un servidor en concreto. Aunque la realidad es que hay tanto servidores disponibles para la descarga que no importa desde donde realicemos la descarga.

En el siguiente paso nos preguntarán que estamos instalando, un sobremesa (desktop), un servidor, etc. En nuestro caso elegiremos un servidor ya que de este modo posibles componentes que necesitemos o configuraciones previas que necesiten hacerse estarán ya hechas.

Tras todo esto nuestro proceso para realizar la instalación ha terminado y solo queda esperar a que se terminen de descargar los paquetes restantes de internet y a la instalación de los mismos.

4.3 Instalación de SSH.

Una vez instalado nuestro sistema operativo nos encontramos en disposición de instalar el resto de componentes necesarios para el desarrollo. Para la instalación de los distintos componentes de software se hubiera podido dotar de un monitor, teclado y ratón al servidor y realizar esas gestiones en frente de la máquina. Pero como es usual el servidor se encontraba apartado de la zona de oficinas por lo que se decidió instalar SSH para establecer una conexión remota con la máquina y realizar estas tareas sin desplazarse.

4.3.1 SSH.

Los pasos que seguiremos para la instalación ssh están basados en la página web servidor-debian.org [3]

Para empezar con la instalación del servicio SSH necesitamos ejecutar esta sentencia en nuestro terminal con permisos de usuario administrador.

```
root@server:~# aptitude install openssh-server openssh-client
```

Una vez ejecutada esta sentencia nuestro servidor ssh solo necesita de la configuración apropiada para la conexión desde una máquina remota.

4.3.2 Configuración de SSH

Todas las configuraciones del servidor ssh se encuentran en el archivo `/etc/ssh/sshd_config`. Para configurar el servidor se debe indicar las direcciones donde el servicio debe responder. En este caso serán conexiones ligadas a la dirección `192.168.0.30`, a través del puerto 22 y utilizando la versión 2 del protocolo `ssh`. En nuestro archivo las líneas modificadas son las siguientes:

```
#What ports, IPs and protocols we listen for
Port 22
# Use these options to restrict which interfaces/protocols sshd will
bind to
ListenAddress 192.168.1.100
Protocol 2
```

Por seguridad, se debe desactivar el `login` como `root`. Para adquirir los privilegios del `root`, se debe hacer un `login` usuario normal y, después, adquirir los privilegios de `root`. De este modo, prevenimos que el `password` del `root` sea objeto de un ataque. Luego en nuestro archivo de configuración debemos establecer la opción `PermitRootLogin` denegada.

```
PermitRootLogin no
```

También se debe verificar que no sean permitidos los `logins` con contraseñas vacías:

```
# To enable empty passwords, change to yes (NOT RECOMMENDED)
PermitEmptyPasswords no
```

Para finalizar el proceso de instalación solo debemos reiniciar el servicio y nuestro servidor estará listo para poder acceder a él a través de ssh.

Por último queda la verificación desde nuestras máquinas. Según lo configurado la conexión debe ser posible a través de la dirección `192.168.1.30`. En las siguientes líneas se verá como fue realizada la primera conexión con el servidor mediante otro equipo Linux.

```
yassin@laptop:~$ ssh 192.168.1.30
The authenticity of host '192.168.1.30 (192.168.1.30)' can't be
established.
RSA key fingerprint is
ee:16:b0:c9:1b:ef:b4:64:e1:86:80:f4:36:9f:08:03.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.30' (RSA) to the list of known
hosts.
yassin@192.168.1.30's password:
Linux server 2.6.32-5-amd64 #1 SMP Fri Oct 15 00:56:30 UTC 2010 x86_64

The programs included with the Debian GNU/Linux system are free
software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

fribeiro@server:~$ logout
Connection to 192.168.1.30 closed.
```

Si en lugar de utilizar una máquina con una distribución de Linux como sistema operativo utilizamos una con Windows podemos realizar la conexión con un emulador del terminal que soporte ssh. Para Windows el caso más sencillo que nos cubre este aspecto es el programa Putty.

4.4 Infraestructura LAMP. Linux Apache, MySQL y PHP.

En el caso que nos ocupa disponíamos de la libre elección de como estructurar nuestra aplicación. Puesto que nuestro servidor Debian debía tener instalado un servidor web, un servidor de base de datos, así como la disposición de un lenguaje de programación para construir nuestras aplicaciones se decidió elegir una solución basada en la infraestructura LAMP. Con este tipo de solución obtenemos todo un conjunto de herramientas de una sola operación, gratuita y además con gran cantidad de documentación.

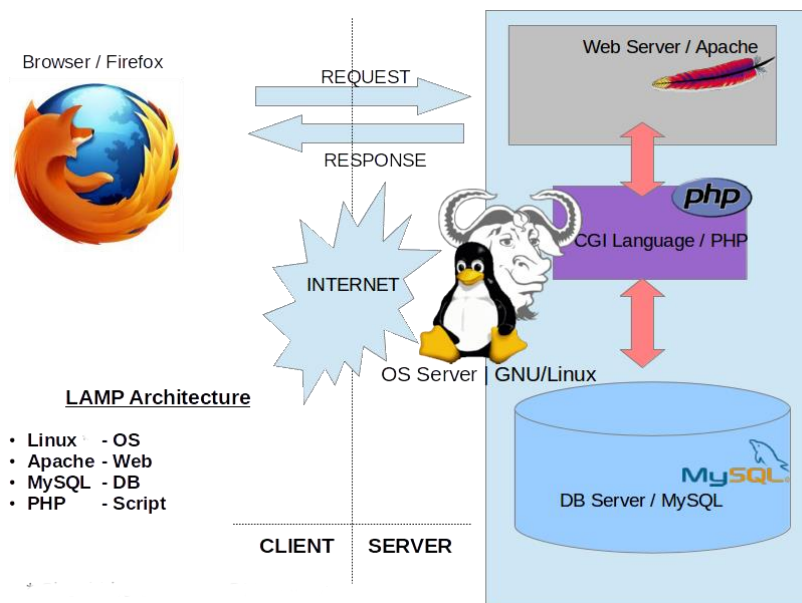


Figura 9. Infraestructura LAMP. (Extraído de wikipedia)

Nuestra infraestructura LAMP se compone de las siguientes herramientas:

- Linux. Sistema operativo donde se alojarán todos los componentes.
- Apache. Servidor web encargado de implementar el protocolo HTTP.
- PHP. Lenguaje de programación en el lado del servidor orientado al desarrollo web.
- MySQL. Servidor de bases de datos junto con el software phpmyadmin para la gestión y administración de la base de datos.

4.4.1 Apache

Apache es un servidor HTTP de código abierto para la creación de páginas y servicios web. Una de las grandes ventajas de Apache es que se trata de un servidor multiplataforma, gratuito y con licencia GPL. Apache sigue siendo desarrollado por la comunidad de usuarios desarrolladores bajo la tutela de la Apache Software Foundation.

Las principales características de Apache son:

- Soporte de seguridad SSL y TLS
- Autenticación de datos mediante una SGBD
- Capacidad para dar soporte a diversos lenguajes como: php, python o perl.

Apache es utilizado para servir páginas web, ya sean estáticas o dinámicas. Para conformar páginas web se suele necesitar de acceso a bases de datos (SQL) así como a distintos lenguajes de programación (php, python, perl).[4]



Figura 10. Apache software. (Extraído de commons.wikimedia.com)

4.4.2 PHP

PHP (acrónimo recursivo de *PHP: Hypertext Preprocessor*) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.[5]

El código de PHP está encerrado entre las etiquetas especiales de comienzo y final `<?php` y `?>` que permiten entrar y salir del "modo PHP".

Lo que distingue a PHP de algo del lado del cliente como Javascript es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá el resultado de ejecutar el script, aunque no se sabrá el código subyacente que era. El servidor web puede ser configurado incluso para que procese todos los ficheros HTML con PHP, por lo que no hay manera de que los usuarios puedan saber qué se tiene debajo.

Lo mejor de utilizar PHP es su extrema simplicidad para el principiante, pero a su vez ofrece muchas características avanzadas para los programadores profesionales.



Figura 11. Php. (Extraído de commons.wikimedia.com)

4.4.3 MySQL y PhpMyAdmin

[6] MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation y está considerada como la base datos open source más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.

Está desarrollado en su mayor parte en ANSI C y C++. Tradicionalmente se considera uno de los cuatro componentes de la pila de desarrollo LAMP y WAMP

Existen varias interfaces de programación de aplicaciones que permiten, a aplicaciones escritas en diversos lenguajes de programación, acceder a las bases de datos MySQL, incluyendo C, C++, C#, Java (con una implementación nativa del driver de Java), Perl, PHP, Python, Ruby, etc... Cada uno de estos utiliza una interfaz de programación de aplicaciones específica. También existe una interfaz ODBC, llamado MyODBC que permite a cualquier lenguaje de programación que soporte ODBC comunicarse con las bases de datos MySQL.

Ciertamente MySQL es un sistema de bases de datos muy utilizado que ofrece una gran cantidad posibilidades. Entre las características más apreciadas entre los desarrolladores nos encontramos con que:

- Permite la ejecución multihilo.
- Gran cantidad de API's en multitud de lenguajes y frameworks de trabajo.
- Su status de open source permite que tanto la utilización gratuita como la modificación del mismo sea posible.
- Se puede descargar su código fuente, favoreciendo de este modo el desarrollo y las continuas actualizaciones.

PhpMyAdmin es una herramienta de software gratuita escrita completamente en el lenguaje de programación php. El propósito de esta herramienta es ofrecer la posibilidad, a través de un servicio web, de gestionar una base de datos mediante un entorno gráfico. PhpMyAdmin permite la total gestión de nuestra base de datos mediante el entorno gráfico pero sin perder la posibilidad de ejecutar sentencias sql contra la base de datos.

Muchas son las posibilidades que ofrece phpMyAdmin y entre ellas se quiere destacar las siguientes:

- Interfaz web intuitiva
- Posibilidad de importar y exportar en diversos formatos: CSV, SQL, XML, PDF...
- Administración de diversos servidores.
- Soporte para la mayoría de las características principales de MySQL:
 - Búsqueda, eliminación, creación, copia y renombramiento de bases de datos, tablas vistas, campos e índices.
 - Manejo de cuentas de usuario y privilegios
 - Manejo de procesos almacenados y de disparadores.



Figura 12. MySQL y PhpMyAdmin. (Extraídos de commons.wikimedia y mysql.com)

4.5 Instalación del servidor LAMP en Debian.

Una vez claros los componentes que conforman la infraestructura LAMP es hora de empezar con la instalación. Para ello el modo de proceder será a través del terminal de Debian.

La información de cómo realizar la instalación de los componentes se ha seguido de la web digitalocean.com [7]. En primer lugar para la instalación de LAMP debemos asegurarnos de tener nuestro sistema actualizado, cosa que podemos realizar con la siguiente instrucción:

```
# aptitude update && aptitude upgrade
```

Una vez el sistema ha comprobado que tiene todos sus componentes actualizados y en caso de no tenerlos haberlos actualizado podemos empezar con la instalación de los distintos componentes de LAMP.

4.5.1 Instalación de MySQL.

Vamos a empezar con la instalación de MySQL con la siguiente instrucción:

```
#aptitude install mysql-server mysql-client
```

Tras la instalación del servidor MySQL, se debe cambiar la contraseña del usuario root. Este paso se produce automáticamente desde la línea de comandos puesto que al finalizar la instalación el propio programa de instalación nos pedirá introducir una contraseña. Veremos en nuestro prompt una línea de este estilo:

```
# /usr/bin/mysqladmin -u root password 'enter-your-good-new-password-here'
```

Una vez modificada la contraseña de usuario supervisor habremos acabado con la instalación de nuestro servidor y cliente MySQL.

4.5.2 Instalación de Apache.

El proceso de instalación del servidor Apache es incluso más sencillo que el de MySQL. Para proceder con la instalación deberemos proceder del siguiente modo:

```
# aptitude install apache2 apache2-doc
```

Tras esto nuestro servidor apache ha sido instalado. Ahora bien, aunque la instalación ha sido sencilla nuestro servidor requiere de una configuración inicial para los directorios de usuario del Apache Web Server que ahora veremos.

En primer lugar debemos habilitar el siguiente módulo:

```
# a2enmod userdir
```

Seguidamente vamos con la configuración del módulo de Apache localizado en el directorio de usuario con la ruta `/etc/apache2/mods-enabled/userdir.conf`. La modificación de este archivo es sencilla, solo deberemos localizar la porción de texto que esta entre las etiquetas `<IfModule>` `</IfModule>` i sustituirlas por:

```
<IfModule mod_userdir.c>
    UserDir public_html
    UserDir disabled root

    <Directory /home/*/public_html>
        AllowOverride All
        Options MultiViews Indexes SymLinksIfOwnerMatch
        <Limit GET POST OPTIONS>
            Require all granted
        </Limit>
```

```
<LimitExcept GET POST OPTIONS>
    Require all denied
</LimitExcept>
</Directory>
</IfModule>
```

Con esta directiva lo que hacemos es establecer una configuración básica para trabajar como un usuario normal y no como supervisor.

Para acabar con nuestra configuración de servidor Apache solo debemos seguir unos pocos pasos más. Nuestro siguiente paso consiste en crear el directorio `public_html` donde residirá nuestra estructura de páginas web. Para ello ejecutaremos la siguiente sentencia, pero no como root sino como usuario:

```
$mkdir /home/$USER/public_html
```

Ahora debemos substituir el grupo de root a usuario y para finalizar resetear nuestro servidor:

```
# chgrp www-data /home/<username>/public_html
# service apache2 restart
```

Hemos finalizado nuestra instalación de apache.

Como comprobación final de que nuestro servidor está funcionando correctamente haremos una pequeña prueba. Dejaremos una pequeña porción de código HTML sencillo en nuestro directorio `~/public_html`. Para ver el resultado de nuestra obra deberemos abrir nuestro navegador web e introducir una url de este estilo `http://localhost/~username`.

Si todo ha ido correctamente veremos nuestra sencilla página web, que en este caso tenía este aspecto:



Figura 13. Pájaro Bici.

4.5.3 Instalación de PHP.

El paso previo a la instalación es comprobar si nuestro Apache es capaz de servir contenido php. Para ello debemos comprobar el archivo `php5.conf` localizado en la ruta `/etc/apache2/mods-available/`. Abrimos el archivo y debemos ver un contenido igual o similar al siguiente:

```
<IfModule mod_php5.c>
  <FilesMatch "\.ph(p3?|tml)$">
    SetHandler application/x-httpd-php
    Require all granted
  </FilesMatch>
  <FilesMatch "\.phps$">
    SetHandler application/x-httpd-php-source
    Require all denied
  </FilesMatch>
  # To re-enable php in user directories comment the following lines
  # (from <IfModule ...> to </IfModule>.) Do NOT set it to On as it
  # prevents .htaccess files from disabling it.
  #<IfModule mod_userdir.c>
  #   <Directory /home/*/public_html>
  #       php_admin_value engine Off
  #   </Directory>
  #</IfModule>
</IfModule>
```

Si el contenido del archivo `php5.conf` luce tal y como hemos visto significa que Apache será capaz de servir contenido php.

Tras esta comprobación ya estamos preparados para la instalación. La instalación de php es de lo más sencilla, con el siguiente comando tenemos instalado nuestro componente:

```
# aptitude install php5 php5-mysql libapache2-mod-php5
```

Tal y como hicimos anteriormente vamos a comprobar si nuestro servidor es capaz o no de mostrarnos contenido php. Para ello generaremos una pequeña aplicación php donde pediremos la información contenida en el `phpinfo`. Si todo funciona correctamente deberíamos ver un contenido como el que sigue:

PHP Version 5.6.19	
System	Windows NT COMPUTER 6.1 build 7601 (Windows 7 Professional Edition Service Pack 1) i586
Build Date	Mar 2 2016 20:00:17
Compiler	MSVC11 (Visual C++ 2012)
Architecture	x86
Configure Command	<pre> cscript /nologo configure.js "--enable-snapshot-build" "--disable-isapi" "--enable-debug-pack" "--without-mssql" "--without-pdo-mssql" "--without-pi3web" "--with-pdo-oci=c:\php-sdk\oracle\x86\instantclient_12_1\sdksdk\shared" "--with-oci8-12c=c:\php-sdk\oracle\x86\instantclient_12_1\sdksdk\shared" "--enable-object-out-dir=../obj/" "--enable-com-dotnet=shared" "--with-mcrypt=static" "--without-analyzer" "--with-pgo" </pre>
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	C:\Windows
Loaded Configuration File	C:\xampp\php\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20131106
PHP Extension	20131226
Zend Extension	220131226
Zend Extension Build	API220131226,TS,VC11
PHP Extension Build	API20131226,TS,VC11
Debug Build	no
Thread Safety	enabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled

Figura 14. Php info. (Extraído de commons.wikimedia)

Tras todo esto nuestra infraestructura LAMP está instalada y preparada para que empecemos a generar contenido.

Capítulo 5. Instalación de componentes para el diseño.

Durante el diseño de aplicaciones web son varias las herramientas que han de usarse para el diseño. No solo la infraestructura es importante, sino que también tenemos que asegurarnos de que nuestro entorno de trabajo sea eficiente y agradable para que la tarea del programador sea exitosa.

Entre las típicas herramientas que se usan en este tipo de desempeños encontramos:

- **Cliente FTP.** Mediante este tipo de herramientas podemos subir nuestros archivos al servidor.
- **Cliente de terminal remoto.** Gracias a ello podemos crear, eliminar o modificar archivos desde nuestro propio ordenador tomando control del servidor.
- **Editor de texto o IDE de programación.** Se hace necesario disponer de un editor de texto diseñado para la programación o un IDE de programación. En nuestro caso debido a que el código no ha de compilarse nos bastará con un editor de texto.
- **Software para el diseño de bases de datos.** Gracias a este tipo de programas podemos pasar del diseño relacional de la base de datos hecho sobre papel al código SQL que lo implementa sin ningún tipo de esfuerzo.

Veamos ahora en que consiste cada una de estas herramientas con un poco más de profundidad así como los programas que han sido utilizados en cada apartado.

5.1 Cliente FTP.

Un cliente FTP hace uso del protocolo FTP para conectar a un servidor FTP con la finalidad de transferir archivos de manera bidireccional en el servidor.

Algunos clientes FTP básicos están ya integrados en los sistemas operativos como Windows, Unix, etc. Sin embargo, existen múltiples opciones entre las que elegir de software libre muy útiles.

Entre la gran multitud de opciones que existen se decidió usar FileZilla por tres importantes razones: ya se tenía experiencia con ese programa, es software libre y su simplicidad de uso hace que sea una solución muy atractiva.

5.1.1 *FileZilla*

FileZilla es un cliente FPT multiplataforma de código abierto y software libre, con una licencia pública General de GNU. Entre los protocolos que soporta se encuentran: FTP, SFTP y FTP bajo SSL/TLS.

Entre las características más atractivas de este cliente FPT se encuentran:[8]

- **Administrador de sitios:** Esta característica permite que el usuario se cree una lista de sitios FTP o SFTP con sus datos de conexión, número de puerto, etc almacenados. Para un inicio normal se guarda el usuario y, opcionalmente, la contraseña.
- **Registro de mensajes:** Se encuentran en la zona superior de la ventana. FileZilla nos ofrece en forma de consola los comandos enviados y las respuestas que han sido enviadas desde el servidor.
- **Vista de archivo y carpeta:** En la parte central de la ventana tenemos una interfaz gráfica para FTP. Los usuarios tienen la posibilidad de navegar entre los distintos directorios, pudiendo ver y alterar el contenido tanto en la máquina local como en la remota, utilizando una interfaz tipo árbol de exploración. Los usuarios tienen la posibilidad de arrastrar y soltar los archivos desde la estructura de carpetas local a la remota ordenando de este modo la transferencia de los archivos.
- **Cola de transferencia:** Esta información reside en la parte inferior de la ventana donde podemos observar a tiempo real el estado de cada transferencia activa o en cola.



Figura 15. FileZilla. (Extraído de lignux.com)

5.2 Cliente SSH de terminal remoto.

Este tipo de clientes permiten la conexión a un host remoto mediante el protocolo secure Shell. Permite el manejo por completo de la computadora mediante un intérprete de órdenes. Además de la conexión a otros dispositivos, nos permite copiar datos de forma segura, gestionar claves RSA y la transmisión de datos de cualquier otra aplicación mediante un canal seguro tunelizado.

Si bien es cierto que la transferencia de archivos la vamos a realizar a través de un cliente ftp el uso de un cliente para terminal remoto no es baladí. Mediante el uso de este tipo de herramientas podemos trabajar directamente sobre los archivos alojados en el servidor en lugar de descargarlos modificarlos y volverlos a subir. Esto se vuelve muy importante cuando necesitamos modificar archivos de configuración, instalar programas o realizar operaciones que con unas pocas instrucciones en el terminal son llevadas a cabo.

Entre las distintas opciones que podemos encontrar se decidió elegir un programa basado en software libre como es Putty. Su simplicidad de uso y gran variedad de utilidades hacen que esta sea una opción interesante.

5.2.1 Putty

Putty es un cliente SSH, Telnet, rlogin y TCP raw con licencia libre. Originalmente solo era disponible para plataformas Windows aunque hoy en día lo vemos disponible también para

distintas distribuciones de Linux y la versión para máquinas Machintosh aunque aún no se encuentra disponible está siendo desarrollada.

Muchas son las características interesantes que putty implementa, así pues veamos algunas de ellas: [9]

- Almacenamiento de hosts y preferencias.
- Control de a clave de cifrado SSH y versión del protocolo.
- Control sobre el redireccionamiento de puertos con SSH.
- Emuladores completos de terminal xterm.
- Soporte para IPv6.
- Soporte para cifrado: DES, 3DES, AES, RC4 y Blowfish.
- Soporte de autenticación de clave pública.

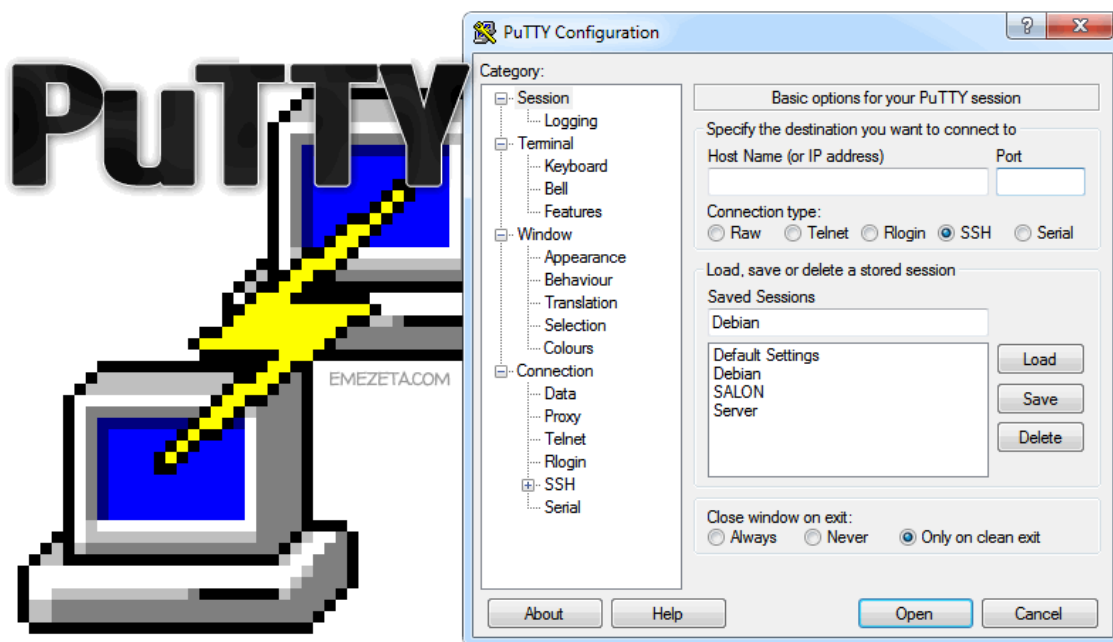


Figura 16. Putty.(Extraído de i.emezeta)

5.3 Editor de texto.

Para la edición de código se hace necesaria una herramienta que nos permita editar nuestro trabajo de un modo cómodo. Pese a que nuestro código será escrito en lenguajes que no requieren de compilación necesitamos de igual manera un editor que nos aporte diversas funcionalidades que nos hagan la tarea llevadera.

Se determinó como editor de trabajo el programa Sublime Text 2. Este software existe tanto para plataformas Windows, Machintosh o Unix. Por otro lado aunque la licencia de este no es gratuita, los desarrolladores determinaron que se podía usar de un modo free con una sola desventaja. La versión free obliga a cerrar un pequeño cuadro de dialogo emergente tras las 5 guardados y la versión de pago no tiene ese mensaje emergente. Salvo este pequeño matiz todos los componentes son accesibles haciendo que la versión gratuita solo difiera en una pequeña nimiedad ante la versión de pago.[10]

5.3.1 Sublime Text 2.

Sublime Text es un editor de texto pensado para aquellos programadores que necesitan no solo una bonita estética en su editor si no también multitud de ítems adicionales para hacer el trabajo más llevadero.

Veamos ahora una serie de características interesantes de este editor:[11]

- **Minimapa:** Se trata de una previsualización de la estructura del código, es algo muy útil para desplazarse por el archivo cuando se conoce bien la estructura de este.
- **Multi Selección:** Hace una selección múltiple de un término por diferentes partes del archivo.
- **Multi Cursor:** Crea cursores con los que podemos escribir texto a nuestro antojo en diferentes posiciones del archivo.
- **Multi Layout:** Trae siete configuraciones de plantilla. Se pueden elegir editar en una sola ventana o hacer una división de hasta cuatro ventanas verticales o cuatro ventanas en cuadrícula.
- **Soporte nativo para infinidad de lenguajes:** Soporta de forma nativa 43 lenguajes de programación y texto plano.
- **Búsqueda Dinámica:** Se puede hacer búsqueda de expresiones regulares o por archivos, proyectos, directorios, una conjunción de ellos o todo a la vez.
- **Auto completado y marcado de llaves.**
- **Soporte de Snippets y Plugins:** Los snippets son similares a las macros o los bundles además de la existencia de multitud de plugins.
- **Coloreado y envoltura de sintaxis:** Al escribir en cierto lenguaje de programación, se resaltan las expresiones propias de la sintaxis de ese lenguaje facilitando de ese modo la lectura.
- **Pestañas:** Se pueden abrir varios documentos y organizarlos en pestañas.
- **Resaltado de paréntesis e indentación:** Cuando el usuario coloca el cursor en un paréntesis, corchete o llave, resalta ésta y el paréntesis, corchete o llave de cierre o apertura correspondiente.

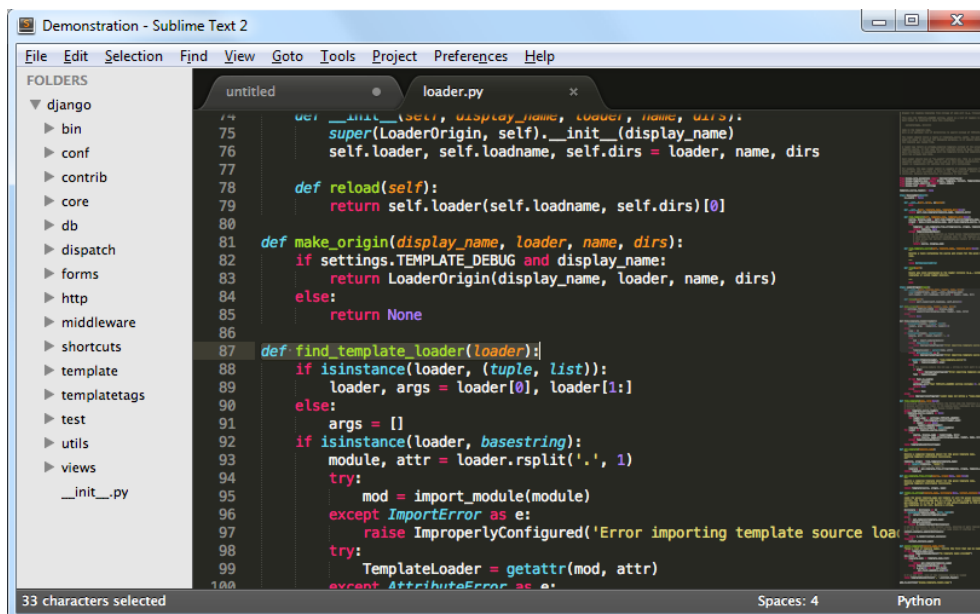


Figura 17. Sublime Text 2. (Extraído de sublimetext)

Sublime Text 2 tiene numerosos plugins a disposición del usuario que facilitan todavía más el trabajo. Entre ellos haremos referencia solo a uno, que sin duda fue al que más uso se le dio.

El plugin de Sublime Text 2 para conexiones sftp es uno de los más interesantes para nuestro trabajo. Mediante este plugin podemos editar nuestro código y cada vez que guardemos el archivo el propio Sublime subirá a nuestro servidor por nosotros esa versión por lo que en cada guardado tendremos nuestros componentes actualizados.

5.4 Herramientas para el diseño de bases de datos.

Para el diseño de la base de datos se decidió utilizar una herramienta gráfica. Gracias a este tipo de software podemos tras definir el modelo relacional sobre papel plasmarlo en el programa directamente. Para ello se decidió utilizar el programa MySQL Workbench que además de ser gratuito se había visto por el alumno en una de las clases impartidas en la ETSIT.

5.4.1 MySQL Workbench

[12] MySQL Workbench es una herramienta que a través de un interfaz gráfico nos permite el diseño, administración, creación y mantenimiento de bases de datos MySQL. Nuestro propósito para con este programa solo está enfocado al diseño puesto que una vez diseñada la base de datos se procederá a la exportación de la misma en código SQL. Dicho código será migrado a nuestro phpmyadmin con el que gestionaremos la base de datos del servidor.

Puesto que es una herramienta de uso sencillo y además gratuita es una genial opción para la tarea que nos ocupa.



Figura 18. MySQL Workbench, (Extraído en mysql)

Capítulo 6. Diseño y desarrollo.

En este capítulo se detallará todo el proceso de diseño y desarrollo tanto de la aplicación web como de la base de datos. En primer lugar veremos el diseño de la base de datos apoyándonos en el esquema del modelo relacional de la base de datos. Así pues, analizaremos por agrupaciones de tablas nuestra base datos. El modelo relacional estará acompañado por los detalles que se necesitaban cubrir por TRANSOFI S.L de este modo no solo veremos las relaciones existentes en la base de datos sino que también veremos el porqué. Tras todo esto se continuará con el modelo de diseño web que se ha seguido describiendo el funcionamiento de cada apartado.

6.1 Diseño de la base de datos

Empezaremos con las agrupaciones de tablas para ver por secciones y entender de un modo más simple la estructura de la misma apoyándonos en las necesidades de la empresa. Tras esta vista en detalle de la base de datos veremos la estructura final del modelo relacional para poder tener una visión de conjunto y ver cómo queda todo unido.

Veamos primero un concepto estrechamente relacionado con los archivos, las cajas y el mobiliario que es de gran importancia en la base de datos.

6.2 Etiquetas

En este caso veremos una imagen extraída de la página web, concretamente del módulo referido a la edición de etiquetas. Como ejemplo a ilustrar veremos un caso complejo entre las empresas clientes de TRANSOFI S.L. La empresa cliente en cuestión es la naviera MSC que debido a su estructura departamental y demás entresijos resultó un gran reto para el diseño de la base de datos.

En primer lugar en la parte superior de la etiqueta tenemos una primera zona con 3 campos. El primero es una abreviatura del nombre de la propia empresa. Esta abreviación es una manera muy visual de reconocer de qué empresa es el archivo. Tras ello vemos una referencia al departamento al que pertenece junto con el año al que pertenecen los documentos contenidos en el archivo. En la zona inmediatamente inferior a esto, vemos una definición de cabecera donde dependiendo del departamento veremos unas posibilidades u otras. El título es el siguiente componente. No todas las etiquetas de archivo necesitan de título pero algunos departamentos en concreto si, por lo que si no se rellenara esta opción la etiqueta tendría ese campo vacío sin mayor problema. Por último nos centraremos en el código de barras y en los campos que contiene la etiqueta.

El código de barras se compone de una serie de campos que hacen fácil la identificación unívoca del archivo. Además el formato con el que el código de barras se compone facilita el entendimiento de la información que nos proporciona la etiqueta de manera visual.

En primer lugar vemos la abreviación del nombre de la empresa seguido del año. Inmediatamente después se observa un código que identifica al departamento a modo de

abreviación. Los campos siguientes vienen referidos al número de archivos que tiene almacenados la empresa y número de archivos contenidos en el departamento.

Ya solo nos queda ver en cuanto al código de barras el formato que se ha seguido. El formato que se siguió fue el Code 128, se trata de un código de barras de alta densidad, usado ampliamente para la logística y paquetería. Con esta codificación se pueden representar caracteres alfanuméricos o solo numéricos y se pueden representar todos los caracteres de la tabla ASCII, incluyendo los caracteres de control. Otros formatos de códigos de barras fueron tomados en cuenta pero solo este tenía la capacidad de comprimir la información hasta tal punto que en cualquier caso podía caber en las dimensiones especificadas para la etiqueta.

Finalmente tenemos los campos que vendrán en la etiqueta. Estos campos vienen referidos a información que identifica a los archivos contenidos en el interior y que serán objeto de búsqueda por los clientes cuando quieran recuperar algún archivo en caso de no conocer otros datos más directos en cuanto al objeto de búsqueda. Habrá tantas filas de datos como documentos almacenados haya en el archivo.

MSC IMPORT 2016

(Campos en la cabecera) Fecha

TITULO

*Dejar en blanco si no se necesita


MSC2016IMP002400010001

Fecha	Buque	Viaje

...



Figura 19. Etiqueta

6.3 Tablas de la base datos.

6.3.1 Empresa, departamento, mobiliario y archivo.

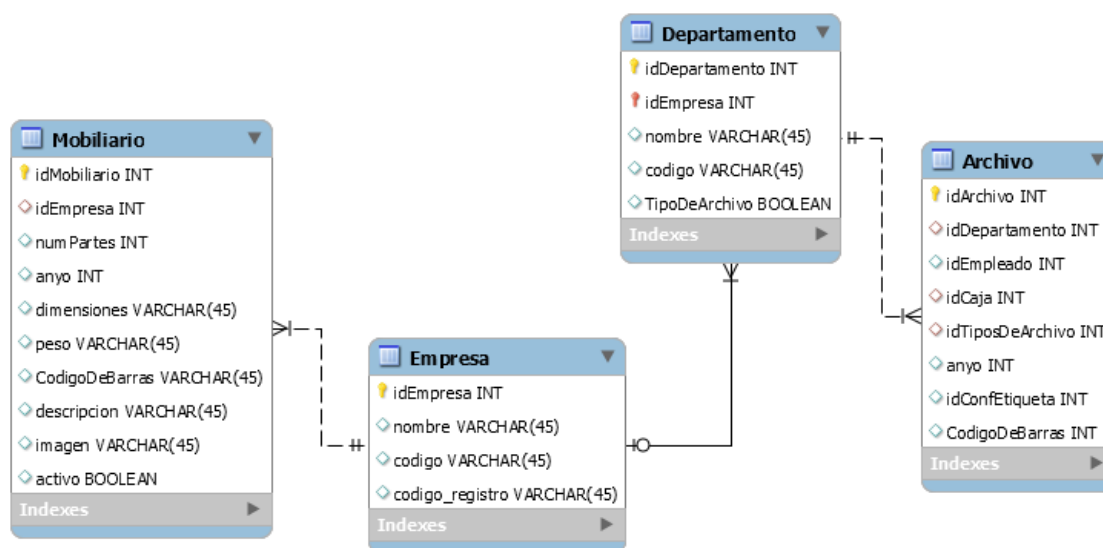


Figura 20. Empresa, departamento, mobiliario y archivo.

Las empresas están almacenadas en su propia tabla y como campos tenemos: idEmpresa, nombre, código, código_registro. La mayoría de campos no necesitan de mucha explicación pues su nombre es auto-descriptivo ahora bien veamos campos no tan evidentes.

El campo código viene referido a una abreviatura que se hace uso en la impresión de las etiquetas que vienen en los archivos y cajas. De este modo se almacena una cadena que abrevia e identifica de una manera visual a la empresa para su posterior uso en el etiquetado.

El campo código_registro viene referido al proceso de registro de las empresas en la aplicación. El registro de empresas es selectivo. Esto significa que cualquiera que quiera registrar su empresa primero ha de ponerse en contacto con TRANSOFI S.L para poder conseguir acceso.

Desde TRANSOFI S.L se procede al registro de la empresa y a la creación de un código el cual deberá ser introducido por el primer usuario que se registre. Este usuario tendrá un nivel de permiso elevado y podrá dar los primeros pasos en la configuración. En el siguiente apartado veremos en detalle la definición de los permisos.

Así pues una vez la empresa queda registrada, el primer usuario, ya puede proceder a registrarse activando de este modo la actividad de la empresa en la aplicación.

La tabla departamento como su nombre indica está referida a los distintos departamentos que puede contener una empresa. Como campos a destacar remarcaremos los campos código y TipoDeArchivo. El campo código identifica al departamento del mismo modo que el campo código identificaba a la empresa por cuestiones de etiquetado. El campo TipoDeArchivo nos identifica la clase de archivador con los que trata el departamento, dependiendo de si son archivos Z o definitivos nuestro campo TipoDeArchivo estará a un valor u otro.

Los archivos los cuales están directamente relacionados con el departamento y por consiguiente con la empresa, puesto que el departamento está relacionado directamente con la empresa, hacen referencia a los archivos generados por los distintos departamentos de la empresa cliente. En la tabla Archivo vemos campos que necesitan de explicación y que ahora detallaremos.

El campo idCaja nos relaciona al archivo con la caja que lo contiene, así pues si el archivo acaba de ser generado y aún no pertenece a ninguna caja tendrá un valor nulo. Una vez ha sido realizada toda la recogida de archivos y se prepara la etiqueta para la propia caja se podrá actualizar el valor de ese campo mediante la aplicación desarrollada para la máquina lectora de códigos de barra.

Otros campos a destacar son el campo CodigoDeBarras el cual contiene el código de barras que se ha generado para ese archivo en formato alfanumérico. Por último tenemos el campo idConfEtiqueta que demomento se va a dejar de lado puesto que para ilustrar el significado de ese campo conviene ver más tablas.

Finalmente vemos la tabla mobiliario. TRANSOFI S.L además de almacenaje de archivos también ofrece la posibilidad de almacenar otro tipo de enseres como por ejemplo mobiliario de oficina. Dicho mobiliario se desmonta y almacena adecuadamente. Por ello vemos campos como numeroDePartes en el que se especifica en cuantas partes se dividió el mueble para que en el momento de la recuperación del mueble o sencillamente de movimiento del mismo se pueda tener un mayor control sobre él.

Otros aspectos son almacenados en esta tabla como por ejemplo una breve descripción, el peso del mismo o las dimensiones. También se guarda una referencia el directorio del servidor donde están almacenadas imágenes del mueble por si los propios clientes no recordaran exactamente de qué se trataba. Por supuesto como todo lo que proviene de los clientes necesita de un código de barras tal y como ilustra el campo CodigoDeBarras.

6.3.2 Empresa, departamento, empleado y atributos del empleado.

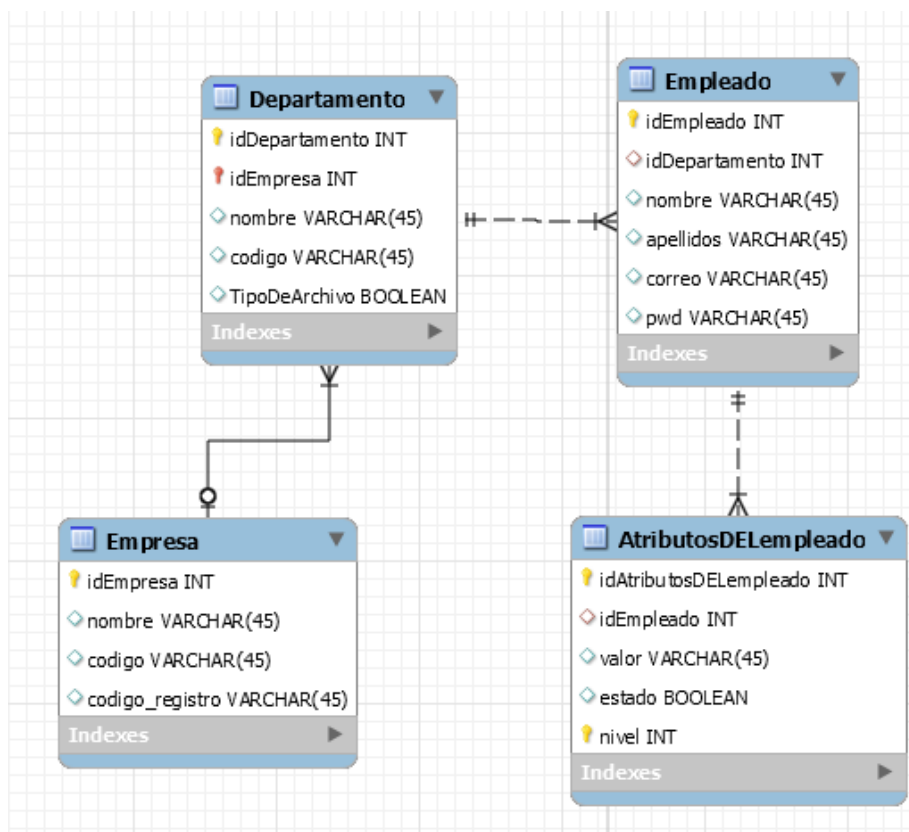


Figura 21.1.1.1 Empresa, departamento, empleado y atributos del empleado.

En esta porción de la base de datos queda representada la estructura típica de una empresa la cual tiene departamentos y su vez esos departamentos tienen empleados. Pasaremos a explicar la parte relacionada con los empleados puesto que de las otras dos tablas ya se ha hablado.

La tabla empleados contiene campos que no requieren de mayor explicación solo destacar que contiene el campo idDepartamento que es el que genera esa relación directa con la tabla departamento. Lo más interesante de la información que se almacena acerca del empleado está en la tabla AtributosDEEmpleado. En esta tabla podemos ver una serie de campos muy interesantes.

El campo valor, que es un de tipo de variable de caracteres, es un campo en el que se puede almacenar algún aspecto a tener en cuenta del empleado. Este campo se dejó para posibles futuros usos, puesto que al ser una cadena de caracteres da una mayor flexibilidad. El campo estado hace referencia a si el empleado que estamos tratando sigue o no sigue contratado. Esa variable estará a cero u a uno en caso de estar en la plantilla activa o de haber sido despedido.

Por último queda hablar de quizás el campo más interesante de todas estas tablas, el campo nivel. Existen 5 niveles que nos indicarán el nivel de privilegio del usuario y por lo tanto determinarán que cosas se les permite hacer. Estos 5 niveles son:

- **Nivel 1:** Este usuario es el de menor nivel. Estamos en este caso cuando el usuario se ha registrado pero el administrador de la empresa aún no le ha dado el alta en la aplicación web. Hasta que el usuario supervisor no realice esta operación dicho usuario no podrá realizar ningún tipo de operación.
- **Nivel 2:** El usuario relacionado con este nivel de permiso es el usuario típico que usará la plataforma. En este caso tenemos un empleado que pertenece a un departamento y realiza las etiquetas, órdenes de trabajo relacionadas con su departamento o la búsqueda de archivos.
- **Nivel 3:** Se corresponde con el usuario supervisor de la empresa cliente. Este usuario da de alta las cuentas de usuario, puede realizar órdenes de trabajo elevadas como por ejemplo las órdenes de destrucción o las órdenes libres y puede generar nuevos tipos de etiquetas. Como añadido puede ver toda la información almacenada de los distintos departamentos.
- **Nivel 4:** Se trata del empleado de TRANSOFI S.L. En cuanto a la página web no tiene ninguna relación con la misma. Ahora bien, ha de quedar reflejado en las órdenes de trabajo tanto el empleado de la empresa cliente que realizó la demanda del servicio como el nombre del empleado de TRANSOFI S.L.
- **Nivel 5:** Usuario de mayor nivel, se trata del gerente de TRANSOFI S.L. Este usuario tiene un rol el cual le permite registrar empresas, eliminar las y hacer las funciones de usuario supervisor de cualquier empresa si fuera menester.

Todos estos niveles van creciendo en cuanto a permisos. Esto significa que el usuario de nivel 3 tiene los permisos del de nivel 2 más una serie de permisos adicionales que le permiten hacer cosas que el usuario de nivel 2 no puede hacer. La excepción a esto es el usuario de nivel 4 que solo está definido por la necesidad de mantener un histórico de quien hizo qué.

6.3.3 Archivo, tipos de archivo, caja y trazabilidad.

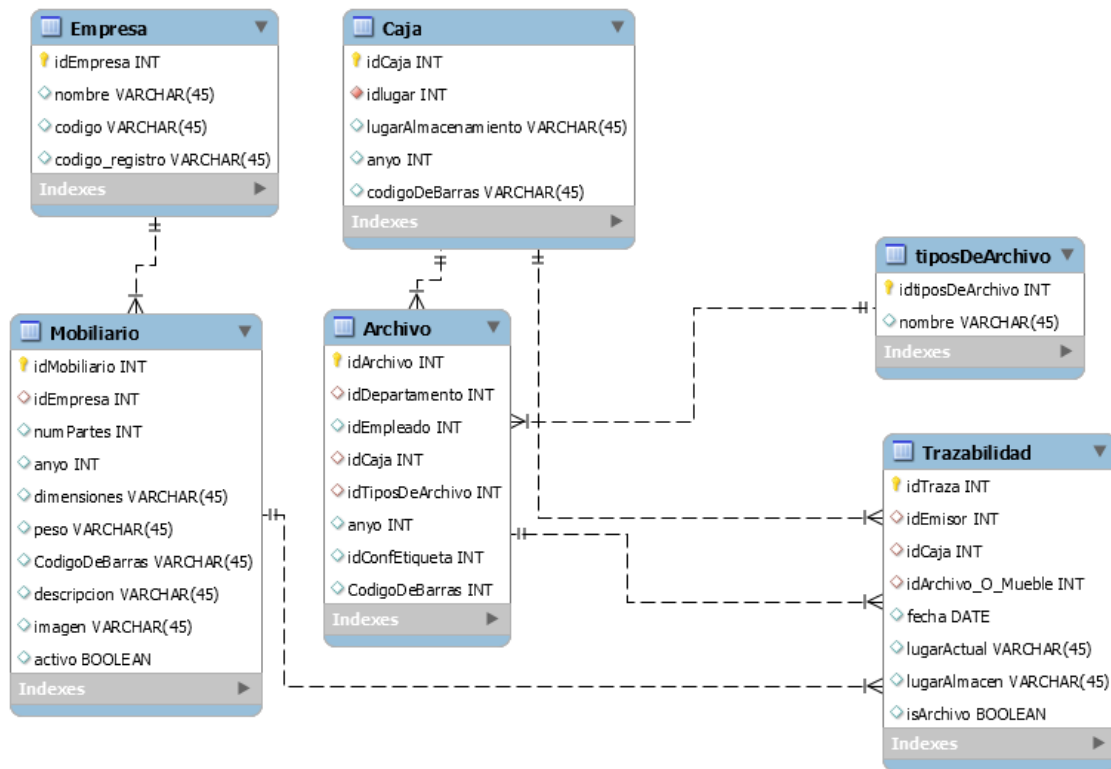


Figura 22. Tablas archivo, tipos de archivo, caja y trazabilidad.

En esta porción del modelo relacional de la base de datos vemos las relaciones en cuanto a los archivos, cajas y mobiliario. En cuanto al mobiliario la única novedad que observamos es que está relacionada con la tabla Trazabilidad. Dicha tabla está también relacionada con los archivos y las cajas.

La idea de guardar un registro de ello es realizar una traza en la cual se pueda ver el trayecto que ha ido realizando el objeto. Aunque pueda parecer que los campo lugarActual y lugarAlmacen hacen referencia al mismo dato, no es así. El campo lugarActual está relacionado con el lugar en el que está el objeto esto es: almacén uno, almacén dos, empresa cliente u otro emplazamiento que quisiéramos referenciar.

Por otro lado el campo lugarAlmacen referencia a la posición exacta en que se encuentra el objeto, por ejemplo: Fila 7 columna 3.7 Como se explicó anteriormente la finalidad de empaquetar documentos en archivos es almacenar todos estos en una caja que luego se guardará en un almacén. Dicha caja al igual que los archivos y el mobiliario almacenado tiene un código de barras asociado.

Por último destacar la tabla tiposDeArchivo que está relacionada con Archivos. Como ya se dijo los archivos pueden ser de dos tipos definitivos o Z. Se creyó conveniente crear una tabla aparte para dejar constancia de que tipo de archivo se trataba. Si bien es cierto que las empresas clientes solo trabajan con este tipo de archivos podría surgir la necesidad de crear un nuevo tipo. De este modo la opción queda abierta a un posible futuro con distintos modelos de archivos.

6.3.4 Configuración de la etiqueta, unión de la etiqueta con el campo, configuración del campo, tipo, valor del campo select y fila.

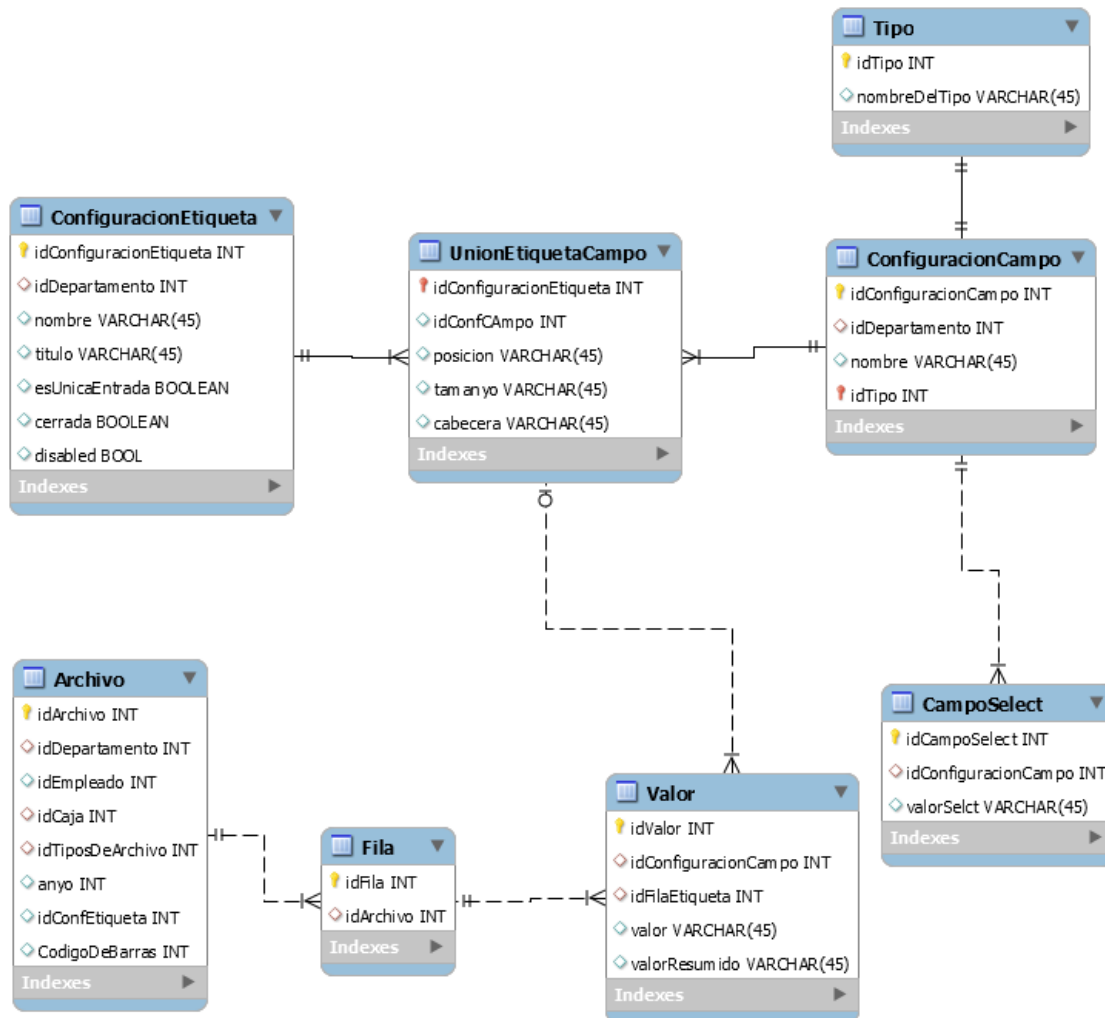


Figura 23. 6.3.4 Configuración de la etiqueta, unión de la etiqueta con el campo, configuración del campo, tipo, valor del campo select y fila.

Cómo ya se expuso anteriormente el modelo de la etiqueta que va adherido al archivo, caja o mueble es variable por lo que se debe poder dar el formato que el departamento guste. Por ello se eligió crear una serie de tablas que permitieran esta edición junto con el almacenaje del modelo para su posterior uso. La configuración de la etiqueta como podemos ver en la tabla correspondiente va ligada al departamento y por ende a la empresa puesto que departamento y empresa son tablas con una relación directa.

Así pues veamos de qué modo se diseñó esta estructura. El objeto de generar etiquetas es almacenar la información que el departamento de la empresa cliente crea necesaria. Así pues, la configuración de la etiqueta debe estar estrechamente ligada con el departamento. Cada departamento ha de tener la libertad de tener varios modelos de etiquetaje si así se requiere. Tal y como vemos en la imagen la tabla ConfiguracionEtiqueta está relacionada directamente con la tabla departamento mediante el campo idDepartamento. A su vez, como cada etiqueta puede contener más de un campo y como cada campo puede estar en diversos modelos de etiquetas como resultado tenemos una relación entre tablas n a n . Esto es una problemática que en las bases de datos SQL no puede darse. La solución para este tipo de casos es la creación de una tabla intermedia que relacione estas dos tablas generando dos relaciones 1 a n con la tabla intermedia. Todo esto se refleja en las tablas ConfiguracionEtiqueta, UniónEtiquetaCampo y ConfiguraciónCampo donde la tabla UniónEtiquetaCampo es este nexo de unión generado para evitar esa relación directa n a n .

Adicionalmente en la tabla UniónEtiquetaCampo se han añadido una serie de campos que fijan parámetros relacionados con la posición, tamaño y la definición de una cabecera si la hubiere. Además en esta tabla tenemos una relación que direcciona al valor del campo en concreto. En la tabla ConfiguraciónCampo tenemos valores relacionados con el nombre del campo del que se trata y con un idTipo que nos relaciona a la tabla Tipo. La tabla tipo nos indica de qué tipo es el campo. Por ejemplo si el campo es una fecha tendrá un identificador numérico y el nombre del tipo el cual será date. La idea de crear esta tabla surgió durante la programación de la página web al ver que era necesario tanto para la introducción de datos como para la edición de la etiqueta saber de qué tipo se trataba el campo.

La inserción de valores mediante cualquier tipo de software difiere mucho entre tipos distintos. Un claro ejemplo es el de las fechas donde para insertar una fecha el usuario debe seleccionar de un calendario desplegable la fecha en concreto. Para poder averiguar de qué tipo de dato se trata y realizar una correcta representación en HTML se hacía necesario el uso de una tabla que nos indicara el tipo del campo.

Junto con la tabla ConfiguraciónCampo tenemos relacionada la tabla CampoSelect. ¿Qué ocurre cuando tenemos un campo select, esto es una lista de valores desplegable en html? Esta lista de valores ha de poderse desplegar de forma adecuada para que el usuario elija entre las opciones posibles. Se han de poder recoger todos los posibles valores desde la base de datos y mostrarlos como un campo select en html. Para ello en la tabla tipos existe un campo en concreto que hace referencia a si es o no una lista de valores. En caso de ser una lista de valores, sabemos que se trata de un campo select y que deberemos listar todos los posibles valores. Ese campo tendrá una lista de valores donde el identificador del campo select, idSelect será distinto pero el identificador de configuración del campo será el mismo y el valor será el que corresponda. De este modo podemos obtener una lista de posibles valores.

Toda esta información ha de ser introducida por los usuarios clientes durante la configuración de la etiqueta y una vez se haya introducido serán capaces de generar etiquetas con los valores que ellos deseen. Como hemos visto hemos cubierto cualquier tipo de campo a introducir sea de la tipología que sea.

En caso de no ser una lista de valores significa que el valor ha de ser introducido ya sea escrito en caso de caracteres y números o seleccionado de un calendario en el caso de fechas. Esto significa que ha de hacerse una representación correcta para la introducción de la información. En este tipo de casos el usuario ha de introducir el dato de forma manual. Ese dato introducido manualmente se almacena directamente en la tabla valor donde se relaciona la fila de la etiqueta con el archivo correspondiente.

Una vez comprendida la primera parte de esta estructura donde ya tenemos la posibilidad de introducir datos veamos cómo se realiza la creación de contenidos para una etiqueta que irá adherida a alguno de los componentes. Cada etiqueta se compondrá de una serie de filas con cierta información, de ahí la creación de la tabla Fila. En esta tabla se relaciona el archivo con cada una de las filas de la etiqueta contenedoras de información relativa al archivo. Esta tabla está relacionada con una tabla que se ha nombrado como Valor. En esta tabla se ha rellenado de la siguiente forma las entradas:

- El identificador del valor, idValor, es un número autoincremental y por lo tanto distinto para cada entrada.
- El identificador de configuración de etiqueta, es decir idConfiguraciónCampo que nos relaciona la tabla con UniónEtiquetaCampo donde podemos ver valores tales como posición, tamaño y cabecera. Se han establecido estos campos en esa tabla puesto que para un mismo modelo de etiqueta estos han de ser iguales. Por lo que añadiéndolos a la tabla UniónEtiquetaCampo evitamos la redundancia de valores.
- El identificador de la fila de la etiqueta, idFilaEtiqueta que nos indica a que número de fila corresponde manteniendo así un orden.
- Los campos valor y valorResumido donde se almacena el valor o un resumen del valor si fuera necesario.

6.3.5 Orden de trabajo de mobiliario, órdenes de trabajo, asociación de la orden de trabajo y orden de proceso.

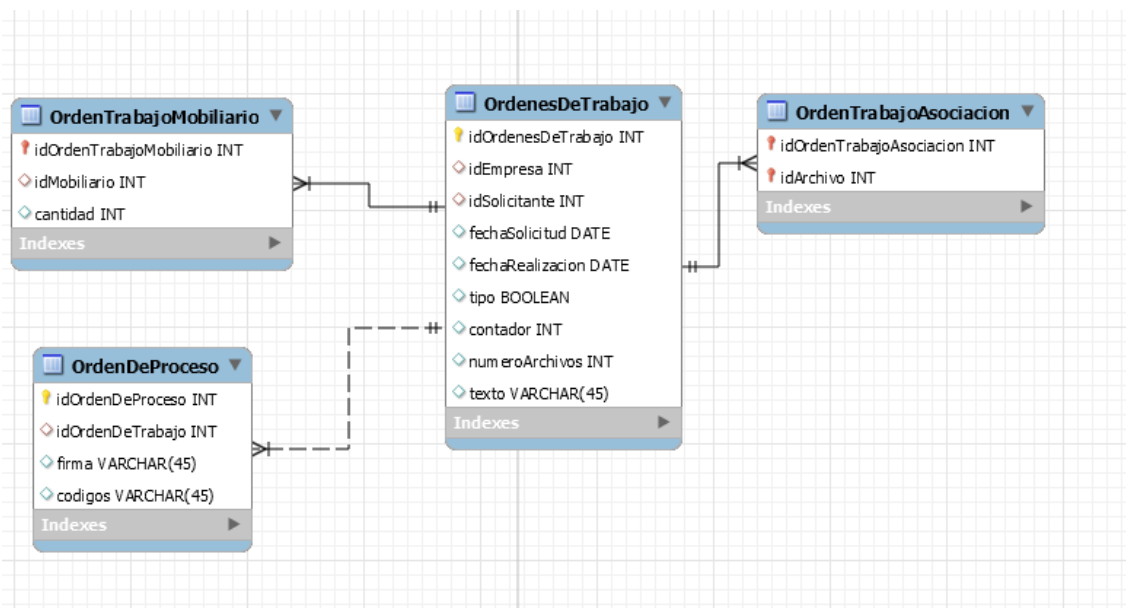


Figura 24. Orden de trabajo de mobiliario, órdenes de trabajo, asociación de la orden de trabajo y orden de proceso.

Las órdenes de trabajo no son otra cosa que una manera forma de pedir un servicio. El problema que había anteriormente, antes de la aplicación web, era que se realizaba un escrito sin formato ninguno y con los datos que al demandante del servicio le parecían adecuados. Esto resultaba a menudo en peticiones incompletas donde faltaban datos por lo que la operación no se podía realizar. Otras veces los datos introducidos eran suficientes para realizar la petición pero el propio cliente olvidaba otras demandas por lo que al final todo resultaba en una cantidad de correos enviados enorme, con la consecuente pérdida de tiempo por las dos partes. De este modo se decidió diseñar un modelo cerrado para las órdenes de trabajo, o por lo menos parcialmente cerrado. Tenemos tres modelos de órdenes de trabajo:

- Órdenes de recogida y Órdenes de recuperación. Estas dos son modelos cerrados de orden donde se requieren una serie de datos específicos.
- Órdenes libres. Estas órdenes de trabajo consisten en un textbox en el cual se especifica que se quiere exactamente, se usan solo para casos especiales y solo el usuario administrador puede generar este tipo de comandas.

Las órdenes van dirigidas a archivos y a mobiliario por ello tenemos un campo booleano donde estando este a cero u a uno fijamos si se trata de un archivo o de un mueble. Los datos necesarios para cada caso son introducidos en la base de datos y una vez ello se ha generado se procede a mostrar un pdf con los datos que ha introducido el usuario. Ese documento se firma y una vez firmado se envía a TRANSOFI S.L por correo para que se pueda proceder. Una vez la orden ha sido procesada se genera una entrada en la tabla OrdendeProceso dejando constancia de ello.

6.3.6 Modelo relacional de la base de datos

Una vez detallado el funcionamiento de la base de datos por partes veamos cómo queda todo unido.

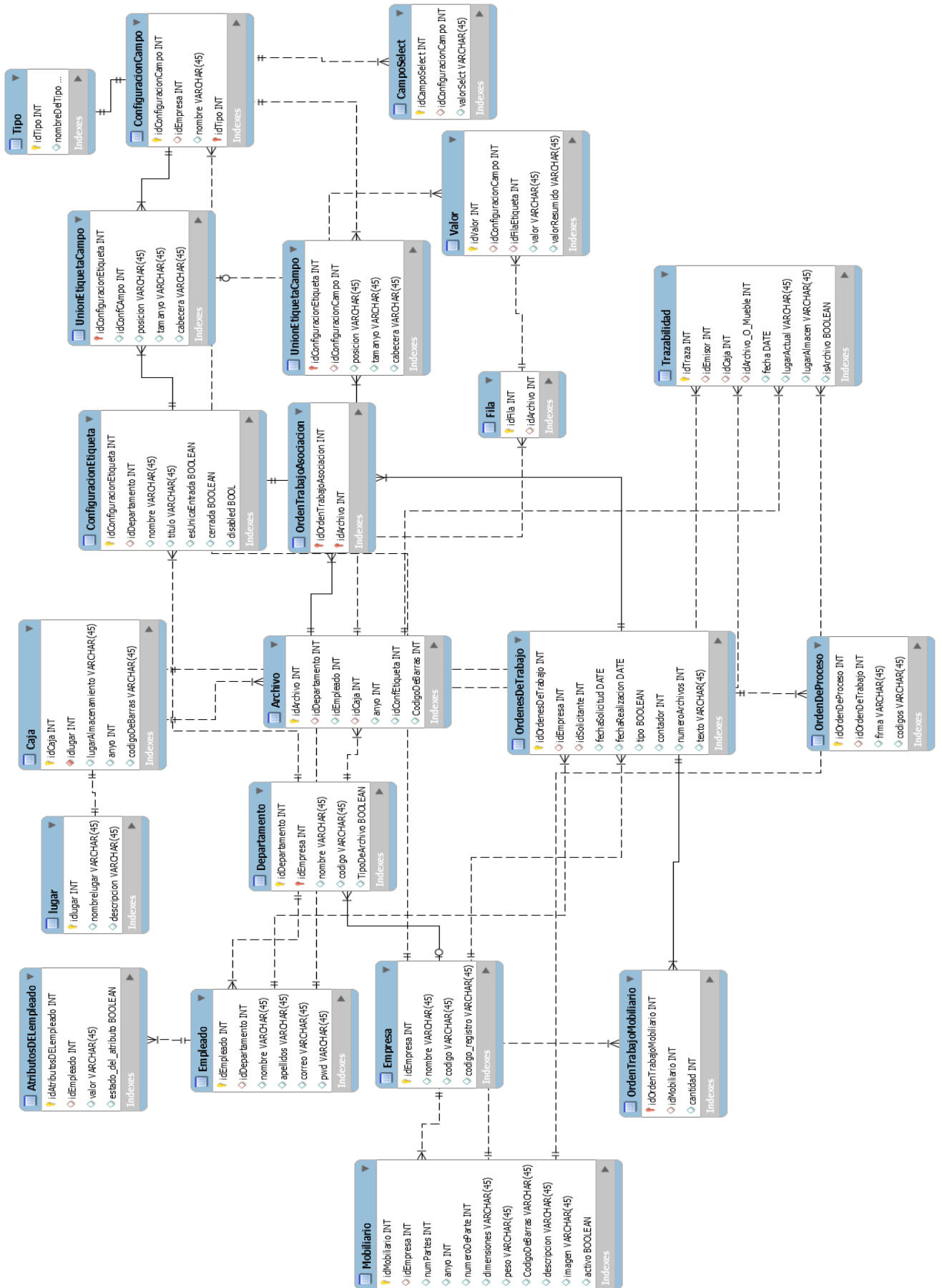


Figura 25. Diseño de la base de datos.

Una vez finalizado el diseño de la base de datos nuestro programa de diseño de bases de datos nos permite desde el modelo gráfico exportar el trabajo en formato de sentencias sql. Con nuestro archivo sql que contiene la estructura de nuestras tablas no tenemos más que acceder a phpmyadmin y crear una nueva base de datos. En esa base de datos ejecutaremos todas las sentencias sql que se han exportado. De esta forma tan sencilla se pudo exportar el trabajo anterior.

6.4 Desarrollo web.

En el desarrollo de nuestra aplicación podemos distinguir dos estados. Un primer estado en el que se carga la parte gráfica de la página web y un segundo estado en el que se debe hacer una o varias consultas a la base de datos para luego mostrar los datos extraídos de la misma. Lo que se acaba de mencionar es muy importante porque define perfectamente la estructura que vamos a seguir para la arquitectura de directorios. Veremos detalladamente en el siguiente apartado como se desarrolla.

6.4.1 Estructura del árbol de directorios y funcionamiento inicial.

En primer lugar se carga la página web y como solo se trata de cargar componentes gráficos los únicos recursos que van a necesitar ser cargados son los correspondientes al directorio *bloc*. En caso de que necesitemos realizar algún cambio en la base de datos o realizar algún tipo de petición se hará a través de formularios. Estos formularios por ser entorno gráfico también son cargados a través de los archivos que residen en el directorio *bloc*.

Ahora bien, cuando necesitemos modificar información de la base de datos se utilizarán los archivos del directorio *proc*. Estos archivos solo contienen una serie de funciones en código .php por lo que al solo ejecutar código en el lado del servidor y realizar las funciones pertinentes no generamos ningún tipo de contenido html visible para el usuario.

Una vez las funciones requeridas han sido ejecutadas se almacena en una variable de sesión el resultado de la operación. En el momento en que se termina de procesar todo lo mencionado se prepara el header. En este header todos los valores que se enviaban a través de los métodos get o post son eliminados. De este modo volvemos a llamar mediante este header al *bloc* que carga la información que se requiere en este momento. Esta información puede ser en el caso de un formulario que se ha rellenado y enviado un mensaje de aviso con información tal como éxito u error.

Con este modo de operar obtenemos una serie de resultados que ahora expondremos:

- **Separación entre parte gráfica y funciones.** De este modo tenemos un código más ordenado y fácil de mantener.
- **Solución al problema de reenvío de información.** Al presionar: F5, no se produce el reenvío del formulario. Lo mismo es aplicable al uso de los botones para desplazarse hacia delante o atrás del navegador. Evitamos de ese modo entradas de información duplicada en la base de datos.
- **Bloques pequeños de código.** Gracias a este modo de trabajo modular, tenemos los elementos bien separados y organizados. A través de incluir en el archivo los módulos necesarios conseguimos la funcionalidad deseada en esa página. Como gran ventaja tenemos una fácil escalabilidad del código, reutilización del mismo y mantenimiento eficiente.

Veamos ahora en modo gráfico como es la estructura de directorios:

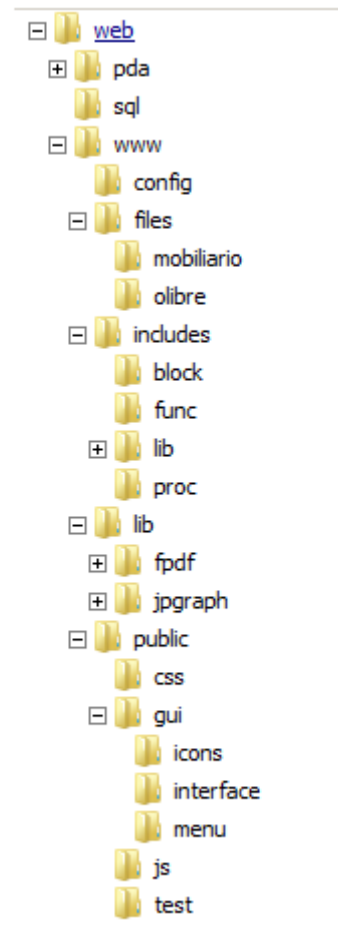


Figura 26. Estructura de directorios.

Tal y como podemos ver los directorios *proc* y *bloc* de los que hemos hablado se encuentran alojados en el directorio *includes*, nombrado así porqué todos los archivos que han de incluirse en otros archivos están en este directorio. El resto de directorios como por ejemplo *lib* hacen referencia a las librerías externas que se utilizan en el código como *fpdf* y *jpggraph*.

Como acabamos de ver el esquema de funcionamiento sigue este modelo:

1. Se carga la página inicial.
2. El usuario elige entre todas las opciones hacia cual quiere dirigirse.
3. Se carga el contenido pedido por el usuario.
4. El usuario rellena una serie de campos que se mandarán hacia el servidor.
5. En lugar de cargar entorno gráfico se ejecutan las funciones necesarias en el directorio *proc* para realizar las modificaciones solicitadas por el usuario.
6. Se almacena en una variable de sesión la información con el resultado de la operación.
7. Se genera un nuevo header para la siguiente carga de contenido donde se hará uso de las variables de sesión para mostrar la información resultante de la operación.

Para clarificar más el modo de funcionamiento, se ilustrará a continuación una figura.

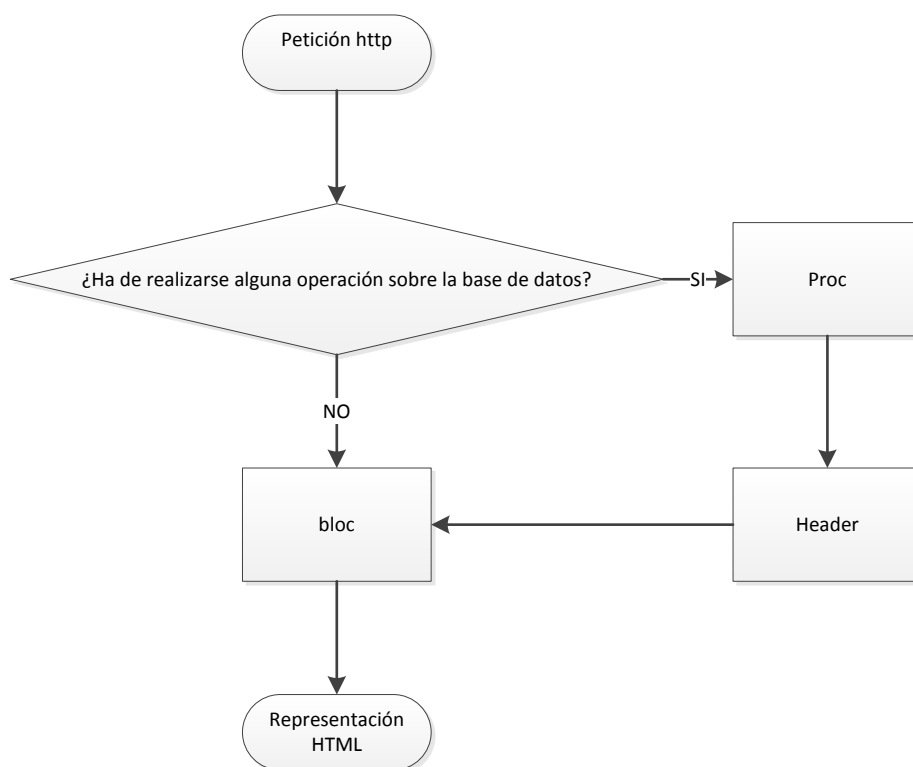


Figura 27. Flujograma de peticiones http.

El directorio al que tienen acceso los usuarios mediante las peticiones http es el directorio public. Dicho directorio contiene los archivos de configuración de estilos css, la configuración de interfaz gráfico gui, archivos de javascript almacenados en el directorio js y archivos .php.

Nuestra operativa de diseño hace que tengamos un solo archivo php donde se centre todo el contenido. El resto de módulos php son pequeños módulos en los que se han programado pequeñas funcionalidades que son cargadas en este archivo index.

Así pues cuando elegimos acceder a cierto recurso de la página web lo que realmente ocurre es que cuando se hace la petición se hace una llamada al index otra vez pero con un parámetro introducido mediante el método get de html.

De este modo tenemos una composición que nos permite a base de cargar los módulos correspondientes modificar el contenido de la página. Esto se traduce en una serie de ventajas tales como:

- **División del problema en pequeños módulos** haciendo más fácil la obtención de la solución.
- **Mejora del mantenimiento del código.** Ante cualquier cambio o eventualidad, solo habría que cambiar el código alojado en los pequeños módulos php que se encargan de realizar ciertas tareas. Por otro lado si lo que sucede, no es un cambio de funcionalidad sino de estética cambiando el estilo en el archivo css y en el archivo index.php está solucionado.
- **Evita la réplica de código.** Todo está contenido en un mismo archivo.
- **Mayor escalabilidad.** Para añadir un módulo solo tenemos que empezar a diseñarlo. Una vez acabado ya estará disponible para usarlo en cualquier momento.
- **Reutilización de código.** Al dividir el problema en pequeños módulos si en algún momento necesitamos usar algún módulo no tenemos más que llamarlo.

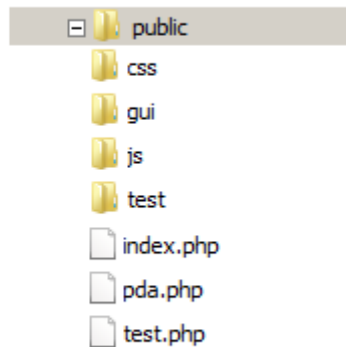


Figura 28. Directorio public.

6.4.2 Módulos *php*.

En esta sección se pretende explicar cómo funcionan los distintos módulos, páginas, que se programaron en la página web. Los módulos más interesantes son los que hacen referencia a operaciones realizadas hacia la base de datos, estos son los archivos *php* contenidos en el directorio *proc*.

Tal y como se ha comentado antes, los archivos *php* contenidos en el directorio *bloc*, hacen referencia al aspecto gráfico. Nosotros nos vamos a centrar en la funcionalidad de la página web y no tanto en el diseño.

Durante toda la navegación de la página hay diversos puntos en común que son:

- **Comprobación de credenciales.** El objetivo es ver si el usuario está registrado y tiene los permisos correspondientes para acceder al recurso que intenta acceder en todo momento.
- **Consultas a la base de datos.** Tanto para mostrar datos referentes al usuario como para ir almacenando en variables de sesión datos de la cuenta de usuario, se hacen consultas a la base de datos.
- **Inserción en la base de datos.** Cada envío de formulario como es lógico resulta en una actualización de la base de datos.
- **Eliminación.** Estamos hablando de: el proceso de destrucción, despido de empleados, etc. Los archivos y otros no son borrados de la base de datos, sencillamente su estado en la base de datos cambia.

6.4.2.1 Login

El proceso de inicio de sesión en la página consta de un formulario muy simple donde se debe indicar el correo y la contraseña.

Para dotar de una mayor seguridad se cifra la contraseña. En primer lugar se hace una codificación en sha256 de la contraseña introducida. Seguidamente se hace un hash del resultado. Finalmente codificamos el resultado anterior en md5 para luego volver a hacer un hash de ese resultado. Esta operación da una cadena de caracteres que mucho difiere de la contraseña por lo que un ataque contra ello no es fácil de realizar.

En la base de datos las contraseñas se guardan cifradas mediante el método anteriormente descrito. Por ello aunque se capturara la contraseña cifrada al intentar introducirla en el formulario de registro el resultado del cifrado sería otra cadena también diferente de la contraseña original. De este modo evitamos posibles robos de credenciales a través de la red.

Tras el cifrado de la contraseña se hace una consulta sql donde se comprueba que el par usuario y contraseña son correctos devolviendo el identificador de empleado y el estado de la cuenta, activa o inactiva.

Existen dos casuísticas a contemplar:

- 1) Las credenciales introducidas son incorrectas. El resultado de ello es un mensaje de error donde se nos solicita introducir un usuario y contraseña correctos.
- 2) Las credenciales introducidas son correctas, pero la cuenta está inactiva. Si la cuenta no está activa aparecerá un mensaje de error donde nos indicará que nuestra cuenta no está activada.
- 3) Las credenciales introducidas son correctas y la cuenta está activa. Se procede a realizar otra consulta a la base de datos donde extraeremos datos relevantes de usuario como: identificador del departamento, identificador de empresa, nivel de permisos, etc.

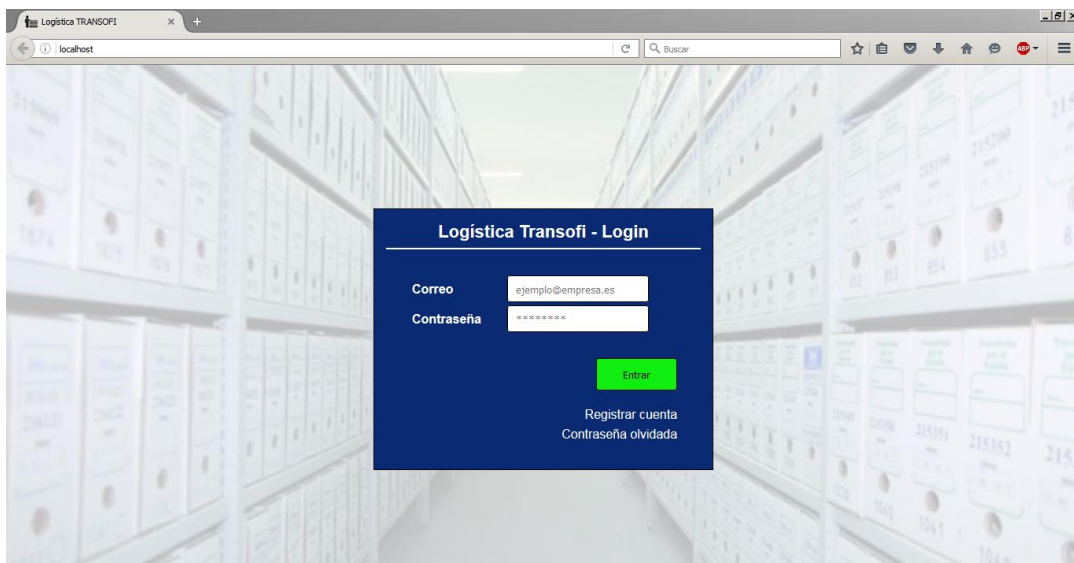


Figura 29. Inicio de sesión.

6.4.2.2 Logout

La opción de desconexión de la página es accesible desde cualquier punto de la página web. El proceso es muy sencillo se trata de destruir todas las variables de sesión asociadas a la página web de TRANSOFI S.L que se han ido generando a lo largo de la navegación. Cómo podemos ver en la siguiente imagen donde acabamos de hacer login la opción de desconectar siempre queda en ese marco sea cual sea la página donde nos encontremos.

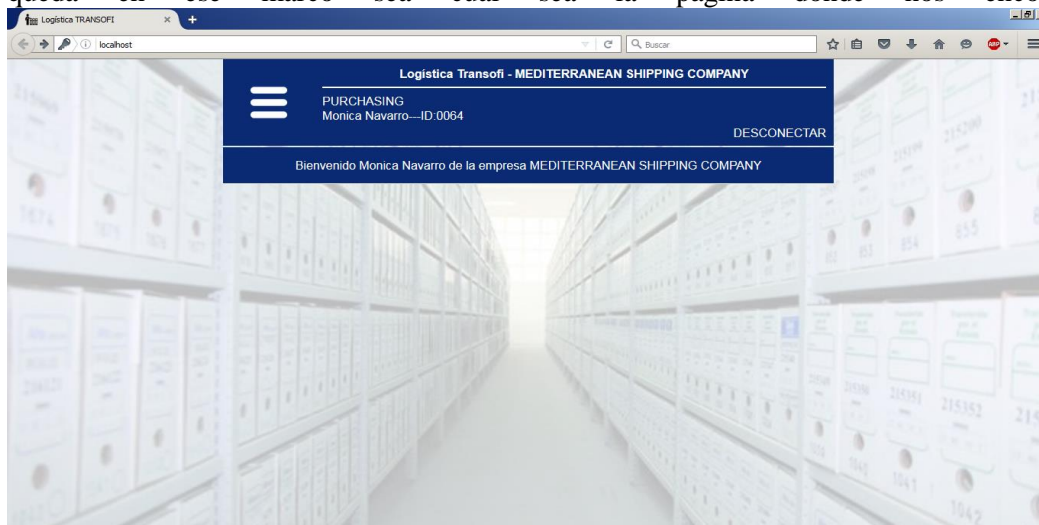


Figura 30. Logout.

6.4.2.3 Crear etiquetas

Ya se introdujo durante la explicación del funcionamiento de la base de datos el funcionamiento de la creación de etiquetas. Veamos ahora pues, como son los módulos que integran este proceso.

Aprovechando la tesitura introduciremos los módulos relacionados con los menús de opciones. Dichos menús tienen un número de opciones variable según el nivel de permisos que tenga el usuario. Como es lógico el usuario con permisos más elevados tendrá más opciones entre las que operar. En la siguiente imagen veremos cómo es un panel de opciones de un usuario que tiene la cuenta activa y que no es usuario administrador.

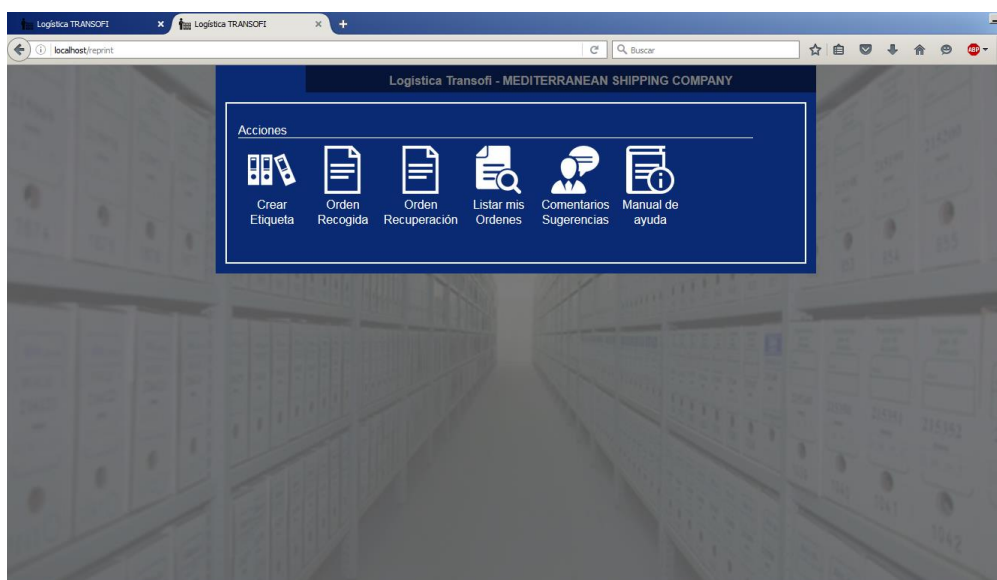


Figura 31. Panel de opciones.

Para generar las etiquetas debemos pulsar en el apartado crear etiquetas. Tras ello cargará una página con posibles etiquetas ya existentes si las hubiera. Una vez hemos elegido que modelo de etiqueta debe tener nuestro archivo veremos una página como la siguiente, donde ya se han introducido algunos datos a modo de ejemplo de funcionamiento.

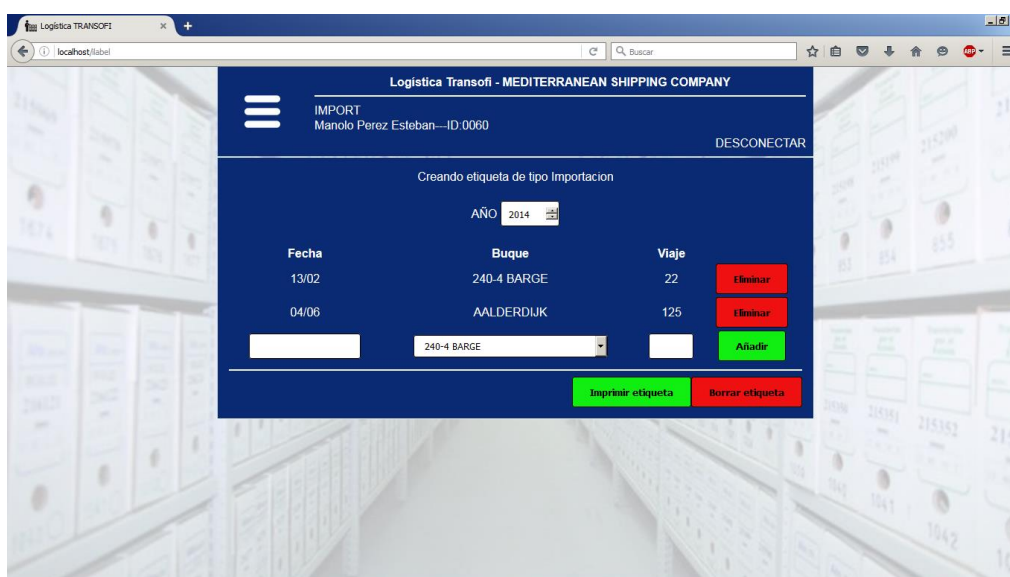


Figura 32. Generación de etiqueta para archivo.

La programación de este módulo consiste en un módulo donde en primer lugar se han cargado las opciones que se deben rellenar. Dichas opciones son dependientes del modelo de etiqueta elegido anteriormente por lo que ese valor se envía a la siguiente página para que pueda hacer la

consulta a la base de datos y podemos cargar los componentes correctos. Como vemos en el ejemplo tenemos la opción de ir generando dinámicamente las entradas e incluso eliminarlas.

Como es de suponer, php no es capaz de realizar este tipo de operaciones puesto que es un lenguaje del lado del servidor. Por ello en este tipo de casos se tuvo que usar el lenguaje JavaScript que está en el lado del cliente. Al igual que php, JavaScript se incrusta en el código html con una serie de etiquetas que declaran que el contenido siguiente es de dicho lenguaje.

Una vez todos los datos han sido introducidos se procede a cerrar el trabajo presionando el botón imprimir etiqueta. Si el contenido se ha generado correctamente veremos como resultado una página tal que:

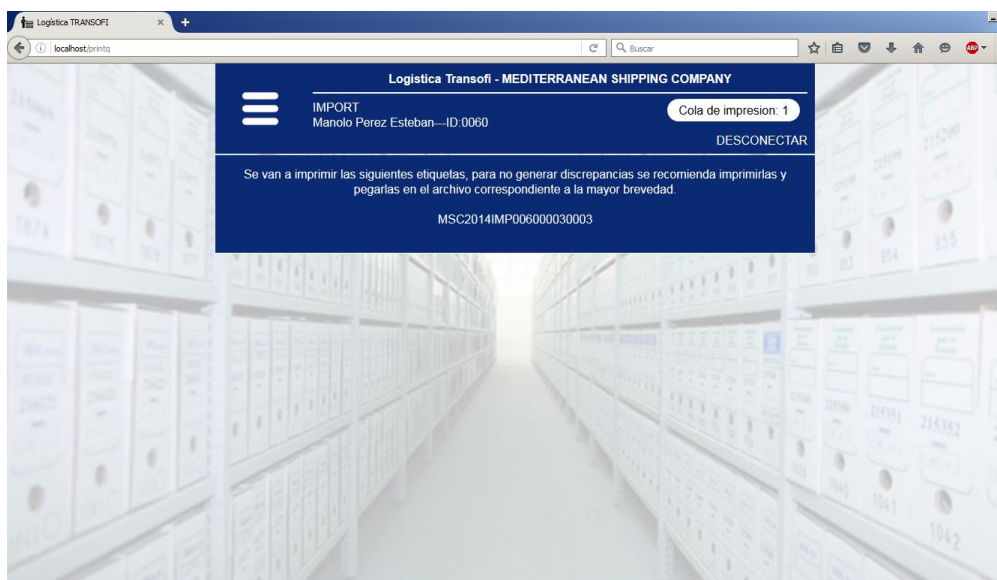


Figura 33. Etiqueta generada correctamente.

Tal y como vemos en la esquina superior derecha hay un botón que reza: Cola de impresión. En él se refleja el número de etiquetas generadas que quedan pendientes de impresión. Si pulsamos sobre ese botón se procede a generar tantos documentos pdf como etiquetas tengamos en la cola. La construcción de un archivo pdf en php se realizó con la librería fpdf incluida en el directorio libs. Dicha librería permite mediante coordenadas x e y escribir un fichero pdf tal y como queramos mediante php. Así pues se prepararon una serie de plantillas para dar un formato inicial que luego iría relleno con la información que se introdujo en las etiquetas.

La idea de la generación de etiquetas se concibió como un trabajo que se inicia y se acaba sin parar. Por supuesto se pueden hacer muchas más cosas una vez generadas las etiquetas, pero hay que imprimir las etiquetas generadas antes de cerrar la sesión. Esto es debido a que se quiso evitar que los empleados generaran contenidos repetidos en la base de datos ya sea por descuido, omisión u olvido.

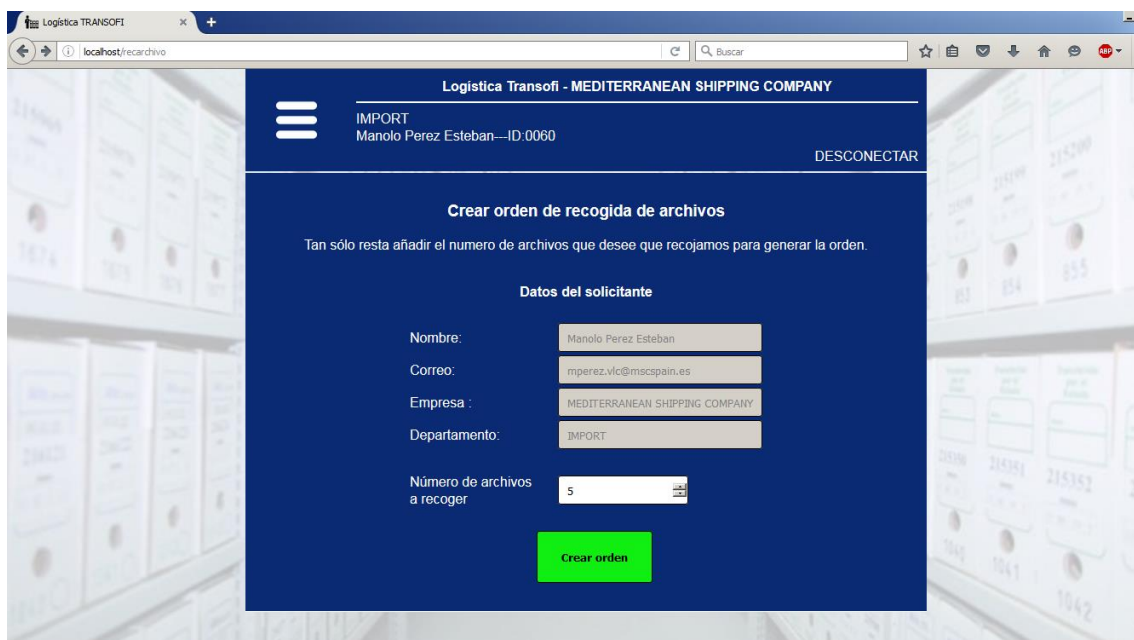
Los datos a imprimir se almacenan en un vector en forma de variable de sesión y una vez se pulsa el botón para imprimir estos se imprimen y su contenido se libera. Si el usuario se desconectara de la sesión este trabajo se perdería puesto que todas las variables de sesión se destruyen en el proceso de desconexión.

6.4.2.4 Generar orden de recogida o de recuperación

La orden de recogida sucede cuando el cliente ha generado archivos junto con sus etiquetas. Tras ello es el momento de generar la orden de recogida.

El módulo que recoge esta función es un pequeño fragmento de código php que da formato a la página y que sencillamente recoge en formato de formulario la información. Los datos que constarán en la orden de recogida contienen información que ha sido recuperada desde las

variables de sesión generadas tras el inicio de sesión. El único dato que falta indicar es el número de archivos que han de ser recogidos. Tras la creación de la orden otro archivo php genera con esa información enviada una actualización en la base de datos dejando constancia de la orden de recogida. Finalmente se compone con dicha información un correo electrónico que va dirigido al gerente de TRANSOFI S.L para que se atienda a la mayor brevedad esa tarea.



Logística Transofi - MEDITERRANEAN SHIPPING COMPANY

IMPORT
Manolo Perez Esteban—ID:0060

DESCONECTAR

Crear orden de recogida de archivos

Tan sólo resta añadir el número de archivos que desee que recojamos para generar la orden.

Datos del solicitante

Nombre: Manolo Perez Esteban

Correo: mperez.vlc@mscspain.es

Empresa: MEDITERRANEAN SHIPPING COMPANY

Departamento: IMPORT

Número de archivos a recoger: 5

Crear orden

Figura 34. Creación de la orden de recogida.

Cuando los empleados de TRANSOFI S.L llegan a las instalaciones y recogen los archivos escanean los código de barras. Las máquinas lectoras de códigos de barras son pdas con posibilidad de conexión a internet por lo que, una vez se escanean dichos códigos la información del archivo en cuanto a su localización cambia.

Se programó una aplicación en el lenguaje de programación C# que atiende a este tipo de necesidades por parte de uno de los empleados. Puesto que el alumno no participó de ese desarrollo se deja esta referencia al uso de este dispositivo sin entrar en detalle en lo que atañe a esa parte de la programación.

Las empresas clientes también necesitan recuperar los archivos por lo que para atender a ello se generó el módulo php que soluciona la recuperación de archivos. Sencillamente hay que introducir el número del código de barras del archivo que queremos recuperar o un espacio en blanco resultando esto en un listado de todos los archivos del departamento.

En ocasiones los propios clientes, ante el volumen de archivos generados, no saben a ciencia cierta si los archivos se encuentran en la empresa o en las instalaciones de TRANSOFI S.L. Esto ocurre cuando tras generar las etiquetas no se genera la orden de recogida, cosa que no podemos controlar puesto que cuando generar la orden de recogida queda en manos de los clientes.

En caso de encontrarse en la empresa cliente el resultado es una página donde vemos información relacionada con el archivo junto con una etiqueta que nos dice que el archivo no está en las instalaciones de TRANSOFI S.L.

Si el archivo que se quiere devuelta no está en la empresa cliente, veremos una página como la siguiente



Figura 35. Creación de orden de recuperación.

Como vemos, la opción de añadir a la solicitud está disponible. Se pueden añadir tantos archivos de recogida como se deseen solo hay que ir realizando búsquedas e ir añadiendo a la solicitud los archivos deseados. Para finalmente realizar la petición de recuperación solo hay que pulsar en la opción cerrar orden donde ello nos re-direcciona a un archivo php que actualiza la base de datos para dejar constancia de que la orden se ha realizado y se envía un correo con dicha información al gerente de TRANSOFI S.L. Tras esto si todo ha sucedido sin ningún problema aparece un mensaje que nos informa de que la operación se ha podido realizar correctamente.

6.4.2.5 Comentarios, sugerencias y manual de ayuda.

Restan dos opciones que seleccionar: un pequeño manual de ayuda y un buzón de sugerencias.

El manual consta de unas breves y sencillas instrucciones que ayudan a los usuarios noveles en la aplicación a manejarse con la web.

Por otro lado los usuarios pueden realizar comentarios y sugerencias. Se trata de un pequeño text-box en el que introducir información por parte de los usuarios para conseguir su feedback.

El envío de esa información consta de un módulo en php que recoge la información introducida y la formatea en un correo electrónico dirigido al gerente de TRANSOFI S.L. Es muy importante que los usuarios remitan problemas ya sean con la página o con la interfaz. El diseño de aplicaciones web consiste en un producto que inicialmente tiene unas características que luego al ser testeadas por los usuarios cambian según sus gustos. Los usuarios finales son los que van a utilizar la aplicación y se debe fomentar y dar herramientas para que puedan comunicar sus experiencias y problemas.

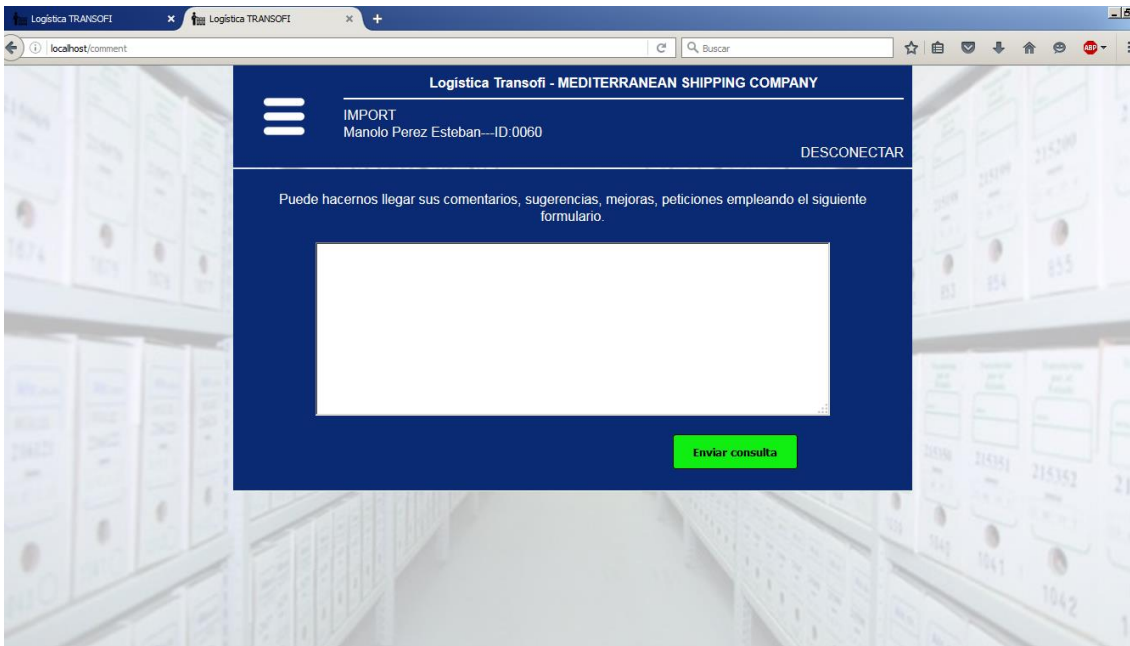


Figura 36. Sugerencias y comentarios.

Se han descrito las funcionalidades que tienen los usuarios de nivel 2, usuarios con cuenta activa que no son supervisores. Resta atender a las funcionalidades desarrolladas para los usuarios supervisores de la empresa y la del gerente de TRANSOFI S.L.

6.4.2.6 Menú de usuario supervisor.

Los usuarios supervisores se encargan de la gestión de: usuarios, departamentos y el listado de órdenes adicionalmente también tienen la posibilidad de listar las órdenes y realizar estadísticas. Un usuario que es supervisor además de las opciones mencionadas anteriormente también tiene la potestad de generar etiquetas nuevas así como de generar órdenes libres. Dichas órdenes no podían ser solicitadas por los usuarios regulares solo el usuario administrador tiene esa libertad. El resto de opciones que componen el abanico son las mismas que las del usuario raso sin ningún tipo de diferencia.

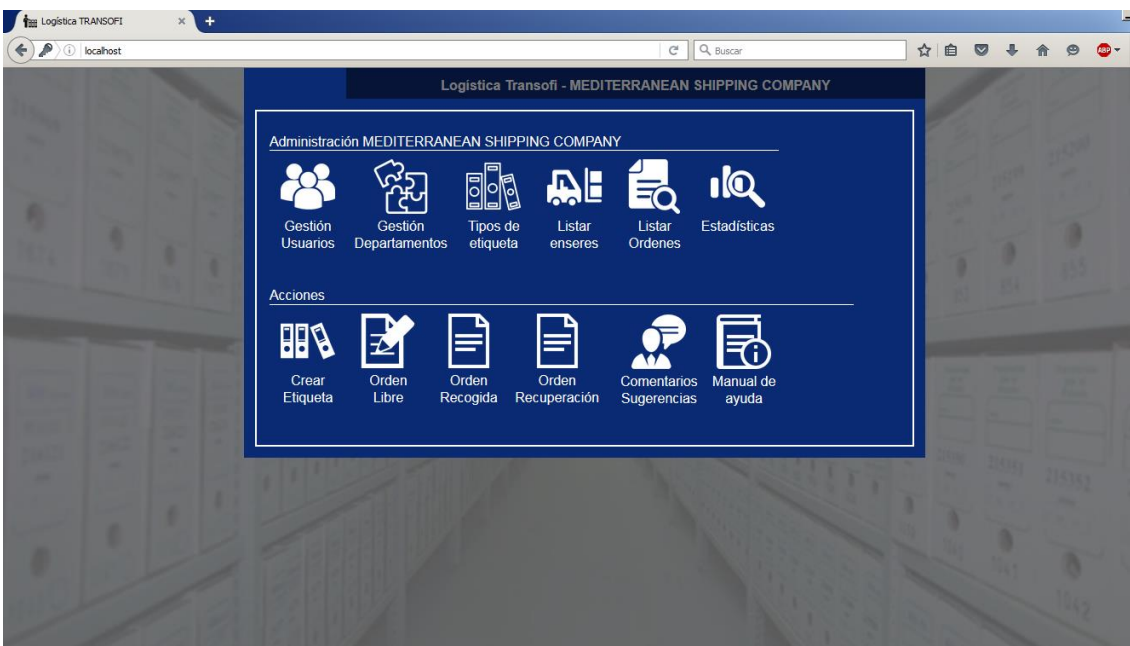


Figura 37. Menú opciones del usuario supervisor.

6.4.2.7 Gestión de usuarios

La gestión de usuarios permite tanto el registro de nuevos usuarios de forma manual como la edición de permisos o datos de la cuenta por parte del administrador. Tal y como veremos en la imagen existe un código que es autogenerado cuando se realiza el registro de la empresa. Ese código debe introducirse antes de proceder al registro del usuario. Si el usuario quiere registrarse debe proveer ese código durante el proceso de registro. Como es lógico pensar, el usuario administrador si dispone de dicho código por lo que ha de proveer a todos los usuarios que quieran registrarse dicha información. Una vez el usuario ha quedado registrado es tarea del administrador editar el campo que referencia a si la cuenta del usuario está activa. Cuando la cuenta está activa el usuario podrá acceder a la página web y trabajar.

En la siguiente captura de pantalla veremos cómo es el proceso de listar todos los usuarios de la empresa junto con la opción de añadir nuevo usuario:

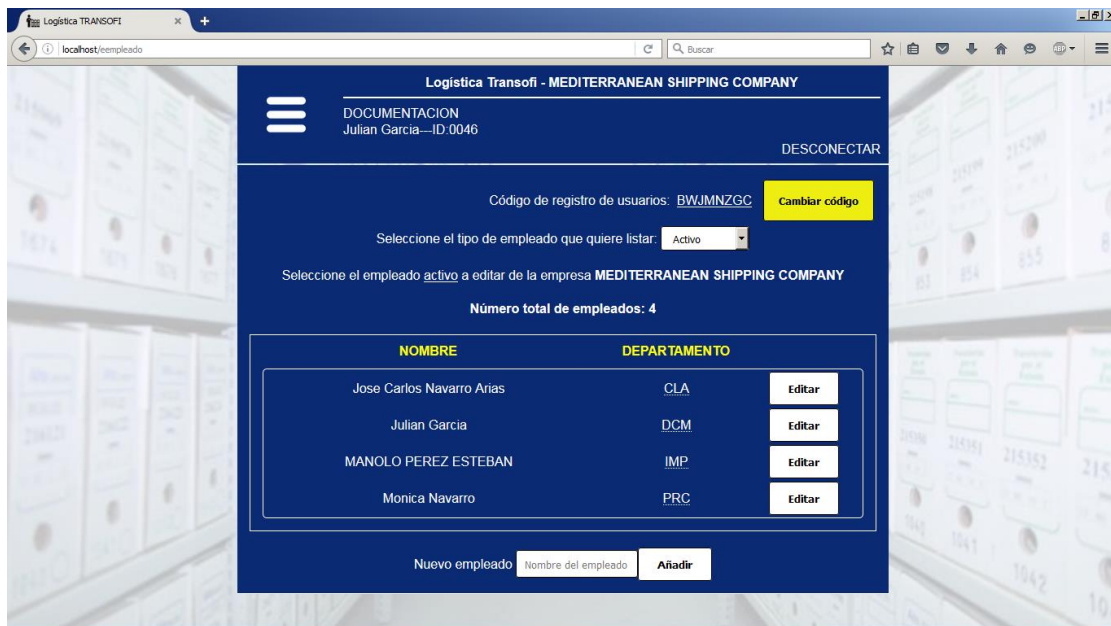


Figura 38. Gestión de usuarios.

El usuario administrador puede editar cualquier aspecto de la cuenta del resto de usuarios con solo presionar el botón editar. Como veremos el administrador es capaz de editar cualquier aspecto configurable de la cuenta incluido el nivel de permisos y el estado de la cuenta.

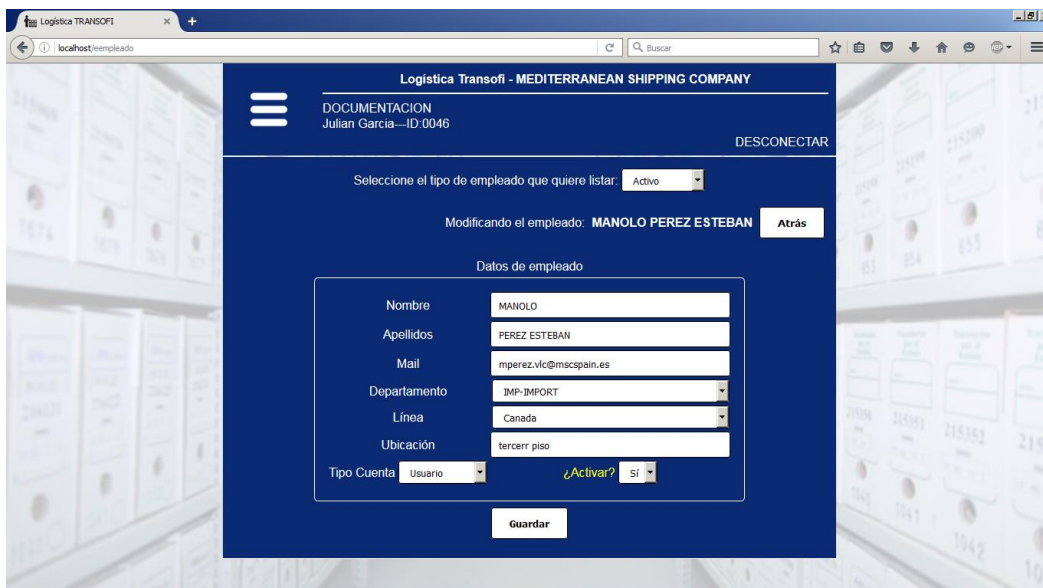


Figura 39. Editar cuenta de usuario.

6.4.2.8 Gestión de departamentos

Cuando una empresa es registrada en la aplicación junto con su usuario administrador lo primero que hay que hacer es generar los departamentos a los que los usuarios posteriores pertenecerán. El usuario administrador tiene la obligación de generar nuevos departamentos y editarlos.

Este es el menú de creación de nuevos departamentos donde tenemos todos los departamentos listados junto con la opción de crear un nuevo departamento.

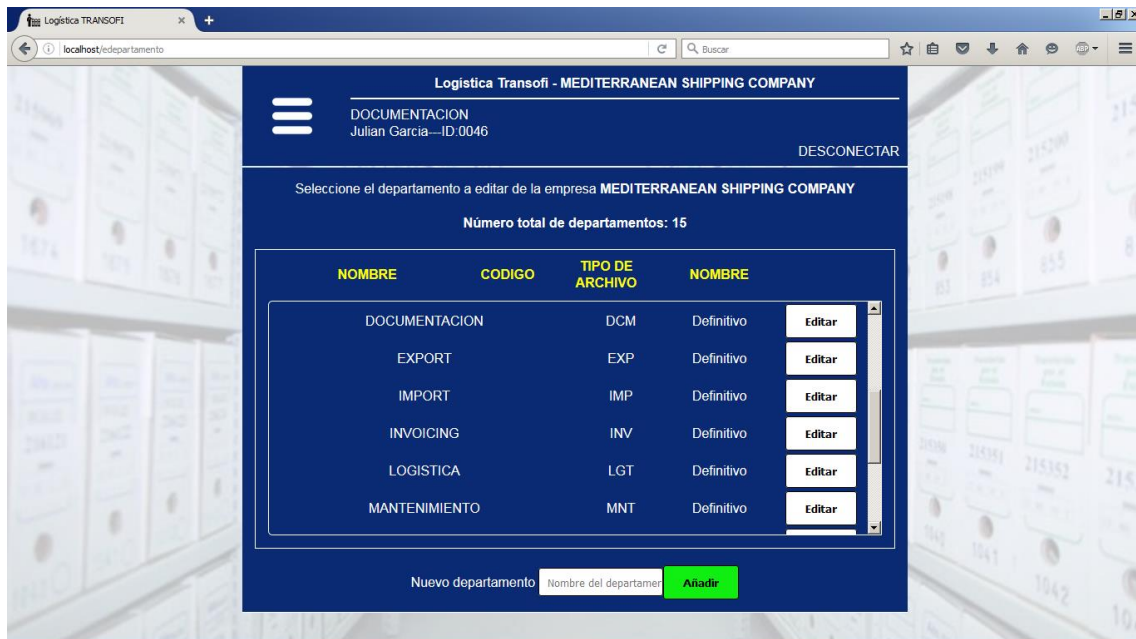


Figura 40. Gestión de los departamentos.

Cuando creamos el departamento debemos establecer una serie de parámetros a introducir. El tipo de archivo con el que el departamento trabajará, el tipo de impresora para el etiquetaje correspondiente y el código del departamento. Todos estos datos son introducidos en la base de datos y usados con asiduidad por lo que deben ser introducidos todos sin excepción.

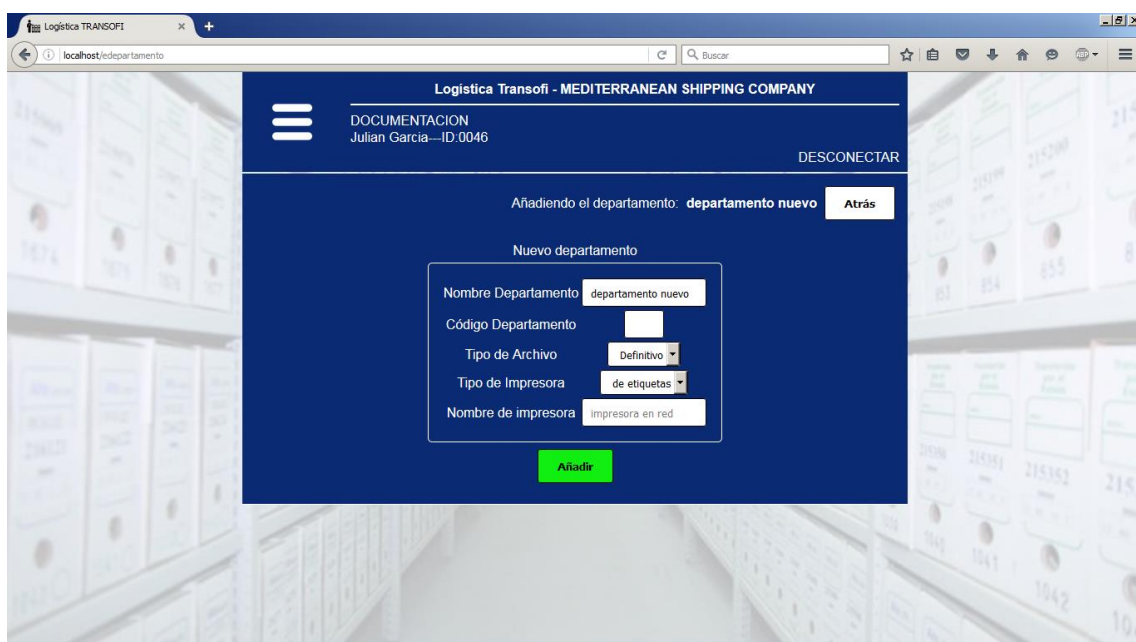


Figura 41. Crear departamento.

6.4.2.9 Edición de las etiquetas

La creación de las etiquetas es una parte muy importante que hay que configurar antes de generar los archivos. Antes de poder crear un archivo se debe generar el tipo de etiqueta y para ello se diseñó el editor de etiquetas.

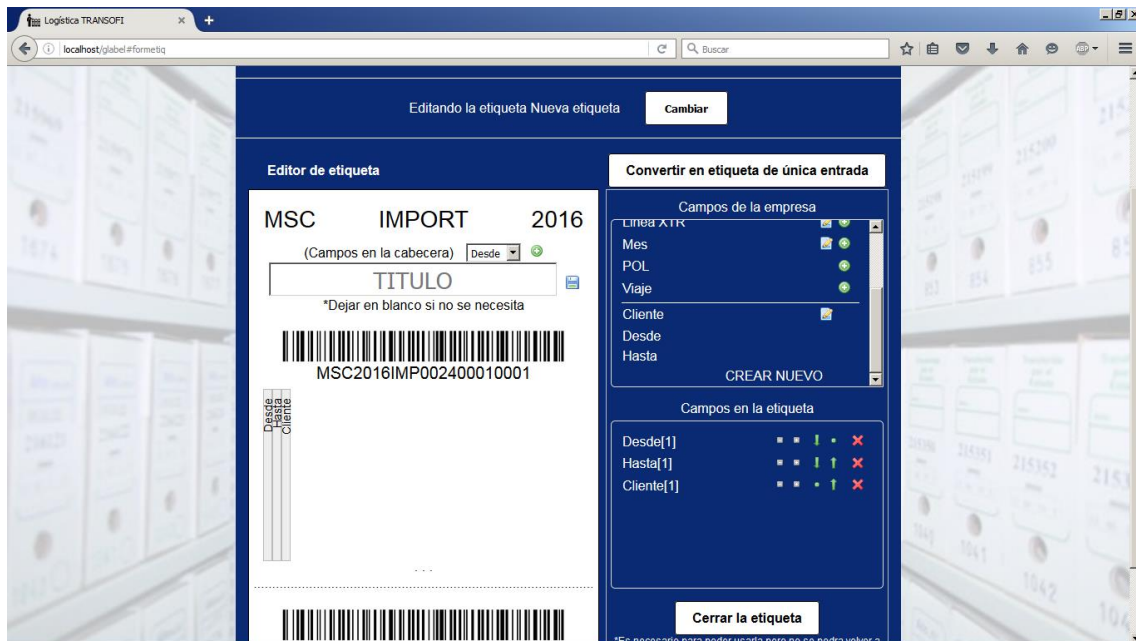


Figura 42. Editor de etiquetas.

Este editor se realizó en lenguaje JavaScript casi por completo. La adición de campos se hace de manera dinámica por lo que cada vez que añadimos un campo nuevo en la etiqueta este se ve reflejado en las zonas de campos de la etiqueta así como en la representación gráfica de la etiqueta. Una vez la etiqueta ha quedado al gusto del usuario, se debe cerrar la etiqueta si se quiere usar. Esto hace que una vez cerrada la etiqueta no se pueda modificar. Ahora bien siempre podemos generar diversas etiquetas asociadas a un departamento mediante el editor. El usuario administrador genera las etiquetas correspondientes a los diferentes departamentos por lo que debe conocer los archivos con los que la empresa se maneja y su etiquetaje.

6.4.2.10 Listado de órdenes y de enseres

Revisar las órdenes de trabajo fue uno de los aspectos a tener muy en cuenta por parte de las empresas clientes. Como es lógico se propuso la generación de un módulo donde se pudiera revisar todas las órdenes de trabajo ya sea introduciendo el código referente a la orden, una palabra clave o un elemento vacío para listar todo.

Se puede revisar también el pdf de la orden de trabajo. Cuando se pulsa en la orden se recuperan todos los datos necesarios para componer los documentos desde la base de datos. Inmediatamente, esta información es atendida por un módulo de php componedor de pdfs. Mediante una serie de plantillas y la ayuda de la librería fdpf se genera el documento el cual está listo para descargar e imprimir si se desea.

Como ya se dijo uno de los servicios que también oferta TRANSOFI S.L es el almacenaje de mobiliario de oficinas. El mobiliario que se almacena de la empresa se puede consultar en el apartado listado de enseres. El formato de búsqueda y representación es muy similar al del listado de órdenes. Como en el caso anterior se puede buscar mediante una palabra clave, el código de barras o realizar una búsqueda vacía que devuelve todo el listado de mobiliario de la empresa.

Aquí veremos un ejemplo donde se puede ver el resultado de una búsqueda del listado de órdenes de trabajo.

Codigo	Fecha	Solicitante	Tipo	Ver	Informe
2016/3	28-06-2016 12:12:11	MANOLO PEREZ ESTEBAN	Entrega		-
2016/2	27-06-2016 20:51:04	MANOLO PEREZ ESTEBAN	Recogida		-
2016/1	27-06-2016 20:16:11	MANOLO PEREZ ESTEBAN	Recogida		-
2015/42	28-09-2015 21:13:56	Julian Garcia	Libre		-
2015/41	28-09-2015 16:11:16	MANOLO PEREZ ESTEBAN	Recogida		-
2015/39	22-09-2015 17:23:15	MANOLO PEREZ ESTEBAN	Entrega		-
2015/38	22-09-2015 17:21:56	MANOLO PEREZ ESTEBAN	Recogida		-
2015/37	22-09-2015 09:53:00	MANOLO PEREZ ESTEBAN	Recogida		-
2015/36	22-09-2015 09:42:52	MANOLO PEREZ ESTEBAN	Recogida		-

Figura 43. Listado órdenes.

6.4.2.11 Órdenes libres.

Las órdenes libres son un tipo de peticiones que nada tienen que ver con la recogida o la retirada de archivos. Este tipo de operaciones se tienen que tramitar de este modo y son casos especiales como por ejemplo todo lo relacionado con el mobiliario o las órdenes de destrucción. Para que una orden de destrucción sea válida ha de adjuntarse un documento escaneado con la firma del que hace la petición. De este modo se tiene una declaración formal de la intención del trabajador y se evitan posibles problemas.

Como veremos en la imagen la orden consta de una serie de campos ya pre-introducidos y de un text-box en el que se redactará la información correspondiente. Una vez se finaliza la orden esta se envía en formato de correo electrónico al gerente de TRANSOFI S.L y se actualiza la base de datos.

Figura 44. Órdenes libres.

La parte referente a los usuarios de la empresa cliente ha sido finalizada ya, solo resta la parte del diseño del administrador de la empresa ofertante del servicio.

6.4.2.12 Menú de opciones del usuario administrador de TRANSOFI S.L

Las obligaciones que tiene el administrador de TRANSOFI S.L son las de gestionar y crear tanto las empresas como los usuarios administradores. Como ya se indicó con anterioridad la empresa que quiera hacer uso del servicio web debe comunicarse con TRANSOFI S.L. Cuando ello sucede, el gerente de TRANSOFI S.L hace el registro de la empresa y genera el usuario administrador de la empresa.

Adicionalmente puede consultar el tracking que han seguido los archivos así como todas las operaciones relativas al almacenaje de archivos en cajas. Las ordenes de trabajo, las órdenes de destrucción y las altas de los enseres también son tarea a revisión por parte de este rol de usuario.

El resto de opciones también están a disposición de este usuario pudiendo cambiar su vista de la página web a como la vería el administrador de la empresa cliente.

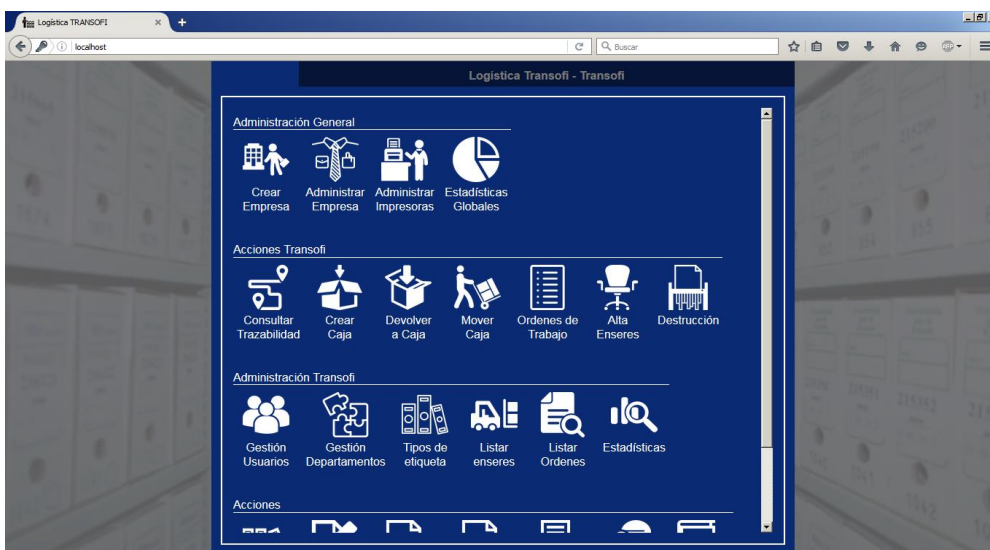


Figura 45. Menú de usuario del administrador de TRANSOFI S.L.

6.4.2.13 Creación de una empresa nueva.

Para el registro de una nueva empresa el administrador solo necesitará introducir el nombre de la empresa y un resumen del nombre. La base de datos se actualizará con la nueva información introducida junto con la creación automática del código de registro de la empresa con el que se podrá registrar al primer usuario. Dicho usuario será el usuario administrador de la empresa cliente. Este tendrá acceso a dicho código por lo que puede registrar a los usuarios manualmente o distribuir el código facilitado.

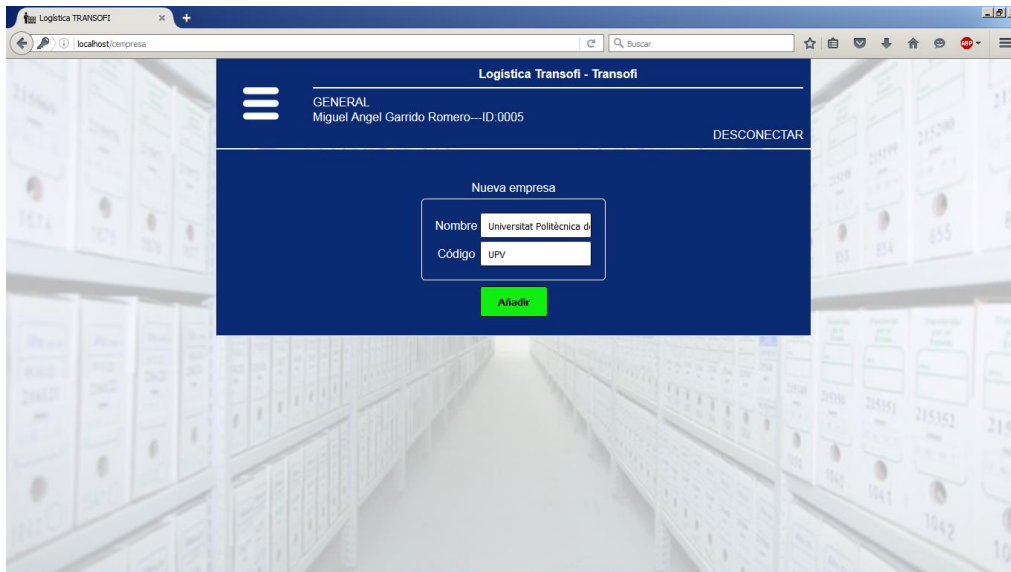


Figura 46. Creación de una nueva empresa.

6.4.2.14 Administración de la empresa cliente.

La administración de la empresa cliente surge como ayuda para solventar problemas que pudieran tener los clientes. Una vez se accede a ese recurso tenemos la opción de administrar con las mismas opciones y con la misma autoridad que el usuario administrador de la empresa cliente.

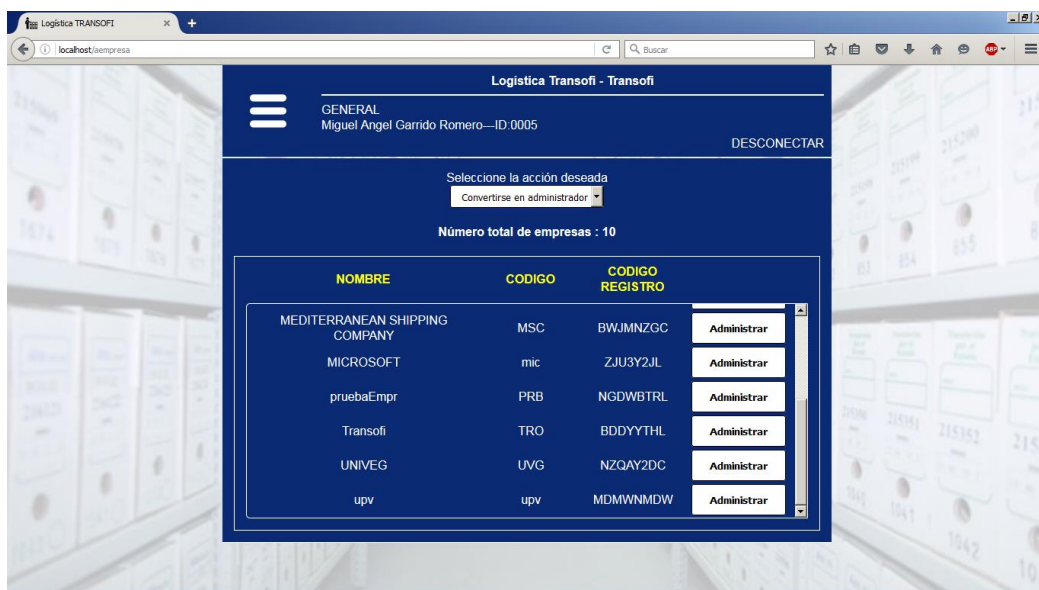


Figura 47. Administrar las empresas clientes.

Se genera un gran cambio en la interfaz puesto que los iconos pertenecientes al usuario administrador de TRANSOFI S.L permanecen igual sin embargo los recursos relacionados con la administración de la empresa cliente se ven en una tonalidad roja tal y como se ilustra en esta imagen.

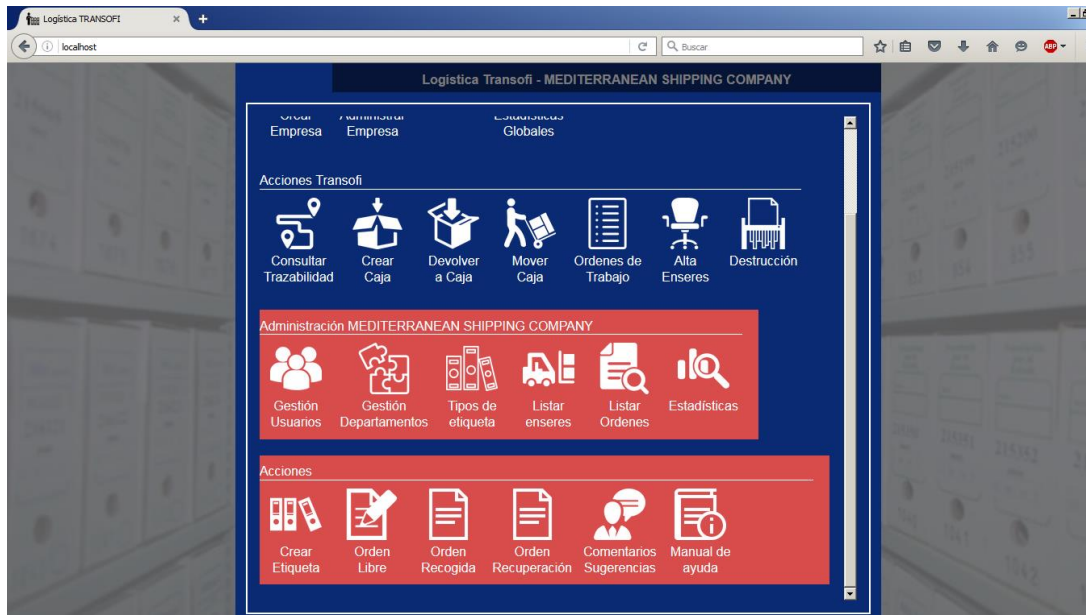


Figura 48. Panel de opciones durante la administración de empresas.

Las opciones de las que disponemos nos permiten hacer y deshacer tal y como haría el propio usuario administrador de la empresa cliente. De este modo si cualquier problemática surgiera el gerente de TRANSOFI S.L podría revisar en primera instancia la problemática y ofrecer una ayuda. Esto es también de muy útil para los usuarios noveles que aún no saben cómo hacer funcionar a la aplicación correctamente.

También se pueden editar los datos referentes a la empresa cliente pudiendo cambiar los datos relacionados con: nombre, la abreviación del nombre y el código de registro.

6.4.2.15 Trazabilidad

La trazabilidad de un archivo nos sirve para ver que trayectos ha ido siguiendo los archivos y cajas a lo largo de una cronología. En ocasiones surgen problemas con la localización de archivos así que se hace necesario tener un histórico de las rutas que ha ido siguiendo el archivo así como la localización actual. Los datos de la localización se van almacenando en la base de datos a través de la página web y la aplicación diseñada para la pda. Una búsqueda de la traza que han seguido los archivos podría resultar en algo como:

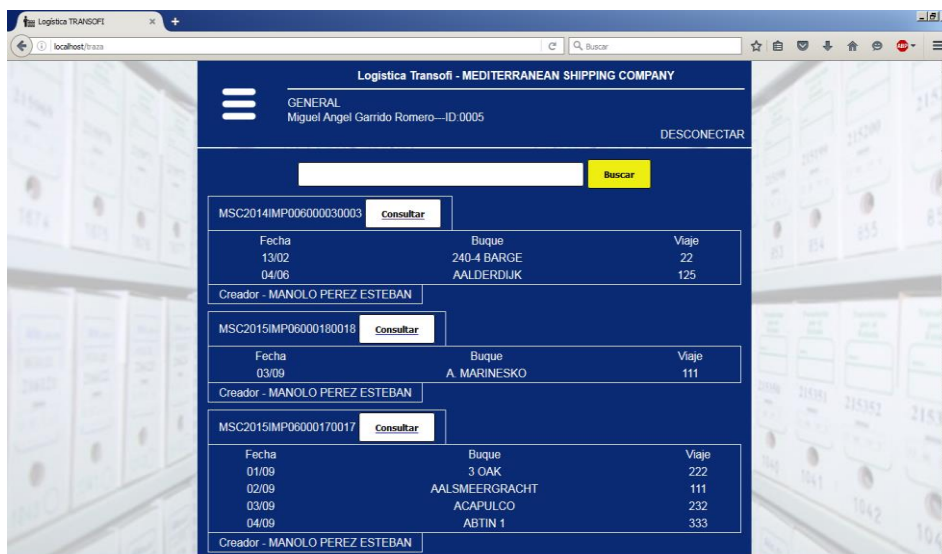


Figura 49. Trazabilidad.

6.4.2.16 Generar caja de almacenaje de archivos y devolver archivo a la caja.

Tal y como sabemos las cajas son el contenedor en el que los archivos son almacenados. Los archivos llegan a las instalaciones de TRANSOFI S.L donde son almacenados en dichas cajas. Antes de llevar la caja al almacén se le pega el código de barras que le corresponde y se precinta. La generación de la caja en la base de datos junto con el código de barras se hace mediante la opción del menú crear caja y en el empeño de generar una caja podríamos tener una visión de este estilo:

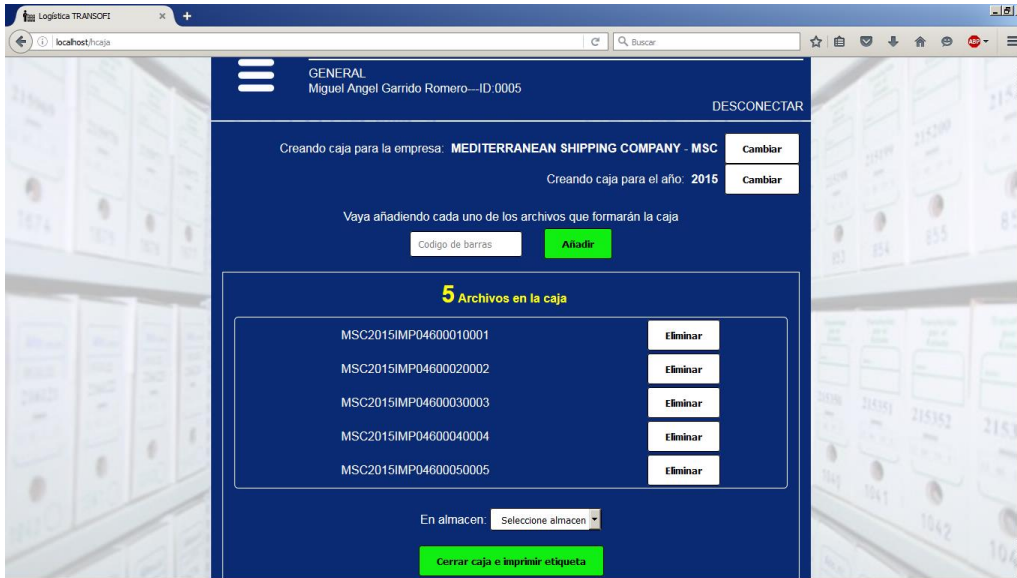


Figura 50. Crear caja.

En cuanto a los archivos de las cajas hay que destacar el caso en que un archivo es pedido por la empresa cliente. El archivo es solicitado a través del módulo que se ocupa de la recuperación de archivos. La información se actualiza en la base de datos y se manda en formato de correo electrónico un aviso al gerente de TRANSOFI S.L para que se atienda a la mayor brevedad. Los archivos son buscados mediante la aplicación web y una vez han sido localizados se procede a la apertura de la caja. Dicha apertura es permitida por la propia empresa que está solicitando dicho archivo. Finalmente el archivo se lleva a las instalaciones de la empresa cliente tal y como se especificó. A efectos de la base de datos el gerente de TRANSOFI S.L debe hacer los cambios pertinentes en la base de datos con la página web. Cuando el archivo vuelve desde la empresa cliente este debe volver a su caja original y por ello se generó el módulo que atendía a esta necesidad.

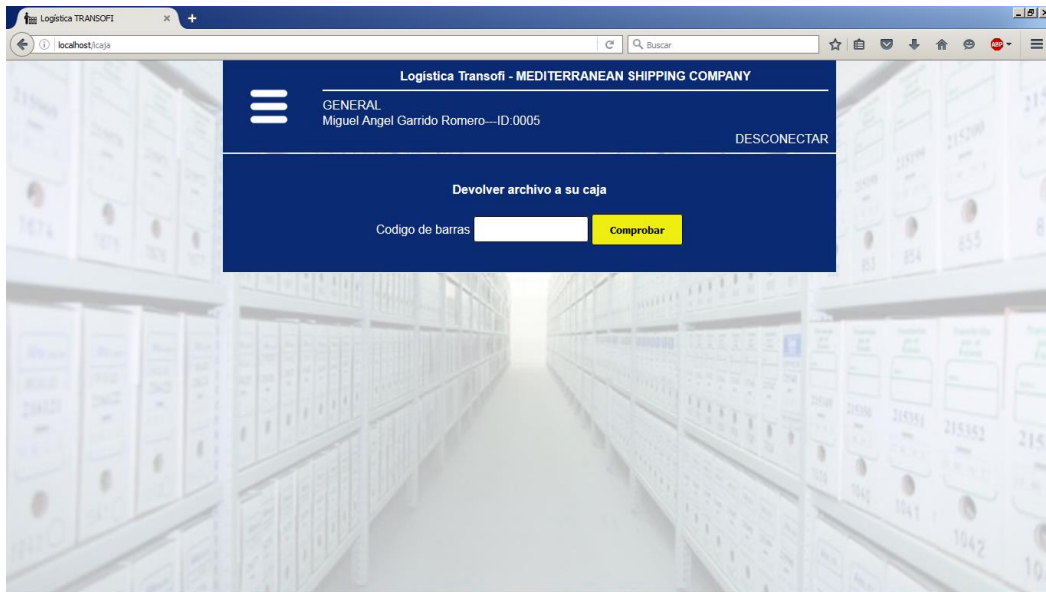


Figura 51. Devolver archivo a su caja.

Cuando el archivo vuelve a TRANSOFI S.L se debe actualizar su localización y su vuelta a la caja. Mediante la introducción del código de barras del archivo se comprueba si el archivo está efectivamente fuera de la caja y en caso de estarlo se procede a actualizar su estado a *en caja*.

6.4.2.17 Mover caja

El proceso de movimiento de cajas corresponde al momento en que estas deben ser guardadas en el almacén. Para realizar esta operación solo debemos introducir el código de barras que corresponde a la base de datos, el almacén al que irá la caja y a su localización dentro del almacén.

El formulario que nos permite hacer el movimiento de cajas es el siguiente:

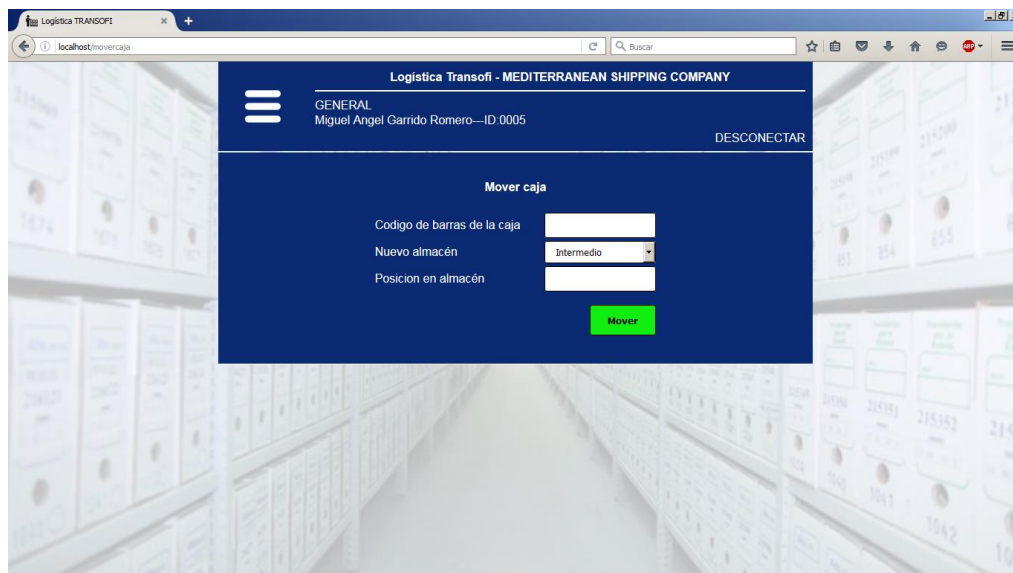
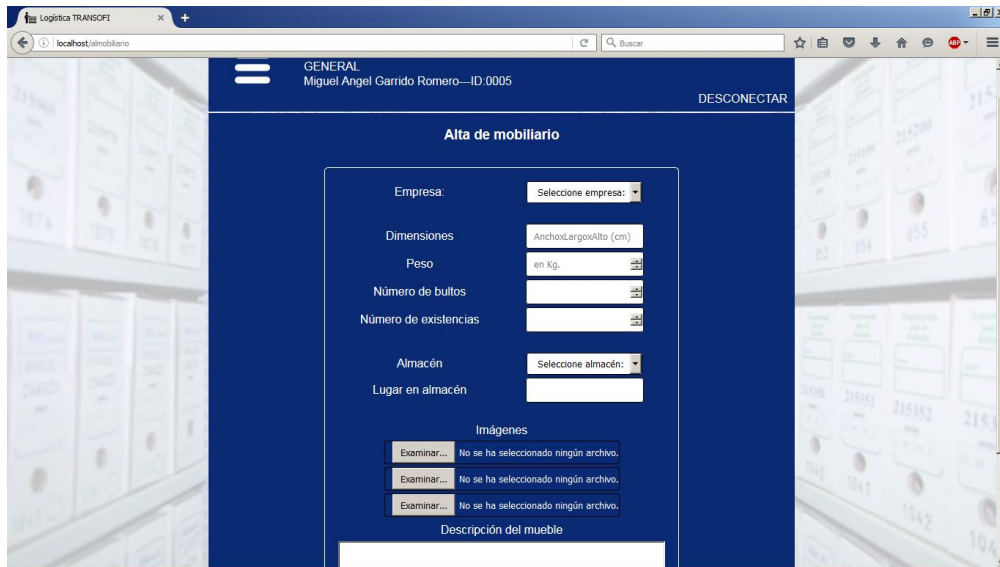


Figura 52. Mover caja.

6.4.2.18 Alta de mobiliario.

Como se comentó en otras secciones la opción de almacenaje de material de oficina es una de los servicios que están disponibles para los clientes. Así pues cuando se hace la orden libre que hace referencia al mobiliario los trabajadores de TRANSOFI S.L hacen la recogida para luego llevarlo a las oficinas. En dichas oficinas se dispone de un almacén donde se desmonta, en caso de poder hacerse, y guardan las piezas en contenedores adecuados. Por otro lado el

administrador de TRANSOFI S.L es el encargado de registrar en la base de datos el mobiliario especificando datos como: empresa, dimensiones, peso, etc. Adicionalmente se pueden adjuntar fotos del mueble para un reconocimiento más fácil.

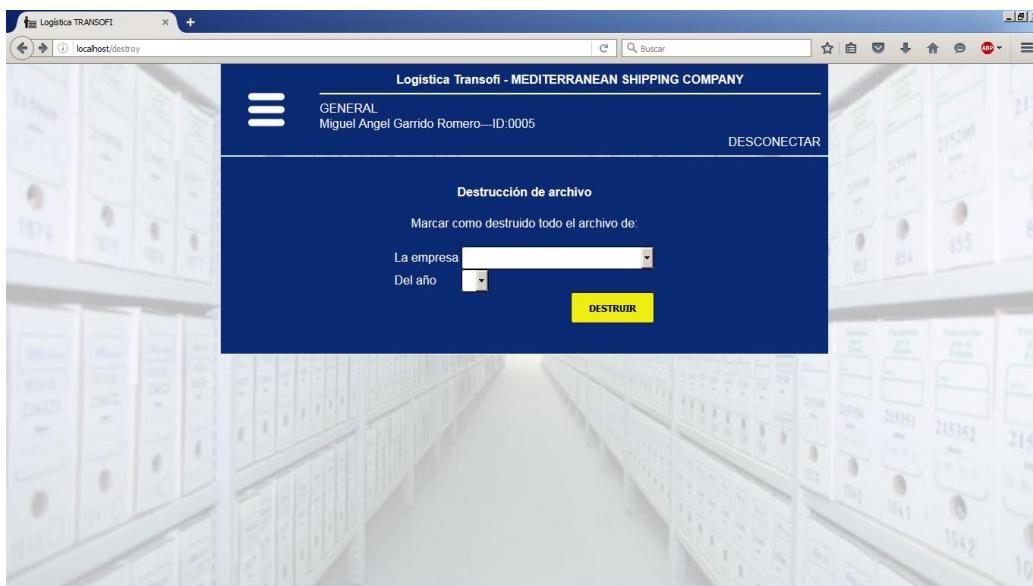


The screenshot shows a web browser window with the URL 'localhost/mobiliario'. The page title is 'Logística TRANSOFI' and the user is logged in as 'GENERAL Miguel Angel Garrido Romero—ID:0005'. The main heading is 'Alta de mobiliario'. The form includes the following fields: 'Empresa' (dropdown menu), 'Dimensiones' (AnchoxLargoxAlto (cm) input), 'Peso' (en Kg. input), 'Número de bultos' (input), 'Número de existencias' (input), 'Almacén' (dropdown menu), and 'Lugar en almacén' (input). There are three 'Imágenes' sections, each with an 'Examinar...' button and the message 'No se ha seleccionado ningún archivo.'. A 'Descripción del mueble' input field is at the bottom. A 'DESCONECTAR' link is in the top right corner.

Figura 53. Alta mobiliario.

6.4.2.19 Destrucción.

Las órdenes de destrucción, las cuales provienen de órdenes libres, son atendidas por el gerente de TRANSOFI S.L. Para reflejar esto se debe seleccionar la empresa y el año que se quiere destruir y con ello nos aparecerá un listado de cajas de la empresa cliente. Puesto que no todos las cajas del mismo año deben destruirse, por ejemplo el departamento jurídico suele querer destruir sus archivos mucho más tarde, tenemos una serie check-box con los cuales seleccionamos que cajas van a ser destruidas. Una vez seleccionados los archivos y dada la orden de destrucción el estado de esos archivos en la base de datos cambia a destruido. Posteriormente se llevan los archivos a la zona de destrucción para ejecutar la operación de forma real.



The screenshot shows a web browser window with the URL 'localhost/destroy'. The page title is 'Logística Transofi - MEDITERRANEAN SHIPPING COMPANY' and the user is logged in as 'GENERAL Miguel Angel Garrido Romero—ID:0005'. The main heading is 'Destrucción de archivo'. The form includes the text 'Marcar como destruido todo el archivo de:', 'La empresa' (dropdown menu), and 'Del año' (dropdown menu). A yellow 'DESTRUIR' button is at the bottom. A 'DESCONECTAR' link is in the top right corner.

Figura 54. Destrucción.

Una vez hemos visto todo lo relacionado con el diseño de la base de datos y web veremos cómo ha sido la configuración del servidor apache.

6.4.3 Configuración del servidor apache.

Para realizar esta tarea fue necesario modificar el archivo llamado htaccess que está localizado en el directorio public.

Mediante este archivo vamos a configurar los directorios web a los que se podrá acceder así como los puertos por los que se permitirá la conexión al servidor. Asegurando de este modo que no se pueda acceder a zonas no destinadas para los usuarios. Para asegurar las conexiones mediante el protocolo https se va a forzar que cualquier conexión que tenga el puerto 80 se re-direccione al puerto 443, que es el puerto destinado para el protocolo https.

Está fue la configuración que se dispuso:

Las líneas precedidas de el carácter # son comentarios

```
ssRewriteEngine on
```

```
RewriteBase /
```

#Con las siguientes instrucciones se declara de manera explícita a que directorios se puede # acceder desde el servicio web cualquier intento de acceso a otro directorio será #denegado.

```
RewriteRule ^(js) - [L]
```

```
RewriteRule ^(test) - [L]
```

```
RewriteRule ^(pda) - [L]
```

```
RewriteRule ^(gui) - [L]
```

```
RewriteRule ^(css) - [L]
```

```
RewriteRule ^(f9msBhHM.html)$ - [L]
```

#Toda conexión web que provenga desde el Puerto 80, http, será re-direccionada a #la dirección web con https.

```
RewriteCond %{SERVER_PORT} 80
```

```
RewriteRule ^(index.php)$ https://logistica.transofi.com [L]
```

#Si la conexión es mediante https se permite la carga del archive index.php y por ende #el acceso web.

```
RewriteCond %{SERVER_PORT} 443
```

```
RewriteRule ^(index.php)$ - [L]
```

#El resto de instrucciones están dirigidas al acceso cuando se le pasa un parámetro por el #método get, si se intenta realizar esta acción mediante http se re-direcciona a https. #Finalmente se permite hacer este tipo de acciones mediante https.

```
RewriteCond %{SERVER_PORT} 80
```

```
RewriteRule ^(.+)$ https://logistica.transofi.com/$1 [R,QSA,L]
```

```
RewriteCond %{SERVER_PORT} 443
```

```
RewriteRule ^(.+)$ index.php?params=$1 [QSA,L]
```

Cada vez que se accede a la página web o se intenta acceder a algún recurso de la misma se ejecuta esta directiva sobre la petición. Se ejecuta de manera secuencial y cuando no se cumple alguno de los parámetros se da un mensaje de error y el resto del archivo deja de ejecutarse.

Cuando la petición coincide con alguna de estas casuísticas se ejecuta la acción prevista y se procede a servir el recurso, el resto de archivo de configuración no se ejecuta puesto que ya hubo una coincidencia.

6.4.4 Migración a un servidor web comercial.

Tras el testeo de la aplicación se produjo el momento en que la aplicación estaba preparada para pasar a un dominio público y dejar de estar alojada en nuestro servidor local. Muchas eran las opciones que se barajaban: comprar un servidor web, alquilar un servidor web dedicado y alquilar un servidor web compartido.

Debido a que el hecho de la compra de un servidor web iba a suponer una gran inversión no solo en material sino en manutención del mismo se prefirió alquilar un servidor. Pero no todo fue debido a la inversión inicial sino también a las medidas de seguridad que había que implementar regularmente para prevenir posibles ataques o la configuración de copias de seguridad de todo el contenido del servidor, etc. La propia empresa deseaba una vez acabado el diseño no necesitar de un desarrollador que mantuviera el sitio y dejar este aspecto a esporádicas contrataciones a través de empresas secundarias de manera anual. Por todo ello la mejor opción era alquilar un servidor web y dejar todo preparado para que funcionara.

Entre las posibles opciones que tenemos para alquilar servidores tenemos la problemática de decidir entre un servidor compartido o dedicado.

Los servidores compartidos son un espacio grande donde se convive con otros usuarios. Esto significa que una cantidad concreta de usuarios puede hacer uso de este servidor. Todos los recursos del servidor son compartidos entre los usuarios: RAM, espacio en disco, CPU, transferencia, ancho de banda, IP, etc.

Como contras a esta solución tenemos que el ancho de banda es compartido. Esto significa que si alguna de las webs alojadas en el servidor acapara la transferencia de datos el resto de páginas web se ralentizarán. No hay que olvidar que el espacio en disco también es compartido por lo que nuestro espacio disponible estará reducido a un número fijo concreto que será mucho menor al de un servidor dedicado.

En contrapartida tenemos los servidores dedicados. En este tipo de servidores tenemos la gran ventaja de no tener que compartir ni uno solo de los recursos de los que se dispone. Ahora bien, este tipo de servicios es mucho más caro que los servidores compartidos y requiere de conocimientos de configuración de servidores.

La solución estaba clara, la elección de un servidor compartido en nuestro caso era la mejor opción por los siguientes motivos:

- Su coste es económico.
- Nuestro servicio no tiene tanto tráfico como para requerir de muchos recursos.
- Nadie será contratado para la gestión del servidor y las empresas de servidores compartidos cubren esta función mientras que las de los dedicados no.
- La empresa de hosting compartido se encarga de garantizar la seguridad del servidor así como de la realización de copias de seguridad.
- El hosting compartido es fácilmente escalable, si en un futuro el usuario necesitara ampliar la capacidad puede contratar de una manera muy sencilla un plan de servicios superior.

6.4.5 Contratación de un servidor web compartido y migración de la página.

Entre la multitud de opciones que se pudieron elegir para contratar el servicio se eligió OVH por anteriores experiencias con esta página por parte del gerente de TRANSOFI S.L. Así pues se procedió a la contratación de un servicio de hosting compartido. Una vez realizado esto se procedió a la instalación de todos los servicios que no estuvieran instalados ya en el servidor tal y como se hizo con nuestro servidor local. Puesto que se trataba también de un servidor linux todo fue instalado del mismo modo así que una vez todo estuvo dispuesto se migró el contenido del directorio web al servidor. Una vez hecho esto se accedió al phpmyadmin de nuestro nuevo servidor alquilado para migrar la base de datos.

Todos los preparativos estaban ya listos solo restaba dar unos últimos pasos de configuración como por ejemplo el nombre de nuestro dominio y algunas sencillas configuraciones más para que nuestra página web estuviera lista para el acceso de nuestros clientes.

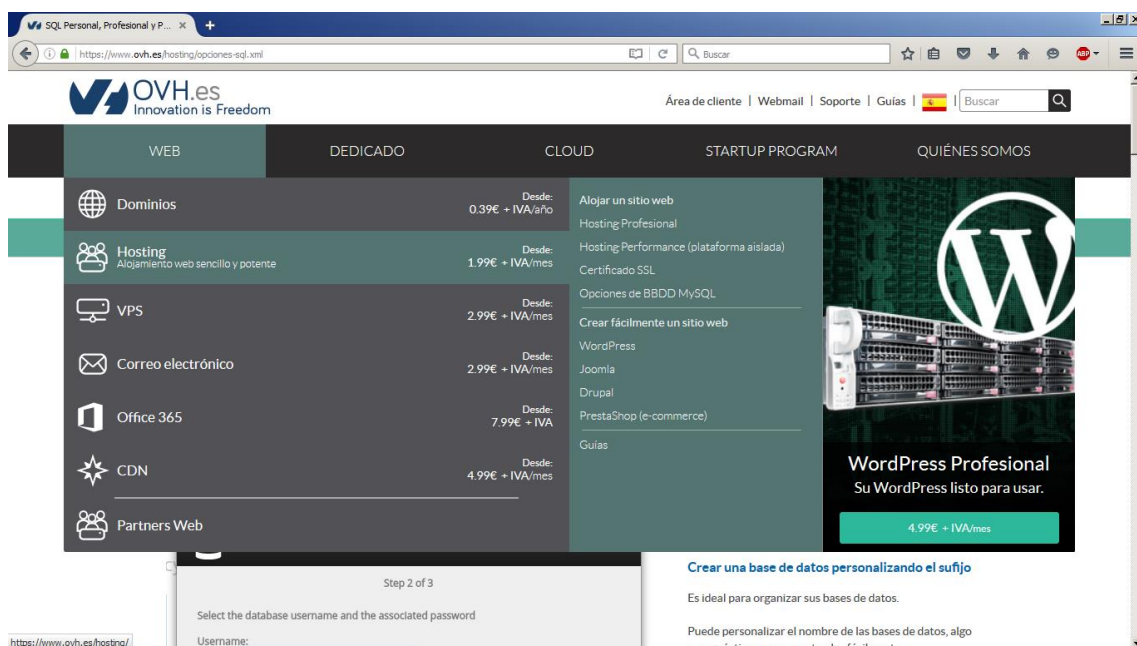


Figura 55. Contratación del servidor web. (Extraído de ovh.es)

Capítulo 7. Conclusiones y líneas futuras.

El desarrollo de una aplicación web requiere de muchas herramientas a usar por lo que la cantidad de material que hubo que aprender no fue poco. Esto fue muy enriquecedor en muchos aspectos puesto que al no tener ningún desarrollador más hasta casi finales de terminar la aplicación hizo que la experiencia resultara en un auto-aprendizaje continuo en temas como: el desarrollo web, programación en distintos lenguajes, configuración de servidores, instalación de infraestructuras de servidores, desarrollo e implementación de bases de datos, procesos de desarrollo del software y planificación temporal.

En un futuro próximo la habilidad para poder desarrollar una tarea desconocida y poder aprender cómo solucionar cualquier problema es vital. Así pues como primera toma de contacto con este próximo mundo laboral la experiencia ha sido muy enriquecedora. Ciertamente este tipo de tareas no son tarea de un solo desarrollador, la forma habitual de realizar este tipo de diseños es mediante desarrolladores especializados en aspectos concretos del diseño. Esto derivó en una gran carga en cuanto a la gran cantidad de distintas tareas a realizar pero sin duda se obtuvo una gran visión de conjunto.

Un aspecto que nada tuvo que ver con el desarrollo fue tratar con los clientes. La aplicación al fin y al cabo va destinada para ellos y no hay que olvidar que una fase importante es reunirse con los clientes y conseguir ver que es lo que desean. Varias reuniones fueron organizadas entre las cuales me gustaría destacar la reunión en la que se hizo una prueba demostrando el funcionamiento de la página web y explicando cada uno de los componentes y su funcionamiento. Los clientes tras la disertación expuesta, propusieron mejoras en el interfaz gráfico y algunas expansiones en la página que aún no contemplaba ciertas casuísticas especiales.

Este aspecto también fue muy interesante puesto que en el desempeño del futuro laboral como graduado en ingeniería en telecomunicación no solo es necesario tener una formación técnica sino que las habilidades comunicativas son también muy importantes.

No hay que olvidar que no solo la experiencia ha sido buena por parte del estudiante sino que la propia empresa contratante recibió un producto hecho a medida con un bajo coste. Sin duda las empresas que empiezan a ofertar prácticas obtienen un muy buen resultado y una grata experiencia.

Muchos fueron los aspectos que se trataron durante el desarrollo sin embargo no todos se pudieron llevar a cabo. El tiempo que se requería era finito y se propuso una serie de materias en las que mejorar o crear para posibles nuevas prácticas en empresa a modo de líneas futuras.

Veamos que posibles nuevas vías o mejoras se plantean para el futuro de la página:

- Establecer un formato de etiquetaje definido mediante el uso de un tipo de impresora concreta con un tipo de papel determinado. La impresora de etiquetas se provee por parte de Transofi S.L a las empresas y es algo que el gerente ya contemplaba.
- Actualizaciones en el código realizado. Algunas funciones de las que se hace uso en un futuro estarán obsoletas cosa que significará que habrá mejores y más seguras maneras de realizarlo.
- Implantar un sistema de firmas digitales para que los documentos como las órdenes de trabajo pudieran ser certificados por parte de las entidades emisoras otorgando una mayor seguridad que una simple firma en papel.
- Optimizar la posición en la que aparece la página en los motores de búsquedas, SEO.

Bibliografía

Texto

[1] Debian, razones para escoger debian, consultado en junio de 2016
https://www.debian.org/intro/why_debian.es.html

[2] Debian handbook, The Debian Administrator's Handbook, Installing Step by Step, consultado en junio de 2016.

<https://debian-handbook.info/browse/stable/sect.installation-steps.html>

[3] Servidor ssh, consultado en junio de 2016..

https://servidordebian.org/es/squeeze/config/remote_access/ssh_server

[4] Servidor http Apache, consultado en junio de 2016.

https://es.wikipedia.org/wiki/Servidor_HTTP_Apache

[5] php, consultado en junio de 2016.

<https://es.wikipedia.org/wiki/PHP>

[6] mysql, consultado en junio de 2016.

<https://es.wikipedia.org/wiki/MySQL>

[7] how to install linux apache mysql php (lamp) stack on Debian, consultado en junio de 2016.

<https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-debian>

[8] FileZilla, consultado en junio de 2016.

<https://es.wikipedia.org/wiki/FileZilla>

[9] Putty, consultado en junio de 2016.

<https://es.wikipedia.org/wiki/PuTTY>

[10] [11], SublimeText, consultado en junio de 2016.

https://es.wikipedia.org/wiki/Sublime_Text

[12] MySQL Workbench , consultado en junio de 2016.

https://es.wikipedia.org/wiki/MySQL_Workbench

Imágenes

Figura 1 shutterstock (consultado en junio de 2016) Disponible en:

http://www.shutterstock.com/pic.mhtml?utm_medium=Affiliate&utm_campaign=Eezy%20Inc&utm_source=38919&id=262678493&irgwc=1&tpl=38919-111120

Figura 3 w3techs (consultado en junio de 2016) Disponible en:

https://w3techs.com/blog/entry/debian_ubuntu_extend_the_dominance_in_the_linux_web_server_market_at_the_expense_of_red_hat_centos

Figura 4 mysecureshell (consultado en junio de 2016) Disponible en:

http://mysecureshell.readthedocs.io/en/latest/_images/logo_debian.png

Figura 5 y 6: debian (consultado en junio de 2016) Disponible en:

www.debian.org

Figura 9 wikipedia (consultado en junio de 2016) Disponible en:

https://en.wikipedia.org/wiki/LAMP_%28software_bundle%29#/media/File:LAMPP_Architecture.png

Figura 10 commons.wikimedia (consultado en junio de 2016) Disponible en:

[https://commons.wikimedia.org/wiki/File:Apache_Software_Foundation_Logo_\(2016\).svg](https://commons.wikimedia.org/wiki/File:Apache_Software_Foundation_Logo_(2016).svg)

Figura 11 commons.wikimedia (consultado en junio de 2016) Disponible en:

<https://commons.wikimedia.org/wiki/File:PHP-logo.svg>

Figura 12 commons.wikimedia (consultado en junio de 2016) Disponible en:

<https://commons.wikimedia.org/wiki/File:Database-mysql.svg>

Figura 13 y 18 mysql (consultado en junio de 2016) Disponible en:

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

Figura 14 commons.wikimedia (consultado en junio de 2016) Disponible en:

<https://commons.wikimedia.org/wiki/File:Phpmyadmin.gif>

Figura 15 lignux (consultado en junio de 2016) Disponible en:

<https://lignux.com/wp-content/uploads/2014/01/filezilla-logo.jpg>

Figura 16 i.mezeta.com (consultado en junio de 2016) Disponible en:

http://i.emezeta.com/cache/img/1511_o.jpg

Figura 17 sublimetext (consultado en junio de 2016) Disponible en:

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

Figura 55. OVH (consultado en junio de 2016) Disponible en:

<https://www.ovh.es/>