



UNIVERSIDAD DE VALLADOLID

E.T.S.I. TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

GRADO EN INGENIERÍA DE TECNOLOGÍAS ESPECÍFICAS DE
TELECOMUNICACIÓN
MENCIÓN EN SISTEMAS DE TELECOMUNICACIÓN

**Ampliación y flexibilización de una
plataforma web de gestión de tutela
Jurídica.**

Autor:

D. Francisco Javier Moñivas Rodríguez

Tutora:

Dña. Miriam Antón Rodríguez

Valladolid, Mayo de 2017

TÍTULO: Ampliación y flexibilización de una plataforma web de gestión de tutela jurídica.

AUTOR: D. Francisco Javier Moñivas Rodríguez

TUTORA: Dña. Miriam Antón Rodríguez

DEPARTAMENTO: Teoría de la Señal y Comunicaciones e Ingeniería Telemática

TRIBUNAL

PRESIDENTA: Dña. Miriam Antón Rodríguez

VOCAL: D. David González Ortega

SECRETARIO: D. Mario Martínez Zarzuela

SUPLENTE: D. Francisco Javier Díaz Pernas

SUPLENTE: Dña. M^a Ángeles Pérez Juárez

FECHA: Junio 2017

CALIFICACIÓN:

Resumen

El principal objetivo de este trabajo de fin de grado consiste en otorgar una flexibilidad a la plataforma web de gestión de tutela jurídica de personas mayores utilizada por la fundación de tutela jurídica FUNDAMAY. Esta flexibilidad consiste en proporcionar las herramientas necesarias a la fundación para la gestión de las diferentes áreas de gestión de personas mayores.

Además, se procede a realizar una ampliación de la aplicación de gestión de tutelas jurídicas, añadiendo una nueva funcionalidad que consiste en la posibilidad de modificar el estilo de la aplicación.

Al término del desarrollo de la flexibilización de la aplicación de gestión de tutelas y añadida su nueva funcionalidad, la aplicación existente se transformará en una aplicación genérica que podría ser utilizada por más entidades.

Palabras clave: PHP, MySQL, Zebra Form, Gestión de Permisos, phpMyAdmin

Abstract

The main objective of this end of degree work consists of providing flexibility to web management platform for legal guardianship of elderly people used by the Foundation's legal guardianship FUNDAMAY. This flexibility is to provide the tools necessary to the Foundation for the management of different areas of management of elderly people.

In addition, is to be an extension of the application of management of legal guardianships, adding a new feature that consists in the possibility to modify the style of the application.

At the end of the development of the management of guardianship application flexibility and added new functionality, the existing application will be transformed into a generic application that could be used by more entities.

Key words: PHP, MySQL, Zebra Form, Gestión de Permisos, phpMyAdmin.

Agradecimientos

A mi padre y a mi madre, Francisco y Angelita, por apoyarme en todo momento, por subirme el ánimo en los momentos más difíciles y por nunca dejar de confiar en mí.

A todos mis amigos de teleco por esos grandes momentos vividos a lo largo de la carrera.

A mi tutora Miriam por confiar en mí y estar siempre ayudándome en la realización de este proyecto.

INDICE DE CONTENIDOS

CAPÍTULO 1: INTRODUCCIÓN15

1.1 INTRODUCCIÓN.....	15
1.2 OBJETIVOS.....	16

CAPÍTULO 2: METODOS DE FLEXIBILIZACIÓN DE LA WEB.....19

2.1 INTRODUCCIÓN.....	19
2.2 AMPLIAR PERMISOS AL DIRECTOR DE LA APLICACIÓN.....	20
2.3 USUARIO DE SOPORTE.....	20
2.4 WEB GESTORA DE PERMISOS WEB.....	21
2.5 CONCLUSIÓN.....	23

CAPÍTULO 3: TECNOLOGÍAS DE DESARROLLO.....25

3.1 INTRODUCCIÓN.....	25
3.2 LENGUAJES DE PROGRAMACIÓN (LADO CLIENTE).....	26
3.2.1 HTML.....	26
3.2.2 CSS.....	27
3.2.3 JAVASCRIPT.....	28
3.2.4 JQUERY.....	28
3.2.5 XML Y AJAX.....	28
3.3 LENGUAJES DE PROGRAMACIÓN (LADO SERVIDOR).....	30
3.3.1 .NET.....	30
3.3.2 JSP.....	31
3.3.3 PERL.....	32
3.2.4 PHP.....	33
3.2.5 XML Y AJAX.....	33
3.4 BASE DE DATOS (BBDD).....	36
3.4.1 BASES DE DATOS RELACIONALES.....	37
3.4.1.1 SQL EL LENGUAJE DE LAS CONSULTAS.....	38
3.4.1.2 MYSQL.....	39
3.4.1.3 ORACLE.....	41
3.4.1.4 MICROSOFT SQL SERVER.....	41
3.4.2 BASES DE DATOS ORIENTADA A OBJETOS (OO).....	41

3.5 CONCLUSIONES.....	41
3.5.1 ZEBRA_FORM.....	43
3.5.2 INPUT="COLOR" HTML5.....	43
CAPÍTULO 4: INICIO DEL PROYECTO.....	45
4.1 NECESIDAD DEL PROYECTO.....	45
4.2 DESCRIPCIÓN DE LA FLEXIBILIZACIÓN.....	46
4.2.1 GESTOR DE WEBS.....	46
4.3 AMPLIACIÓN DEL GESTOR DE TUTELAS.....	47
4.4 OBJETIVOS Y VENTAJAS DEL PROYECTO.....	47
CAPÍTULO 5: DESCRIPCIÓN Y ANÁLISIS TÉCNICO.....	49
5.1 NUEVA WEB DE GESTIÓN.....	49
5.1.1 DISEÑO DE LA WEB.....	49
5.1.1.1 CABECERA.....	50
5.1.1.2 MENÚ PRINCIPAL.....	51
5.1.1.3 SUBMENÚ.....	51
5.1.1.4 CUERPO.....	52
5.1.1.5 PIE DE PÁGINA.....	53
5.1.1.6 ESTÉTICA.....	53
5.1.2 USABILIDAD DE LA NUEVA WEB.....	54
5.1.3 ORGANIZACIÓN DE LOS ARCHIVOS.....	55
5.1.4 DISEÑO DE LA WEB.....	57
5.1.5 PERMISOS DE LA WEB.....	60
5.1.6 FORMULARIOS DE REGISTROS Y CONTROL.....	61
5.1.7 SEGURIDAD.....	66
5.1.8 CASOS DE USO.....	67
5.1.9 BASE DE DATOS.....	70
5.1.10 COMUNICACIÓN ENTRE WEBS.....	75
5.1.11 NUEVA WEB DE GESTIÓN DE TUTELAS.....	76
5.1.12 CONCLUSIÓN.....	76
5.2 AMPLIACIÓN DE LA WEB DE FUNDAMAY.....	80
5.2.1 HOJAS DE ESTILO CON EXTENSIÓN PHP.....	80
5.2.2 NUEVA FUNCIONALIDAD DE LA APLICACIÓN.....	82
5.2.3 DISEÑO DE LA WEB.....	83
5.2.3.1 DISEÑO POR MEDIO DE INPUT="COLOR".....	84
5.2.3.2 ESTILOS YA CREADOS CON ANTERIORIDAD.....	85
5.2.4 CASOS DE USO DE LA NUEVA FUNCIONALIDAD.....	88

5.2.5 BASES DE DATOS.....	89
5.2.6 CONCLUSIONES DE LA AMPLIACIÓN.....	92

CAPÍTULO 6: MANUAL DE USUARIO.....93

6.1 INTRODUCCIÓN.....	93
6.1.1 SECCIONES DE LA PLATAFORMA.....	93
6.2 NUEVA FUNCIONALIDAD EN ADMINISTRACIÓN.....	94
6.2.1 CONFIGURACIÓN WEB.....	95
6.2.1.1 NAVEGADORES QUE SOPORTAN INPUT="COLOR"..	95
6.2.1.2 NAVEGADORES QUE NO LO SOPORTAN.....	100
6.3 WEB DE GESTIÓN DE PERMISOS.....	105
6.3.1 CABECERA DE LA WEB.....	106
6.3.2 MENÚ PRINCIPAL Y SUBSECCIONES.....	106
6.3.3 INFORMACIÓN DEL USUARIO.....	106
6.3.4 SECCIÓN MANTENIMIENTO WEB.....	107

CAPÍTULO 7: PRESUPUESTO ECONÓMICO.....117

CAPÍTULO 8: CONCLUSIONES Y LÍNEAS FUTURAS.....119

§.1 CONCLUSIONES.....	119
§.2 LINEAS FUTURAS DEL PROYECTO.....	122

BIBLIOGRAFÍA.....125

ÍNDICE DE IMÁGENES

FIGURA 1. Modelo clásico de aplicación web vs modelo Ajax.....	30
FIGURA 2. Porcentajes de empleo de lenguajes lado servidor.....	34
FIGURA 3. Esquema cliente-servidor de una base de datos.....	37
FIGURA 4. Compatibilidad de input="color" en distintos navegadores web.....	44
FIGURA 5. Diseño de la web de gestión de permisos.....	50
FIGURA 6. Cabecera de la web gestora de permisos.....	50
FIGURA 7. Menú principal.....	51
FIGURA 8. Submenú.....	51
FIGURA 9. Menú y submenú del jefe de mantenimiento (I).....	51
FIGURA 10. Menú y submenú del jefe de mantenimiento (II).....	52
FIGURA 11. Cuerpo de la web con datos.....	53
FIGURA 12. Pie de página.....	53
FIGURA 13. Paleta de colores utilizada en la web de mantenimiento.....	54
FIGURA 14. Iconos para simplificar la web.....	56
FIGURA 15. Distribución de los archivos.....	58
FIGURA 16. Diagrama de flujo de carga de una página de la web.....	58
FIGURA 17. Página de login.....	59
FIGURA 18. Tabla de permisos de usuarios.....	61
FIGURA 19. Controles formularios Zebra_Form.....	62
FIGURA 20. Diagrama de flujo de los formularios.....	63

FIGURA 21. Fragmento de código. Crear nuevo usuario.....	64
FIGURA 22. Formulario de permisos.....	65
FIGURA 23. Ejemplo de formulario.....	65
FIGURA 24. Mensajes de ayuda en los formularios.....	66
FIGURA 25. Fragmento de código para proporcionar seguridad.....	67
FIGURA 26. Archivos del sistema.....	67
FIGURA 27. Diagramas de casos de uso sección acceso al sistema.....	68
FIGURA 28. Diagramas de casos de uso sección de mantenimiento de las webs.....	69
FIGURA 29. Diagramas de casos de uso de la sección mantenimiento de los usuarios de la web.....	70
FIGURA 30. Tabla de usuarios de la base de datos.....	71
FIGURA 31. Tabla websadministradas de la base de datos.....	74
FIGURA 32. Conexión entre las bases de datos.....	75
FIGURA 33. Web con todos los permisos.....	76
FIGURA 34. Web sin los permisos de primer nivel Voluntariado y Listado.....	76
FIGURA 35. Web sin todos los permisos de primer nivel.....	77
FIGURA 36. Web con todos los permisos de segundo nivel.....	77
FIGURA 37. Web sin los permisos médicos y seguimiento.....	77
FIGURA 38. Web con todos los permisos de tercer nivel o nivel del datos del paciente.....	78
FIGURA 39. Web sin los permisos de mostrar datos de traslado de residencias de un paciente.....	78
FIGURA 40. Cabecera necesaria en las hojas de estilo .php.....	80
FIGURA 41. Fragmento de código de una hoja de estilos .php.....	81
FIGURA 42. Distribución de los archivos.....	82
FIGURA 43. Nuevo apartado de configuración web.....	83

FIGURA 44. Configuración de la apariencia web mediante navegador google Chrome.....	84
FIGURA 45. Configuración apariencia web mediante navegador Safari.....	85
FIGURA 46. Estilo de colores claros.....	86
FIGURA 47. Estilo de colores oscuros.....	86
FIGURA 48. Estilo de colores blancos y negros.....	87
FIGURA 49. Estilo de colores amarillos y verdes.....	87
FIGURA 50. Actores del sistema.....	89
FIGURA 51. Tabla de estilos web.....	90
FIGURA 52. Web de tutela inicialmente.....	94
FIGURA 53. Menú del director.....	95
FIGURA 54. Submenú configurar web en google Chrome.....	96
FIGURA 55. Paleta de colores.....	97
FIGURA 56. Botón de cabecera.....	97
FIGURA 57. Botón menú principal.....	97
FIGURA 58. Botón del submenú.....	98
FIGURA 59. Botón pie de página.....	98
FIGURA 60. Botón parte superior derecha.....	98
FIGURA 61. Comparación de los distintos estilos.....	100
FIGURA 62. Opciones de estilos.....	101
FIGURA 63. Vistas de los estilos disponibles.....	104
FIGURA 64. Pantalla login.....	105
FIGURA 65. Cabecera de la plataforma.....	106
FIGURA 66. Menú principal.....	106
FIGURA 67. Menú de subsección.....	106
FIGURA 68. Información del usuario logueado.....	107

FIGURA 69. Subsección webs administradas.....	107
FIGURA 70. Subsección configuración web.....	107
FIGURA 71. Información detallada de los permisos de una web.....	108
FIGURA 72. Formulario de edición de los permisos.....	109
FIGURA 73. Misma web con distintos permisos.....	111
FIGURA 74. Sección de mantenimiento de usuarios web. Vista del jefe de mantenimiento.....	113
FIGURA 75. Formulario nuevo usuario.....	114
FIGURA 76. Configurar usuarios.....	115

CAPÍTULO 1

Introducción al proyecto

1.1 INTRODUCCIÓN

En la actual configuración sociodemográfica de los países más avanzados, las personas mayores han adquirido un papel muy importante, debido entre otros factores a la baja tasa de natalidad y una esperanza de vida creciente, de lo que resulta una sociedad en proceso de envejecimiento. La Comunidad de Castilla y León no ha permanecido ajena a este proceso, sino que se ha constituido en la Comunidad con mayor tasa de envejecimiento de España. Esta situación afecta a todas las esferas de la sociedad donde las relaciones entre las generaciones cambian y las personas mayores, la familia y la sociedad deberán adaptarse a estos cambios [1].

Con este envejecimiento de la sociedad, se espera que el número de estas personas mayores incapacitadas o con necesidad de ayuda para desempeñar labores provocando que aumente el número de entidades dedicadas a tareas de gestión de tutelas de personas mayores.

A parte de este factor mencionado, la tecnología ha evolucionado notablemente en los últimos años llegando a informatizar prácticamente todas las actividades cotidianas, desde el simple pago de una factura, hasta contar las calorías y kilómetros que caminamos a lo largo del día.

Con estos dos factores, las distintas entidades que llevan a cabo labores de tutela jurídica están en crecimiento, aunque no todas lleven a cabo las mismas funciones. Aquí encontramos la gran posibilidad de explotar un nuevo sector laboral al cual podemos facilitar las labores con las aplicaciones web.

Una de las entidades de tutela de persona mayores de mayor importancia de Castilla y León es FUNDAMAY.

FUNDAMAY, es una entidad sin ánimo de lucro que se encarga del ejercicio de la tutela de personas mayores en la Comunidad de Castilla y León. Pretende ser un recurso social que garantice la protección personal y patrimonial de las personas que por su deterioro cognitivo, enfermedad o situación de riesgo han perdido su capacidad de obra, buscando el mayor nivel de protección y bienestar de las personas que tutela [2].

Pero no es la única entidad que trabaja en este sector, sino que hay muchas más entidades que desempeñan labores similares a FUNDAMAY, aunque no se centren en las personas mayores sino que en otro tipo de personas que no se valgan por sí

mismas, como por ejemplo FECLEM, fundación tutelar de personas con discapacidad mental o FUTUDIS, fundación tutelada encargada de personas con discapacidad intelectual o del desarrollo.

Viendo la cantidad de fundaciones de gestión de tutelas y que cada una se centra en una serie de labores concretas, nosotros intentamos ampliar nuestro sector de trabajo y no solo centrarnos en facilitar la labor a FUNDAMAY, sino que deseamos ayudar a todo este tipo de entidades. Gracias a todo esto, vemos la posibilidad de desarrollar una aplicación web flexible en la que las diferentes sociedades de tutela puedan apoyarse para desempeñar los diferentes trabajos y tareas que llevan a cabo.

En conclusión, en el siguiente documento desarrollaremos una idea de flexibilizar una plataforma de gestión de tutelas de personas mayores. Esto consistirá en proporcionar a las diferentes entidades las secciones deseadas para llevar a cabo sus tareas de gestión de tutelas, pudiendo elegir entre diversas secciones concretamente de un área social, jurídica, de voluntariado y de SAOS (Servicio de Atención y Orientación Social).

Pero esto no termina aquí, sino que se proporcionara una ampliación de la plataforma web de la que partimos que consistirá en la capacidad de modificar la apariencia de la propia web.

1.2 OBJETIVOS

El objetivo general de este trabajo consiste en flexibilizar la herramienta de gestión de tutelas jurídicas con la finalidad de proporcionar solo funcionalidades concretas a las diferentes entidades de tutela según el campo en el que trabajen.

Inicialmente, solo trabajaremos con FUNDAMAY ya que partimos de su propio desarrollo y gracias a este desarrollo podremos ampliar el número de entidades a las que ofreceremos este servicio de gestión de tutelas.

Además, llevaremos a cabo labores de ampliación de la propia aplicación web de gestión que posee dicha entidad, en concreto, dando la posibilidad, si lo desea, al administrador de esta aplicación a modificar la apariencia de la web.

1.3 FASES DEL TRABAJO

El siguiente trabajo que vamos a desarrollar consta de diferentes fases a la hora de su elaboración [3]:

- Búsqueda de información sobre servicios de gestión de aplicaciones web. En este apartado conseguiremos la información necesaria para decidir la forma de proporcionar la flexibilidad a la aplicación de gestión de tutelas, optando por una de las posibilidades y centro su estudio y desarrollo en esta decisión.
- Análisis de las herramientas y tecnologías a utilizar a la hora de realizar el desarrollo necesario para proporcionar la flexibilidad necesaria. Sobre todo nos centraremos en herramientas de gestión de permisos y de gestión de bases de datos.
- Diseño de una plataforma de gestión de las distintas áreas que posee la entidad de tutela jurídica.
- Desarrollo de dicha plataforma utilizando las herramientas y las tecnologías elegidas anteriormente.
- Fase de pruebas, una fase muy importante, que lleve a cabo pruebas de seguridad para comprobar que funciona correctamente todos los campos y funcionalidades deseadas y desarrolladas en el apartado anterior.
- Una vez superadas estas pruebas, procederemos a una ampliación de la propia plataforma de gestión de tutelas de FUNDAMAY con la capacidad de modificación de la propia apariencia de la plataforma.
- Finalmente, se realizaran unas últimas pruebas y se elaborará una memoria del proyecto explicando paso a paso la funcionalidad tanto del gestor de las plataformas de tutela como de la nueva funcionalidad de la plataforma creando un manual de usuario.

Procedemos a explicar todos los puntos del trabajo de forma detallada.

CAPÍTULO 2

Métodos de flexibilización de la web

En el próximo capítulo se procede a explicar los diferentes métodos posibles para flexibilizar la aplicación existente de FUNDAMAY, con sus ventajas e inconvenientes llegando a una conclusión de cual elegiremos para desarrollar.

También se explicará con detalle la elección de que método que desarrollaremos a la hora de flexibilizar nuestra web con sus conclusiones de forma detallada.

2.1 INTRODUCCIÓN

En el proyecto, el cual consisten en flexibilizar una aplicación web, necesitábamos tomar una decisión muy importante antes de comenzar a desarrollarlo. Tras la búsqueda de información y de una serie de reuniones se llega a una serie de posibilidades mediante las cuales podremos flexibilizar la web. Estas posibilidades para proporcionar la flexibilidad son [3]:

-Al propio director de la web de Fundamay, otorgar los permisos de visión de las distintas áreas de tutela.

- Externalizar los permisos de la aplicación web de FUNDAMAY creando una aplicación gestora de permisos web.

- Crear un usuario, normalmente se le denomina usuario de soporte, el cual tendrá unos privilegios especiales para otorgar permisos de visión de los diferentes apartados de la aplicación web.

Este era el principal problema a la hora de flexibilizar la aplicación de FUNDAMAY por lo que se decide buscar información y tomar decisiones según las diferentes posibilidades anteriormente descritas.

A continuación procedemos a describir las dos posibilidades que tenemos a la hora de desarrollar el proyecto.

2.2 AMPLIAR PÉRMISOS AL DIRECTOR DE LA APLICACIÓN GESTORA DE TUTELAS

La primera posibilidad, y a priori, la más sencilla, para otorgar flexibilidad a la web es dar esta nueva funcionalidad al usuario existente de la aplicación, denominado “Director”.

Este usuario poseería la capacidad de hacer visible o no las diferentes áreas que tiene la web otorgando así la flexibilidad deseada al comienzo del proyecto. Pero esta posibilidad conlleva una serie de consecuencias negativas y positivas que veremos a continuación [3].

La gran ventaja que posee esta opción y a la vez es su mayor inconveniente es que toda la funcionalidad de permisos tanto de usuarios como de visión de las áreas cae en una única figura, la del “Director”. Por lo que la aplicación no queda flexibilizada en términos genéricos ya que un propio usuario de la aplicación tiene la capacidad de acceso a todas las áreas de la aplicación.

Una de las ventajas es el tiempo de desarrollo el cual es ínfimo, solo consiste en añadir una nueva funcionalidad a un usuario que ya existe.

Es un problema centrar la función de flexibilización en el “Director” de la plataforma ya que no se flexibiliza la plataforma sino que se está añadiendo una nueva funcionalidad al usuario y no realizando la tarea deseada al comienzo del proyecto.

Por lo tanto, sería una opción válida en caso de no disponer apenas tiempo de desarrollo o en caso que deseemos tener todos los permisos y funcionalidades entorno a una única figura, la del “Director”.

2.3 USUARIO DE SOPORTE

Como segunda posibilidad para otorgar flexibilidad a la web es la creación de un usuario especial denominado “Usuario de soporte”. Este usuario tiene la función principal y única de dar permisos a la aplicación de FUNDAMAY para que sean visibles o no las distintas áreas de tutela, dejando al usuario denominado “Administrador” como el único capaz de modificar los datos y permisos de los usuarios de la aplicación de FUNDAMAY [3].

Hay que explicar, que en la aplicación existente de FUNDAMAY, posee distintos tipos de usuarios, entre los cuales el “Director” posee la capacidad de ver todas las áreas de tutela y a la vez de dar permisos y modificar los datos del resto de usuarios de la aplicación pero no tiene la capacidad de dar esta flexibilidad a la hora de ver o no las distintas áreas de la web. Por lo tanto surge el nuevo usuario del que hemos hablado antes, el usuario de soporte.

Esta posibilidad de crear un nuevo usuario ofrecía una serie de ventajas e inconvenientes que vamos a explicar detalladamente.

Este usuario tendría la única finalidad de otorgar permisos de visión de las distintas áreas de tutela de pacientes de la propia aplicación de FUNDAMAY, según los criterios que necesitaran o contrataran el director de la fundación.

Una de las primeras ventajas de la creación de este usuario es que formaría parte de la misma base de datos de usuarios que posee esta aplicación, siendo un usuario más, que simplemente tendría unos permisos especiales de edición de permisos de visualización de las diferentes áreas de la web. La creación de este usuario es un proceso simple y sencillo y no tendría una gran carga de trabajo a la hora de modificar la aplicación web.

Con la nueva figura, se separa la tarea de flexibilizar la plataforma con la tarea de facilitar la gestión de las distintas áreas de tutela que se posee.

El usuario de soporte, como lo hemos denominado, aunque normalmente para distinguirlo bien del resto de usuarios de la web se lo denomina como “support”, estará integrado dentro de la misma aplicación gestora teniendo únicamente permisos para editar la visión de las áreas de la aplicación sin poder ver los datos de los diferentes pacientes que posee la fundación.

Otra desventaja de introducir este usuario en la misma BBDD que el resto de usuarios de la aplicación de FUNDAMAY, es en términos de seguridad. En el caso hipotético de robo de datos, pérdida de datos o caída del propio servidor de la web, no podríamos acceder como usuario de soporte para modificar estos permisos de la web lo cual quedaría como inservible la flexibilización. Por lo tanto es aconsejable, externalizar dicho usuario a otra BBDD diferente a donde se encuentran los usuarios de la aplicación web con el fin de proporcionar una cierta seguridad a la aplicación.

En conclusión, esta segunda opción para flexibilizar la aplicación de FUNDAMAY, tiene una serie de inconvenientes que hace que posea una menor seguridad y no otorgamos toda la flexibilidad que deseamos a la aplicación existente, ya que el usuario nuevo a crear sería un usuario más de la aplicación de tutela jurídica, un usuario especial.

2.4 WEB GESTORA DE PERMISOS WEB

La última posibilidad que pensamos para flexibilizar la aplicación web de FUNDAMAY, consiste en externalizar al usuario o usuarios que proporcionen los permisos en una aplicación externa y cuya única función sea dar soporte de visión de las diferentes áreas de la aplicación web [3].

Esta aplicación externa es denominada como “Gestor de permisos web”, donde al principio la única finalidad es dar una flexibilidad a la aplicación de FUNDAMAY.

La web gestora de permisos, consiste en una aplicación externa a la propia aplicación existente de FUNDAMAY a la que se desea dar un soporte y/o servicio de flexibilización, que posee de su propia funcionalidad, sus propios usuarios con distintos roles y de una propia BBDD. Es decir, separamos por completo la funcionalidad de la aplicación de tutela con la aplicación de otorgar permisos de visión de las diferentes áreas de la web de FUNDAMAY [3].

Pero con esta opción no solo estamos flexibilizando la aplicación de FUNDAMAY, sino que estamos abriéndonos a más mercados y fundaciones de tutela ya que podríamos gestionar los permisos de esta web de FUNDAMAY para otras instituciones que deseen utilizarla para hacer más ameno su trabajo. Es decir, estamos haciendo una aplicación genérica y a su vez estamos flexibilizando la aplicación con lo que podemos ofrecérsela a otras fundaciones que lo deseen y a través de nuestro gestor de webs podemos otorgar los permisos de las diferentes áreas de tutela que deseen contratar cada fundación.

Podemos ver lo anterior como una gran ventaja que posee esta opción pero necesitaremos desarrollar una nueva web gestora de permisos la cual lleva su tiempo de desarrollo y otro gran problema es hacer que la web existente desarrollada para FUNDAMAY pueda ser vendida y/o utilizada como una web genérica para otras fundaciones.

Dentro de las ventajas, cabe destacar, al poseer una separación completa entre los usuarios que dan soporte a la aplicación de tutela y los usuarios de la propia aplicación de tutela, los datos de cada fundación a la que ofrezcamos estos servicios, están ocultos a los usuarios de soporte por lo que seguirán siendo confidenciales.

Al poseer una BBDD propia para esta web de permisos, obtenemos una mayor seguridad ya que los datos donde se encuentran los permisos de visión de cada una de las áreas de la aplicación de gestión de tutelas se encuentran fuera de la propia BBDD de la aplicación sin tener acceso a su modificación.

En conclusión, es una gran idea la de externalizar por completo el usuario que se encarga de dar los permisos de visión de las diferentes áreas pero necesitaremos permisos de FUNDAMAY para poder utilizar la aplicación que se los ha desarrollado y transformarla en una aplicación genérica para poder ofrecérsela a otras fundaciones y así dar sentido a la creación de esta web gestora de permisos.

2.5 CONCLUSIÓN

Una vez visto las diferentes posibilidades para flexibilizar nuestra web, tenemos que tomar una decisión.

Esta decisión se toma en base a las siguientes conclusiones:

- Deseamos una aplicación 100% flexible.
- Otorgar una mayor seguridad a la web y los datos.
- No homogeneizar todas las funcionalidades en una única figura.

Por todo esto, se decide desarrollar una aplicación externa cuya finalidad es otorgar los permisos necesarios para visualizar o no las diferentes áreas de tutela que se poseen en la aplicación de FUNDAMAY.

Con esta decisión tendremos completamente separada la funcionalidad de la propia web de tutela jurídica de la funcionalidad de otorgar los permisos necesarios para visualizar o no las diferentes áreas de tutela.

La seguridad proporcionada al separar la BBDD que contiene los datos de los diferentes pacientes de tutela con otra BBDD que simplemente tiene los permisos que otorgan la visión de la áreas de la web, proporciona la mayor seguridad posible protegiendo los datos de los pacientes, que es lo más importante.

Además, si conseguimos los permisos de FUNDAMAY para que la aplicación que tienen actualmente, pueda ser una web genérica para otras fundaciones que realicen trabajos similares de tutela, podremos dar servicio a muchas más fundaciones.

Cabe destacar, que esta decisión a la hora de desarrollar nuestro proyecto, se toma por todos los motivos anteriormente mencionados, y también con la ayuda de la empresa Techrules S.A , que tiene una forma de trabajo similar otorgando esta serie de permisos similares pero centrado en otro área de negocio, concretamente en el área de la banca.

CAPÍTULO 3

Tecnologías de desarrollo

En el próximo capítulo se detallaran las tecnologías de desarrollo tanto en el lado del cliente como en el lado del servidor y las diferentes tecnologías que se utilizan para manejar las bases de datos.

También explicaremos un atributo muy interesante para proporcionar la ampliación de la web de FUNDAMAY, que consiste en modificar su apariencia de colores.

Para finalizar el capítulo, se tomará la decisión de elegir entre este conjunto de tecnologías para el desarrollo de nuestro proyecto.

3.1 INTRODUCCIÓN

A la hora de comenzar a desarrollar cualquier tipo de proyecto, es de vital importancia el estudio de las diferentes tecnologías y herramientas existentes que permitan el desarrollo del mismo, viendo los puntos fuertes y débiles de cada una y tomar una decisión que sirva para poder conseguir desarrollar o cumplir todos los objetivos prefijados del proyecto.

En todos proyectos hay que realizar este estudio previo, en nuestro caso el proyecto es un proyecto de desarrollo de software, por lo tanto necesitamos hacer un estudio de todas las tecnologías y herramientas que nos permitan desarrollar el proyecto cumpliendo todas las funcionalidades deseadas y otorgando un mínimo de calidad deseada por los usuarios finales del proyecto.

Por lo tanto, en un proyecto de software hay que realizar un estudio de las tecnologías a emplear en distintos pasos:

- Sistema operativo.
- Hardware para el servidor web
- Sistema gestor de la base de datos.
- Lenguaje o lenguajes de programación a utilizar.
- Software del servidor web.

Cuando realizamos el estudio de los apartados anteriores veremos que la elección de un tipo de tecnología u otra condiciona al resto de apartados. Como es el

caso de los sistemas operativos los cuales no funcionan en todo tipo de hardware o que todos los lenguajes de programación puedan conectar a todo tipo de base de datos.

Esta fase de estudio es de vital importancia para todo el proyecto, por lo que hay que tomar una decisión muy importante y la cual ha de ser la más idónea para cumplir los requisitos fijados en el proyecto.

Nosotros nos vamos a centrar el estudio en los lenguajes de programación tanto del lado de cliente como del servidor, en la propia base de datos necesaria para el desarrollo de nuestro proyecto y en todas las tecnologías que existen para facilitarnos el desarrollo de nuestro proyecto.

3.2 LENGUAJES DE PROGRAMACIÓN (LADO CLIENTE)

Comenzamos el análisis de las tecnologías que existen para la programación del lado del cliente. Un lenguaje de programación del lado de cliente son aquellos que se usan para su integración en páginas web. Las características de estos lenguajes son las siguientes [4]:

- Los programas residen en el servidor pero se ejecutan en el cliente.
- Se descarga de trabajo a los servidores.
- La ejecución del programa requiere una transmisión por la red del código necesario para ello.
- Las respuestas a las acciones de los usuarios sobre el programa pueden ser invocadas sin necesidad de realizar transmisiones por la red.
- Para la correcta ejecución del programa se requiere que el cliente tenga instalados programas o plug-ins adecuados.
- Si en un cliente no está instalado alguno de los programas intérpretes o plug-ins, la página no se ejecutará correctamente.
- Al transferirse el código, el cliente tiene acceso a dicho código y puede obtener a partir de él información que pueda resultar comprometida.
- Se pueden integrar los programas en las páginas alojadas en cualquier servidor web.

3.2.1 HTML

HTML, (Hyper Text Markup Language, lenguaje de marcas de Hipertexto), es el lenguaje de marcas de texto utilizado normalmente en la www (World Wide Web) [5].

Fue creado en 1986 por el físico nuclear Tim Berners-Lee, el cual tomo dos herramientas preexistentes:

- El concepto de Hipertexto (Conocido también como link o ancla) el cual permite conectar dos elementos entre sí.
- El SGML (Lenguaje Estándar de Marcación General) el cual sirve para colocar etiquetas o marcas en un texto que indique como debe verse.

HTML no es propiamente un lenguaje de programación como C++, Visual Basic, etc., sino un sistema de etiquetas. HTML no presenta ningún compilador, por lo tanto algún error de sintaxis que se presente éste no lo detectará y se visualizara en la forma como éste lo entienda.

El entorno para trabajar HTML es simplemente un procesador de texto, como el que ofrecen los sistemas operativos Windows (Bloc de notas), UNIX (el editor vi o ed), o Visual Studio. Estos documentos pueden ser mostrados por los visores o "browsers" de páginas Web en Internet, como Netscape Navigator, Mosaic, Opera y Microsoft Internet Explorer.

También existe el HTML Dinámico (DHTML), que es una mejora de Microsoft de la versión 4.0 de HTML que le permite crear efectos especiales como, por ejemplo, texto que vuela desde la página palabra por palabra o efectos de transición al estilo de anuncio publicitario giratorio entre página y página.

3.2.2 CSS

Hojas de Estilo en Cascada (Cascading Style Sheets), es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura [6].

Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos.

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento.

CSS funciona a base de reglas, es decir, declaraciones sobre el estilo de uno o más elementos. Las hojas de estilo están compuestas por una o más de esas reglas aplicadas a un documento HTML o XML. La regla tiene dos partes:

- Un selector que funciona como enlace entre el documento y el estilo.
- La declaración es la parte de la regla que establece cuál será el efecto.

CSS como hemos explicado anteriormente, permite al desarrollador generar un patrón de estilo que es aplicable a todos los documentos de una web y organizar el contenido de la misma.

La principal desventaja de las hojas de estilo es que no son compatibles con todos los navegadores, algo que ha de tener en cuenta el desarrollador a la hora de crear estas hojas de estilo para la aplicación.

3.2.3 JAVASCRIPT

JavaScript es un robusto lenguaje de programación que puede ser aplicado a un documento HTML y usado para crear interactividad dinámica en los sitios web. Fue inventado por Brendan Eich. Es un lenguaje interpretado basado en scripts que se introducen en el propio documento HTML [7].

Las características principales de este lenguaje de scripts son las siguientes:

- Es un lenguaje interpretado (no necesita compilación, es el navegador del cliente el encargado de traducir el código).
- Se ejecuta en el cliente, nunca en el servidor.
- No es (muy) extensible.
- Es un lenguaje de alto nivel y multiplataforma.
- Parte de un conjunto limitado de objetos y clases.
- No permite interacción cliente-servidor, únicamente opera en el cliente (Relativamente) integrado a HTML.

Las aplicaciones de JavaScript se diseñan para programas sencillos y pequeños. Suele utilizarse para realizar tareas repetitivas o para programar eventos de usuario. Incluso para la función de validación de datos y proporcionar efectos en las páginas web.

Por todo esto, JavaScript nos permite desarrollar aplicaciones que se ejecutan en el cliente directamente sin necesidad de conexión con ninguna base de datos.

3.2.4 JQUERY

jQuery es uno de los complementos más esenciales para el desarrollo web, usado en millones de sitios en toda la web, ya que nos facilita mucho el desarrollo de aplicaciones enriquecidas del lado del cliente, en JavaScript, compatibles con todos los navegadores [8].

jQuery no es un lenguaje, sino una serie de funciones y métodos de JavaScript. Por tanto, JavaScript es el lenguaje y jQuery es una librería que podemos usar opcionalmente si queremos facilitar nuestra vida cuando programamos en JavaScript.

A veces nos podemos referir a jQuery como framework o incluso como un API de funciones, útiles en la mayoría de proyectos web.

Con jQuery conseguimos simplificar la manera de interactuar con los documentos HTML ahorrándonos mucho tiempo a la hora de desarrollar el código.

3.2.5 XML Y AJAX

XML (eXtensible Markup Language ó Lenguaje extensible de marcas) es un conjunto de reglas que sirven para definir etiquetas semánticas para organizar un documento. Además el XML es un metalenguaje que te permite diseñar tu propio lenguaje de etiquetas. A diferencia del lenguaje de etiquetas normal (HTML), XML te permite definir tu propio lenguaje [9].

Viéndolo desde un punto de vista simple y sin meternos en definiciones muy técnicas, XML nos ayuda a tener nuestra información estructurada jerárquicamente por medio de etiquetas ó Tags que nosotros mismos crearemos.

Para crear un documento XML no se necesita nada más que un procesador de texto simple y guardar el documento con la extensión .xml.

AJAX es el acrónimo de *Asynchronous Javascript and XML*, es decir: JavaScript y XML Asíncrono. Este acrónimo fue utilizado por primera vez por Jesse James Garret en 2005, en su publicación *Ajax: a New Approach to Web Applications* si bien los componentes en que se basan y los recursos técnicos de que hace uso ya existían desde muchos años antes.

Normalmente, AJAX se define como una técnica para el desarrollo de páginas (sitios) web que implementan aplicaciones interactivas.

AJAX permite que una página web que ya ha sido cargada solicite nueva información al servidor. Dicho así, no supondría en realidad ningún invento novedoso. Una página web que contiene un enlace permite que se solicite al servidor nueva información cada vez que se pincha dicho enlace. Una página web que contiene un formulario envía información al servidor y recibe de él nueva información,

normalmente la respuesta ante los datos que se han enviado. En ambos casos hay una conexión entre el cliente y el servidor.

La diferencia es que con AJAX no es necesario recargar toda la página web, como ocurre cuando pinchamos en un enlace web o cuando pulsamos el botón submit de un formulario.

Con AJAX es posible realizar una conexión a un servidor desde dentro de una página web usando un programa JavaScript. Dicho servidor enviará una respuesta; esta respuesta se almacenará en una variable del programa JavaScript y, una vez almacenada en la variable, podremos hacer con ella lo que deseemos.

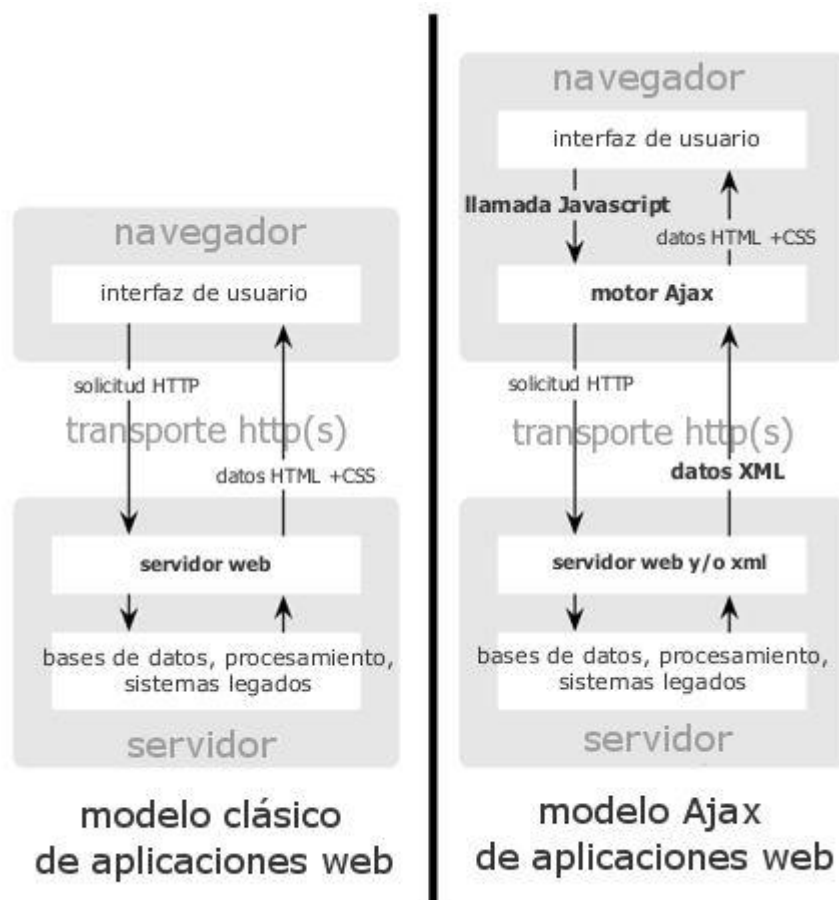


Figura 1. Modelo clásico de aplicación web vs modelo Ajax

3.3 LENGUAJES DE PROGRAMACIÓN (LADO SERVIDOR)

Un lenguaje del lado del servidor es aquel que se ejecuta en el servidor web, justo antes de que se envíe la página a través de Internet al cliente. Las páginas que se

ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red, y otras tareas para crear la página final que verá el cliente.

Los lenguajes de lado servidor más ampliamente utilizados para el desarrollo de páginas dinámicas son el ASP, JSP, PERL y PHP.

3.3.1 .NET

ASP (Active Server Pages) es un lenguaje comercializado por Microsoft, y usado por programadores para desarrollar entre otras funciones, sitios web.

ASP.NET es el sucesor de la tecnología ASP, fue lanzada al mercado mediante una estrategia de mercado denominada .NET [10].

Se desarrolló para resolver las limitantes que brindaba su antecesor ASP. Para el desarrollo de ASP.NET se puede utilizar C#, VB.NET o J#. Los archivos cuentan con la extensión (aspx). Para su funcionamiento de las páginas se necesita tener instalado IIS con el Framework .Net. Microsoft Windows 2003 incluye este framework, solo se necesitará instalarlo en versiones anteriores.

El lenguaje ASP consiste en una serie de clases .NET utilizadas para crear aplicaciones Web, tanto del lado cliente (Web Form) como del lado servidor (Web Service). La integración de nativa .NET Framework con el sistema operativo Windows Server 2003 hace que su ejecución sea más estable y rápida que otros lenguajes de programación.

.NET tiene una serie de limitaciones debido a que es un producto de Microsoft y es que no funciona con todo tipo de servidores, se requiere concretamente de un servidor Web de Microsoft. Si deseamos utilizar esta tecnología en otro tipo de servidores se necesita un software intérprete como por ejemplo Chilisoft, Instant ASP. Dentro de sus ventajas podemos detallar:

- Completamente orientado a objetos.
- Controles de usuario y personalizados.
- División entre la capa de aplicación o diseño y el código.
- Facilita el mantenimiento de grandes aplicaciones.
- Incremento de velocidad de respuesta del servidor.
- Mayor velocidad.
- Mayor seguridad.

Respecto a sus inconvenientes:

- Mayor consumo de recursos.
- Tecnología propietaria.
- Hospedaje de sitios web costosos

Resumiendo este lenguaje, es idóneo para trabajar con clases y es muy fácil de aprender, pero tiene el gran inconveniente que es la limitación de que no puede ejecutarse en cualquier plataforma.

3.3.2 JSP

JSP es un lenguaje para la creación de sitios web dinámicos, acrónimo de Java Server Pages. Está orientado a desarrollar páginas web en Java [11].

JSP es un lenguaje multiplataforma que fue desarrollado por Sun Microsystems.

Comparte ventajas similares a las de ASP.NET, desarrollado para la creación de aplicaciones web potentes. Posee un motor de páginas basado en los Servlets de Java. Para su funcionamiento se necesita tener instalado un servidor Tomcat.

JSP posee un código separado de la lógica del programa. Las JSP se diferencian de otras tecnologías del lado del servidor como las ASP en dos aspectos principalmente: por un lado los JSP y servlets se ejecutan en una máquina virtual Java, lo cual permite que, en principio, se puedan usar en cualquier tipo de ordenador, siempre que tenga instalado esa máquina virtual.

Por otro lado, un programa JSP se compila a un programa en Java la primera vez que se invoca, y del programa en Java se crea una clase que se empieza a ejecutar en el servidor como un Servlet. De esta manera los Servlets no se ejecutan cada vez que se recibe una petición, sino que persisten de una petición a la siguiente, lo que permite realizar operaciones como la conexión a bases de datos o manejo de sesiones de una manera más eficiente.

El código JSP puede ser incrustado en código HTML. Los elementos que pueden ser insertados en las páginas JSP son los siguientes:

- Código, pudiendo incrustar código Java.
- Directivas, que permiten controlar parámetros del Servlet.
- Acciones, permiten alterar el flujo normal de ejecución de una página.

Las principales ventajas de este lenguaje es que posee una muy rápida ejecución del Servlets, se pueden crear páginas del lado del servidor, es

multiplataforma, el código está muy bien estructurado. La parte dinámica está escrita en java, un lenguaje más sencillo.

Respecto a su gran desventaja es muy complejo de aprender.

Por lo tanto, no nos decantaremos por esta tecnología debido a su complejidad aunque posea una serie de ventajas muy buenas como que son ejecutables en cualquier dispositivo.

3.3.3 PERL

PERL (Practical Extraction and Report Language) es un lenguaje de programación desarrollado por Larry Wall inspirado en otras herramientas de UNIX como son: sed, grep, awk, c-shell, para la administración de tareas propias de sistemas UNIX [12].

No establece ninguna filosofía de programación concreta. No se puede decir que sea orientado a objetos, modular o estructurado aunque soporta directamente todos estos paradigmas; su punto fuerte son las labores de procesamiento de textos y archivos.

Lenguaje de programación basado en scripts portable a casi cualquier plataforma. Es muy utilizado para escribir CGIs. Uno de sus elementos más potentes son las expresiones regulares, que a partir de su versión en Perl han sido adoptadas por otros lenguajes y plataformas como .NET o JavaScript.

Es un buen lenguaje “pegamento”. Se pueden juntar varios programas de una forma sencilla para alcanzar una meta determinada. Los usuarios de Windows agradecerán esta propiedad ya que normalmente adolecen de un buen lenguaje tipo “script”. Es relativamente rápido para un lenguaje tipo “script”.

Está disponible en múltiples plataformas y sistemas operativos (UNIX, Linux y Windows). Un programa que se escriba teniendo en cuenta la compatibilidad puede ser escrito en una plataforma y ejecutado en otra. El desarrollo de aplicaciones es muy rápido.

Hay una colección enorme de módulos que pueden ser incorporados a cualquier “script” de Perl. Están disponibles en el CPAN (“Comprehensive Perl Archive Network”). En particular existe una extensión para cálculo numérico denominada PDL.

Perl es gratuito. Mucho más que eso, es “Software Libre”. Esto quiere decir que el código fuente está disponible para que cualquiera lo pueda ver o modificar, y lo que es más importante, siempre lo estará. Aunque nunca pretendas cambiar el código, es importante disponer de la posibilidad de hacerlo, ya que siempre se podrá contratar a una tercera persona para que lo modifique en el caso de que haya un error, y debería ser posible solucionarlo.

Le otorga al programador mucha libertad para que haga el programa como quiera. Tal como dice el eslogan de Perl “Hay más de una forma de hacerlo”.

La principal desventaja de Perl se encuentra en el tiempo de ejecución de un programa, ya que un programa Perl es compilado cada vez que se ejecuta, por lo que puede resultar más lento que un programa similar escrito en otro lenguaje. Sin embargo, se han implementado técnicas para mejorar esta situación como guardar el compilado del programa en memoria y retrasar la compilación hasta que sea necesitada.

Resumiendo, es un lenguaje que posee muchísimas características buenas pero el tiempo de ejecución elevado provoca que no nos decantemos por estas tecnologías para desarrollar nuestra aplicación, ya que necesitamos una mayor rapidez que la proporcionada por PERL.

3.3.4 PHP

PHP (PHP Hypertext Preprocesor) es un lenguaje de programación interpretado, cuyos comandos se ejecutan en el servidor y permiten la creación de documentos HTML dinámicos [13].

Su sintaxis es similar a la de otros lenguajes como C, Perl, Java o JavaScript. PHP es un producto de código abierto, lo que quiere decir que se puede acceder a su código, usarlo, modificarlo y distribuirlo de forma gratuita sin que suponga coste alguno, al contrario de lo que ocurre con los productos comerciales. Esta es, sin duda, una de las principales ventajas de PHP.

PHP se encuentra disponible para muchas plataformas incluyendo Windows, Unix o Linux y con la ventaja de que las aplicaciones desarrolladas en PHP se pueden transportar de una plataforma a otra sin necesidad de modificaciones, es decir que PHP presenta una portabilidad elevada, lenguaje multiplataforma.

Entre las ventajas de PHP se encuentra también la de permitir la conexión y el acceso a multitud de sistemas gestores de bases de datos como MySQL, PostgreSQL u Oracle.

PHP fue concebido en 1994 por Rasmus Lerdorf con la intención de crear un contador para averiguar el número de visitas que recibía su CV virtual. Sin embargo, con el tiempo ha sido adoptado por otros desarrolladores que lo han transformado y convertido en la herramienta que es hoy en la actualidad.

PHP es utilizado por el 81.9% de los sitios web de los que se conoce el lenguaje del lado del servidor utilizado.

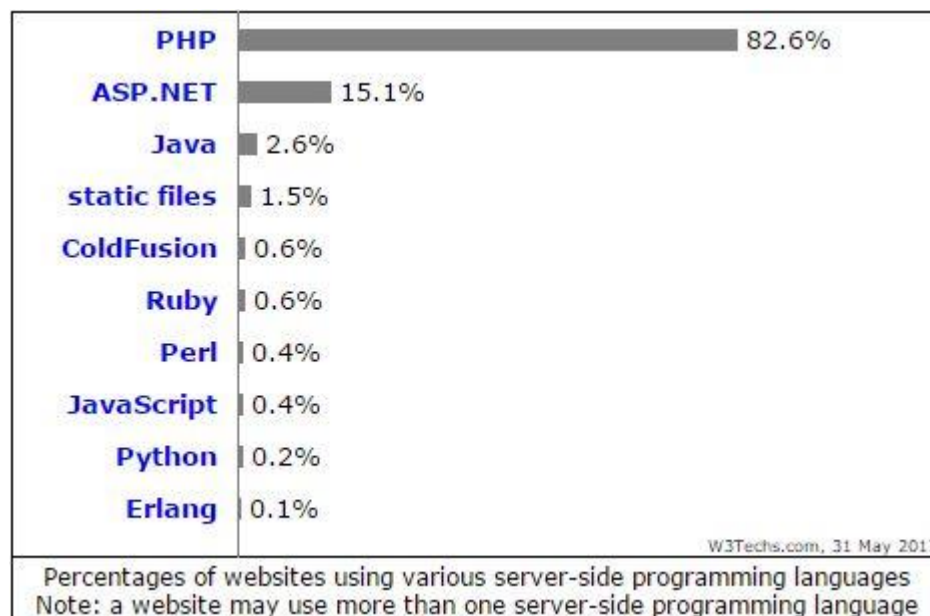


Figura 2. Porcentajes de empleo de lenguajes lado servidor.

PHP es un lenguaje interpretado, no necesita ser compilado para ejecutarse, se ejecuta por medio de un intérprete que se encuentra en el servidor web el cual genera la salida correspondiente para el cliente que solamente recibe una página con el código HTML resultante de la ejecución de la PHP. Como la página resultante contiene únicamente código HTML, es compatible con todos los navegadores.

PHP es un poderoso lenguaje e intérprete, ya sea incluido como parte de un servidor web en forma de módulo o ejecutado como un binario CGI separado, es capaz de acceder a archivos, ejecutar comandos y abrir conexiones de red en el servidor. Estas propiedades hacen que cualquier cosa que sea ejecutada en un servidor web sea insegura por naturaleza.

PHP está diseñado específicamente para ser un lenguaje más seguro para escribir programas CGI que Perl o C, y con la selección correcta de opciones de configuración en tiempos de compilación y ejecución, y siguiendo algunas prácticas correctas de programación.

Aparte de las ventajas comentadas anteriormente, podemos ver una serie de ventajas que hace que PHP sea el lenguaje más utilizado en la actualidad en el lado del servidor:

- PHP es muy eficiente. Mediante el uso de un único servidor, puede servir millones de accesos al día.
- Integración de bases de datos. PHP dispone de una conexión propia a todos los sistemas de base de datos.

- Como se ha diseñado para su uso en la Web, PHP incorpora una gran cantidad de funciones integradas para realizar útiles tareas relacionadas con la Web.

- PHP está disponible para una gran cantidad de sistemas operativos diferentes. Se puede escribir código PHP en todos los sistemas operativos gratuitos del tipo Unix, o en las diferentes versiones de Microsoft Windows. El código funcionará sin necesidad de aplicar ninguna modificación a los diferentes sistemas que ejecute PHP.

- Se dispone de acceso al código fuente de PHP. A diferencia de los productos comerciales y de código cerrado, si se desea modificar algo o agregar un elemento al programa, se puede hacer con total libertad.

A parte de estas ventajas hay que destacar el tema de la seguridad, PHP es un poderoso lenguaje e intérprete, ya sea incluido como parte de un servidor web en forma de módulo o ejecutado como un binario CGI separado, es capaz de acceder a archivos, ejecutar comandos y abrir conexiones de red en el servidor. Estas propiedades hacen que cualquier cosa que sea ejecutada en un servidor web sea insegura por naturaleza.

PHP está diseñado específicamente para ser un lenguaje más seguro para escribir programas CGI que Perl o C, y con la selección correcta de opciones de configuración en tiempos de compilación y ejecución, y siguiendo algunas prácticas correctas de programación

Como conclusión, PHP es uno de los lenguajes más potentes y seguros del lado del servidor el cual es libre y fácil de aprender y que puede ejecutarse en cualquier plataforma.

3.4 BASE DE DATOS (BBDD)

Por último, y lo más importante, estudiaremos las bases de datos. Para ello tenemos que comenzar hablando de lo que es un Sistema de Gestión de Bases de Datos (SGBD), consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a los mismos [14].

Un SGBD permite el almacenamiento, manipulación y consulta de datos pertenecientes a una base de datos, as representaciones integradas de los conjuntos de entidades instancia correspondientes a las diferentes entidades tipo del SI y de sus interrelaciones, organizada en uno o varios ficheros. En el modelo más extendido (base de datos relacional) la base de datos consiste, de cara al usuario, en un conjunto de tablas entre las que se establecen relaciones. A pesar de sus semejanzas (ambos manejan conjuntos de tablas) existen una serie de diferencias fundamentales entre un SGBD y un programa de hoja de cálculo, la principal es que un SGBD permite:

- El método de almacenamiento y el programa que gestiona los datos (servidor) son independientes del programa desde el que se lanzan las consultas (cliente).

- El objetivo fundamental es permitir consultas complejas, cuya resolución está optimizada, expresadas mediante un lenguaje formal.

-El almacenamiento de los datos se hace de forma eficiente aunque oculta para el usuario y normalmente tiene, al contrario de lo que ocurre con las hojas de cálculo, poco que ver con la estructura con la que los datos se presentan al usuario.

-El acceso concurrente de múltiples usuarios autorizados a los datos, realizando operaciones de actualización y consulta de los mismos garantizando la ausencia de problemas de seguridad (debidos a accesos no autorizados) o integridad (pérdida de datos por el intento de varios usuarios de acceder al mismo fichero al mismo tiempo).

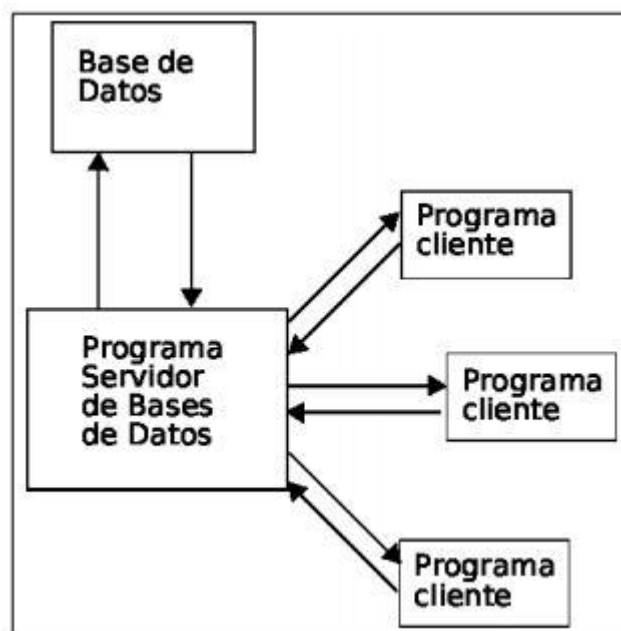


Figura 3. Esquema cliente-servidor de una base de datos.

Como podemos ver en la descripción de lo que es una base de datos, es primordial para nuestro proyecto, por lo que es el punto más importante a la hora de elegir la tecnología idónea para desarrollarla y así obtener la mayor seguridad posible a nuestros datos. Procedemos a explicar los distintos tipos de bases de datos que hay:

-Orientadas a objetos

-Relacionales

Vamos a estudiar por separado los dos tipos de bases de datos.

3.4.1 BASES DE DATOS RELACIONALES

Es el modelo más utilizado hoy en día. Una base de datos relacional es básicamente un conjunto de tablas, similares a las tablas de una hoja de cálculo, formadas por filas (registros) y columnas (campos). Los registros representan cada uno de los objetos descritos en la tabla y los campos los atributos (variables de cualquier tipo) de los objetos. En el modelo relacional de base de datos, las tablas comparten algún campo entre ellas. Estos campos compartidos van a servir para establecer relaciones entre las tablas que permitan consultas complejas [15].

La idea básica de las bases de datos relacionales es la existencia de entidades (filas en una tabla) caracterizadas por atributos (columnas en la tabla). Cada tabla almacena entidades del mismo tipo y entre entidades de distinto tipo se establecen relaciones.

Las tablas comparten algún campo entre ellas, estos campos compartidos van a servir para establecer relaciones entre las tablas.

Los atributos pueden ser de unos pocos tipos simples:

- Números enteros.
- Números reales.
- Cadena de caracteres de longitud variable.

Estos tipos simples se denominan tipos atómicos y permiten una mayor eficacia en el manejo de la base de datos pero a costa de reducir la flexibilidad a la hora de manejar los elementos complejos del mundo real y dificultar la gestión de datos espaciales, en general suponen un problema para cualquier tipo de datos geométricos.

Las relaciones que se establecen entre los diferentes elementos de dos tablas en una base de datos relacional pueden ser de tres tipos distintos:

- Relaciones uno a uno, se establecen entre una entidad de una tabla y otra entidad de otra tabla.
- Relaciones uno a varios, se establecen entre varias entidades de una tabla y una entidad de otra tabla.
- Relaciones varios a varios, se establecen entre varias entidades de cada una de las tablas.

3.4.1.1 SQL, EL LENGUAJE DE LAS CONSULTAS

SQL (Structure Query Language), es un lenguaje de consulta estructurado establecido como un lenguaje de alto nivel estándar para sistemas de bases de datos

relacionales. Fue desarrollado originalmente por IBM y los responsables de publicar este lenguaje como estándar fueron la ANSI (Instituto Americano de Normalización) y la ISO (Organismo Internacional de Normalización) [16].

Al ser un estándar, este lenguaje se puede encontrar en cualquiera de los DBMS relacionales que existen en la actualidad como por ejemplo ORACLE, SQL SERVER, MYSQL.

SQL agrupa tres tipos de sentencias particulares para el manejo de las bases de datos:

- Lenguaje de definición de datos (DDL, Data Definition Language). Permite al usuario crear, eliminar y modificar definiciones de tablas y vistas.
- Lenguaje de manipulación de datos (DML, Data Management Language). Son el grupo de sentencias del SQL para manipular los datos que están almacenados en las bases de datos a nivel de filas y/o columnas.
- Lenguaje de control de datos (DCL, Data Control Language). Conjunto de sentencias para controlar las funciones de administración que realiza el DBMS, como son la atomicidad y la seguridad.

SQL proporciona una elevada seguridad con todos los mecanismos anteriores, justo el punto más importante en la protección de datos.

Como se ha comentado anteriormente, SQL es un lenguaje, pero necesitamos conocer y elegir cual es el sistema de gestión de bases de datos que existen y cual elegiremos para la realización de nuestro proyecto.

3.4.1.2 MySQL

Es un sistema de gestión de bases de datos relacional muy robusto, fue creada por la empresa sueca MySQL AB [17].

MySQL es un software de código abierto, licenciado bajo la GPL de la GNU, aunque MySQL AB distribuye una versión comercial, en lo único que se diferencia de la versión libre, es en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario, ya que de otra manera, se vulneraría la licencia GPL.

El lenguaje de programación que utiliza MySQL es Structured Query Language (SQL) que fue desarrollado por IBM en 1981 y desde entonces es utilizado de forma generalizada en las bases de datos relacionales.

MySQL tiene muchos competidores de los que hablaremos más tarde, como son ORACLE, Microsoft SQL SERVER...

Se pueden destacar las siguientes características principales:

- Velocidad y robustez.
- Soporta gran cantidad de tipos de datos para las columnas.
- Gran portabilidad entre sistemas, puede trabajar en distintas plataformas y sistemas operativos.
- Cada base de datos cuenta con 3 archivos: Uno de estructura, uno de datos y uno de índice y soporta hasta 32 índices por tabla.
- Aprovecha la potencia de sistemas multiproceso, gracias a su implementación multihilo.
- Flexible sistema de contraseñas (passwords) y gestión de usuarios, con un muy buen nivel de seguridad en los datos.
- El servidor soporta mensajes de error en distintas lenguas.

Con las principales características que se han detallado podemos destacar que posee unas ventajas como son la velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento, su bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una máquina con escasos recursos sin ningún problema, la facilidad de configuración e instalación y sobre todo que soporta gran variedad de Sistemas Operativos.

Pero también posee desventajas respecto a otros sistemas gestores de bases de datos, sobre todo a que un porcentaje elevado de las utilidades que posee no están documentadas y que no es intuitivo.

Y lo más importante, la seguridad, posee una baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.

3.4.1.3 ORACLE

Oracle es básicamente una herramienta cliente/servidor para la gestión de Bases de Datos [18].

Es un producto vendido a nivel mundial, aunque la gran potencia que tiene y su elevado precio provocando que sólo se vea en empresas muy grandes y multinacionales, por norma general. En el desarrollo de páginas web pasa lo mismo: como es un sistema muy caro no está tan extendido como otras bases de datos.

Para desarrollar en Oracle utilizamos PL/SQL un lenguaje de 5ª generación, bastante potente para tratar y gestionar la base de datos, también por norma general se suele utilizar SQL al crear un formulario.

Como vemos que es muy caro, desestimamos ORACLE como nuestro sistema gestor de bases de datos.

3.4.1.4 Microsoft SQL Server

SQL Server es un sistema de gestión de bases de datos relacionales (RDBMS) de Microsoft que está diseñado para el entorno empresarial. SQL Server se ejecuta en T-SQL (Transact -SQL), un conjunto de extensiones de programación de Sybase y Microsoft que añaden varias características a SQL estándar, incluyendo control de transacciones, excepción y manejo de errores, procesamiento fila, así como variables declaradas [19].

Bajo el nombre código Yukon en su etapa de desarrollo, SQL Server 2005 fue lanzado en noviembre de 2005. Se dice que el producto 2005 proporcionó una mayor flexibilidad, escalabilidad, confiabilidad y seguridad a las aplicaciones de base de datos, y permitió que fueran más fáciles de crear y desplegar, lo que reduce la complejidad y el tedio involucrado en la gestión de bases de datos. SQL Server 2005 también incluía más soporte administrativo.

El código original de SQL Server ha sido desarrollado por Sybase; a finales de 1980, Microsoft, Sybase y Ashton-Tate colaboraron para producir la primera versión del producto, SQL Server 4.2 para OS/2. Posteriormente, tanto Sybase como Microsoft ofrecieron productos de SQL Server. Sybase cambió después el nombre de su producto a Adaptive Server Enterprise.

Como conclusión, observamos que posee las mismas características que cualquier sistema gestor de bases de datos, pero este no es de código abierto, es posesión de Microsoft.

3.4.2 BASES DE DATOS ORIENTADA A OBJETOS (Modelo OO)

El modelo OO da lugar a las bases de datos orientados a objetos. Es un concepto totalmente distinto al de las bases de datos relacionales que responde al paradigma de la orientación a objetos desarrollado en programación de ordenadores en los últimos años [20].

Al no estar constreñido por el formato de tablas, cuyas columnas responden a tipos atómicos, permite una mayor flexibilidad a la hora de incorporar tipos más complejos como los tipos geométricos (puntos, líneas, polígonos, etc.) por tanto es un modelo, a priori, más adecuado para el trabajo con un Sistema de Información Geográfica.

Se parte del concepto de clase que agrupa a todos los objetos que comparten una serie de atributos, estos atributos pueden incluir la geometría del objeto, las relaciones topológicas y propiedades temáticas.

Por su complejidad, las bases de datos orientadas a objetos no utilizan SQL e incluyen un lenguaje específico para hacer las consultas.

Las bases de datos orientadas a objetos no han tenido, sin embargo, un gran desarrollo, al menos hasta el momento. Entre las causas de este hecho cabe destacar el éxito de SQL y su tremenda eficiencia y el carácter altamente intuitivo de las tablas del modelo relacional. Por ello, se ha desarrollado un modelo híbrido que trata de capturar lo esencial de la orientación a objetos sin perder la eficiencia del modelo relacional. Se trata de las bases de datos objeto-relacionales.

3.5 CONCLUSIONES

Una vez realizado el estudio de todas las tecnologías disponibles, al menos las más importantes y conocidas a días de hoy, procedemos a sacar conclusiones y a la elección de las diferentes tecnologías que utilizaremos para desarrollar nuestro proyecto.

Comenzamos con la elección del lenguaje del lado del cliente, necesitaremos mostrar a los usuarios de la aplicación un resultado visible para realizar las diferentes tareas de mantenimiento, flexibilización y gestión de permisos de la web, por lo tanto necesitaremos HTML, concretamente HTML5 apoyado de CSS3 para proporcionar estilos en modo cascada y de JavaScript, junto a jQuery, para proporcionar eventos, validaciones y funcionalidades a los distintos apartados de la vista, como son los botones. Para el desarrollo necesitaremos de un editor de texto que utilizaremos Notepad++ y Dreamweaver.

Respecto al lado del servidor, vimos ASP, aunque fácil de aprender y muy útil para trabajar con clases, tiene unos inconvenientes que hace que lo descartemos ya que no puede ejecutarse en cualquier plataforma y además tiene un elevado consumo de recursos.

JSP también lo descartamos, aunque posee un elevado rendimiento, su complejidad a la hora de aprenderlo y del propio desarrollo, lo desestimamos utilizar para el proyecto.

La decisión final respecto al lenguaje lado servidor esta entre PHP y PERL. La balanza se decanta por PHP ya que es uno de los lenguajes más potentes y seguros del lado del servidor el cual es libre y fácil de aprender y que puede ejecutarse en cualquier plataforma.

Y por último, queda elegir nuestra base de datos. Elegimos una base de datos relacional ya que tenemos que relacionar las distintas tablas que poseemos con

diferentes datos, en nuestro caso estas relaciones serán para permisos, por lo que elegimos una base de datos relacionales.

Vimos dentro de las bases de datos relacionales el lenguaje que utilizan, el SQL, y ahora necesitamos un gestor de bases de datos entre los que teníamos MySQL, ORACLE y SQL SERVER.

ORACLE lo desestimamos debido a su alto coste y SQL SERVER debido a que pertenece a Microsoft y es un lenguaje no abierto. Nos decantamos por MySQL por sus características y sobre todo por su elevada seguridad que nos otorga. MySQL también es elegido ya que al elegir PHP como lenguaje del lado del servidor provoca que MySQL sea la primera elección como sistema gestor de bases de datos.

Como resumen, se ha elegido PHP como lenguaje del lado del servidor junto a MySQL como sistema gestor de bases de datos y en el lado del cliente las tecnologías de JavaScript, jQuery, HTML5 y CSS.

Dentro del apartado de conclusiones tenemos que explicar una serie de detalles a utilizar en nuestro proyecto.

Tendremos la necesidad de realizar diferentes formularios tanto para la inserción de datos como para modificarlos o eliminarlos para ello necesitamos realizarlo con HTML, para su vista, PHP para la conexión con la base de datos y de JavaScript para mandar los datos y validarlos. Para esta situación poseemos una librería muy potente de PHP como es Zebra_Form.

A lo largo de todo el documento escrito, hemos estado hablando de la flexibilización de la web, pero hay que detallar que también necesitamos realizar una ampliación que consiste en otorgar la funcionalidad de cambio de apariencia de la web, es decir, el cambio de colores de las vistas. Para el desarrollo de este último apartado, utilizaremos un atributo que posee los formularios de php, los botones tipo imagen que posteriormente explicaremos

3.5.1 ZEBRA FORM

Zebra_Form es una librería gratuita de PHP que simplifica el proceso de creación y validación de formularios HTML. Su estructura orientada a objetos promueve el rápido desarrollo de formularios HTML y anima a los desarrolladores a escribir código limpio y fácil de mantener [21].

Se libera a los desarrolladores de la tarea repetitiva de escribir el código de validación de formularios, ofreciendo potente capacidad de lado del cliente y validación del lado del servidor.

El mecanismo de validación de los datos se realiza por medio de jQuery, en una versión superior a 1.5.2 y por el lado del servidor, en PHP.

Esta librería nos proporcionara mucha rapidez y agilidad para validar todos los formularios necesarios en el proyecto.

3.5.2 INPUT = "COLOR" HTML5

El elemento input, será primordial para otorgar la posibilidad de modificar la apariencia en la web [22].

El elemento input, teniendo el valor "color" en su atributo type, representa un campo para la entrada de un color. En los navegadores modernos los campos de color son habitualmente representados por controles que permiten a los usuarios cambiar su valor de manera gráfica, en lugar de tener que ingresarlo directamente como una cadena.

Como hemos comentado anteriormente, no es compatible con todos los navegadores existentes por lo que necesitamos otra tecnología en esos navegadores y la idea es crear hojas de estilo por defecto que el usuario elegirá para modificar la apariencia.



Figura 4. Compatibilidad de `input="color"` en diferentes navegadores.

CAPÍTULO 4

Inicio del proyecto

En el cuarto capítulo se va a explicar en la situación que nos encontramos y el por qué se realizara este proyecto.

Todo se centra en que la aplicación existente para la gestión de tutela de FUNDAMAY se la ve como una potente herramienta para proporcionar ayuda a los trabajadores de este sector y a que es muy rígida actualmente. Surge la necesidad de su flexibilización.

4.1 NECESIDAD DEL PROYECTO

El mundo de hoy en día esta informatizado y globalizado casi al 100% . Gracias a todos los avances conseguidos en el mundo de la informática y las comunicaciones, surgen herramientas muy potentes que hacen al propio ser humano realizar sus tareas más sencillas.

Nos centramos en nuestro propio proyecto del cual no partimos de cero para comenzar a desarrollarlo. Partimos de que una entidad dedicada a la tutela de personas mayores necesitaba ayuda para tener un cierto control a la hora de realizar sus tareas, esto consistía en desarrollarles una aplicación de tutela para llevar los datos de los diferentes pacientes y áreas.

Esta aplicación de la que hablamos ya estaba creada por otros desarrolladores, la cual era una aplicación que tenia diferentes áreas de control de los pacientes. Estas áreas que desarrollaron eran:

- Área social donde se detallan los datos del ámbito social de las personas tuteladas (datos personales, médicos, prestaciones sociales, enseres personales y las urgencias).
- Área económica, para tener en cuenta los ingresos, seguros, obligaciones y las diferentes cuentas bancarias de los tutelados.
- Área jurídica, en el caso de que un paciente tuviera necesidad de tener un control judicial, esta área permitía saber los datos del juzgado, las fechas, retribuciones y las autorizaciones judiciales.
- Voluntariado e informe, son dos áreas para los empleados de FUNDAMAY, en los cuales pueden ver diferentes informes de los pacientes, residencias, juzgados u otros datos.

- SAOS, que se emplea para llevar un control de los asesoramientos realizados por parte de la fundación a cualquier persona que se ponga en contacto con ellas, debiendo rellenar la fecha, tipo y solicitante del asesoramiento.

Viendo todas las áreas que posee la aplicación existente, nos surge la posibilidad de poder flexibilizar la aplicación.

4.2 DESCRIPCIÓN DE LA FLEXIBILIZACIÓN

Una vez visto en la situación en la que nos encontramos pasamos a hablar del proceso de flexibilización en qué consiste y los pasos que vamos a realizar para conseguirlo.

La flexibilización de la aplicación existente por parte de FUNDAMAY, surge debido a que cada día la población mundial está más envejecida debido a que la esperanza de vida de las personas es mayor, provocando que muchas personas mayores en un tiempo de su vida necesiten cierto tipo de ayuda en las tareas diarias o llevar un cierto control sobre alguna necesidad de las personas, provocando que las fundaciones de tutela de personas, cada día surjan nuevas entidades que se dediquen a las mismas actividades o solo a una serie de áreas concretas, por ejemplo solo se centren en tutelas relacionadas con el área económico y social de una persona, o solo con sus datos jurídicos.

Esto nos da la idea de que la aplicación que existe la podemos flexibilizar, esto quiere decir, proporcionar una herramienta concreta a cada una de estas fundaciones que puedan necesitarla sin desaprovechar todo el poder de la propia aplicación.

Por lo tanto, nos centramos en flexibilizar la herramienta que ya existe. Para ello, el primer paso es pedir permiso al propietario de la aplicación, para poder utilizarla y otorgarla esta nueva funcionalidad.

La flexibilización, como vimos anteriormente podíamos otorgarla de tres maneras, pero nos centramos en poder hacer una herramienta genérica para todo tipo de entidades por eso el proyecto que estamos a punto de desarrollar consistirá en elaborar una web gestora de permisos.

4.2.1 GESTOR DE WEBS

Esta web gestora de permisos será una herramienta muy útil e importante para poder proporcionar los servicios necesarios y/o contratados por parte de las entidades.

La web que desarrollaremos consistirá en una aplicación web que tendrá dos tipos de usuarios:

- Empleado de mantenimiento.
- Jefe de mantenimiento, o coordinador.

El empleado de mantenimiento, es una persona que se encargará de realizar tareas de soporte de la herramienta de gestión de tutela jurídica, es decir, proporcionar los permisos necesarios y/o contratados por las diferentes entidades que soliciten utilizar esta herramienta.

Mientras que el coordinador, tendrá esta misma funcionalidad pero tendrá los permisos necesarios para gestionar los propios usuarios de este gestor de webs (añadir nuevos usuarios, eliminarlos, modificar sus datos).

Este gestor debe tener una base de datos con las características necesarias para poder almacenar los permisos que otorgará a la herramienta de tutela proporcionada a las distintas entidades.

4.3 AMPLIACIÓN DEL GESTOR DE TUTELAS

La segunda parte del proyecto nos centramos en un aspecto visual de la aplicación. La aplicación de gestión de tutelas es estática incluso en su apariencia, esto quiere decir, que posee siempre la misma paleta de colores.

Si vamos a proporcionar una herramienta genérica para diferentes entidades es normal que cada fundación posea o desee tener sus propios colores o estilos de la web. Por lo que tendremos que desarrollar una nueva funcionalidad que permita al administrador de la aplicación de tutelas jurídicas tener la posibilidad de modificar los colores de la aplicación de tutela.

Es un proceso sencillo a priori pero que en el análisis técnico de la aplicación se contara con detalle en qué consiste y como se ha elaborado.

4.4 OBJETIVOS Y VENTAJAS DEL PROYECTO

El objetivo principal del proyecto es poder abrirnos a un mercado mayor para poder utilizar la herramienta de tutela que existe.

Al poder expandirnos en el mercado flexibilizando la herramienta de tutela jurídica, podremos dar servicio y ayuda a distintas personas que soliciten ayuda a diferentes entidades de tutela, proporcionando a los empleados de estas fundaciones una herramienta que los facilite las labores de su trabajo.

Con el desarrollo que proponemos y que se va a realizar, conseguiremos tener una herramienta aun más potente y útil sin perder en ningún momento recursos disponibles, esto quiere decir, no otorgar a diferentes entidades áreas de tutela que no necesiten para realizar las labores de trabajo.

Además, tendremos una seguridad mayor ya que podremos incluso gestionar diferentes problemas de la misma desde nuestro propio gestor de aplicaciones.

CAPÍTULO 5

DESCRIPCIÓN Y ANÁLISIS TÉCNICO

Una vez visto los primeros puntos de forma teórica y los objetivos principales a los que deseamos llegar, procedemos a detallar el análisis técnico de la aplicación.

Este análisis técnico detallaremos el diseño y las fases de desarrollo que hemos llevado a cabo a la hora de elaborar el proyecto incluyendo las distintas pruebas que se han realizado para comprobar el perfecto funcionamiento e indicando los problemas surgidos a lo largo del desarrollo. Dividiremos en dos partes el análisis, comenzaremos con la nueva web de gestión de permisos y por último la ampliación realizada en la web de FUNDAMAY.

5.1 NUEVA WEB DE GESTIÓN

El primer paso en la elaboración consiste en crear una nueva web que se encargue de llevar todos los permisos relacionados con la visibilidad de las diferentes áreas de gestión de datos de pacientes tutelados. Para su diseño nos basamos en mantener el mismo diseño que se poseía en la web de FUNDAMAY, para dar un aspecto conocido y tener sus características de diseño.

5.1.1 DISEÑO DE LA WEB

Procedemos a explicar y mostrar como es el diseño y estructura de la web gestora de permisos, como está estructurada y la motivos de este diseño que se ha realizado de la forma que mostramos.

En la imagen número 5 se puede ver como es la web de nuestra aplicación que posteriormente iremos explicando en los diferentes apartados.

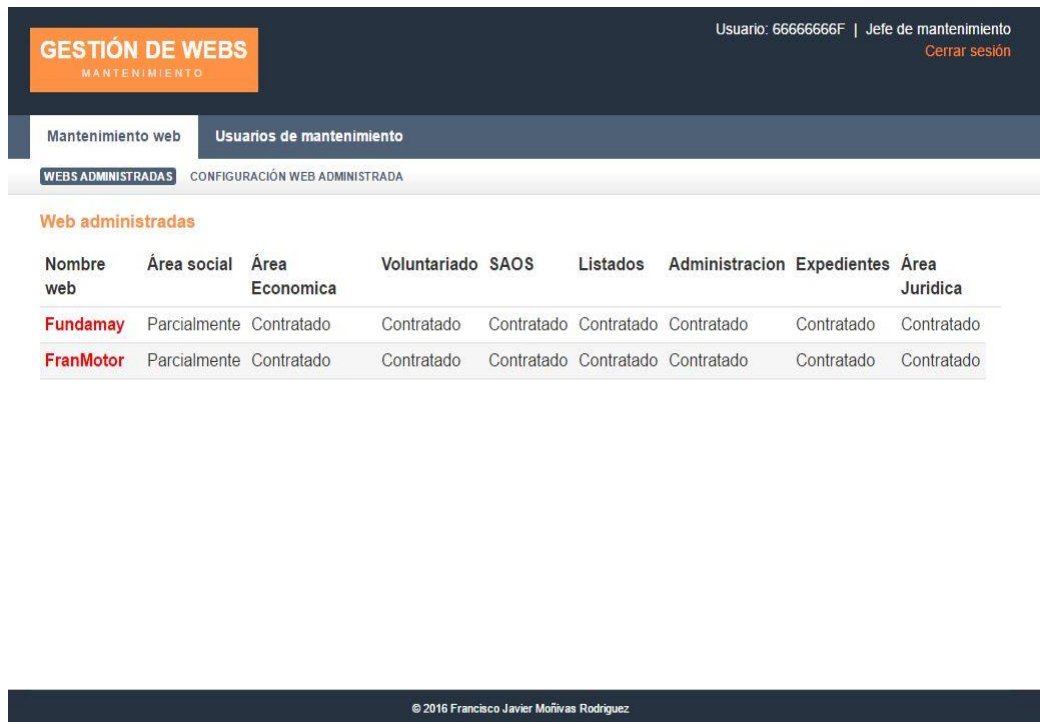


Figura 5. Diseño de la web de gestión de permisos.

5.1.1.1 CABECERA

La cabecera de la web desarrollada se encuentra en la parte superior, la cual es común y estática para todos los usuarios.



Figura 6. Cabecera de la web gestora de permisos.

En la cabecera se pueden distinguir diferentes datos que a continuación se van a explicar:

- Parte superior izquierda: contiene un logotipo de nuestra aplicación e indicar que es una web de mantenimiento.
- Parte superior derecha: nos encontramos con el DNI de la persona encargada de mantenimiento que se encuentra logueado en ese mismo instante para interactuar con la web, el cargo desempeñado por esa misma persona que hay dos cargos disponibles: personal de mantenimiento o jefe de mantenimiento.

Debajo del usuario logueado aparece la opción de cerrar sesión para finalizar la sesión del usuario registrado.

5.1.1.2 MENÚ PRINCIPAL

El menú principal se encuentra situado debajo de la cabecera. Contiene las principales secciones de nuestra web que son dos secciones, cada una con sus submenús.



Figura 7. Menú principal.

Las dos secciones del menú principal son:

- Mantenimiento web: sección que contiene el submenú necesario para mostrar los datos de las distintas aplicaciones de gestión de tutelas que se lleva su mantenimiento y/o control de permisos. También la contiene el submenú con la funcionalidad necesaria para realizar la flexibilización de las webs.
- Usuarios de mantenimiento: sección que solo es visible para el jefe de mantenimiento, contiene el submenú necesario para modificar y gestionar los diferentes usuarios que acceden a esta web.

5.1.1.3 SUBMENÚ

El submenú se encuentra debajo del menú principal. Muestra las distintas secciones contenidas dentro de la sección seleccionada en el menú principal.



Figura 8. Submenú.

El submenú contiene las siguientes subsecciones dependiendo del usuario logueado en la web, siendo un empleado de mantenimiento o un jefe de mantenimiento los dos tipos de usuarios que pueden acceder.

Si se ha accedido a la web siendo un empleado de mantenimiento el submenú que será visible es el siguiente:



Figura 9. Menú y submenú del jefe de mantenimiento (I).



Figura 10. Menú y submenú del jefe de mantenimiento (II).

Si se ha accedido a la región de mantenimiento web se observan dos submenús:

- Webs administradas: muestra las diferentes webs que se las da soporte con los servicios contratados.
- Configuración de webs administradas: muestra las webs que se las da soporte y al seleccionar una de ellas, redirige a una pantalla que muestra los permisos que podemos modificar en esa web.

Si se ha accedido a la región de mantenimiento de usuarios se observan tres submenús:

- Mostrar usuarios: listado de todos los usuarios que tienen acceso a esta web de mantenimiento.
- Nuevo usuario: permite añadir un nuevo usuario para que realice tareas de mantenimiento de permisos de las webs que se administren.
- Configurar usuarios: permite editar los datos de un usuario o eliminarlo.

En el caso de que el usuario sea un empleado de mantenimiento solo tendrá acceso al primer menú, el mantenimiento de webs. Este usuario no tendrá los permisos necesarios para editar usuarios o añadir usuarios a esta web.

5.1.1.4 CUERPO

El cuerpo se encuentra situado en la zona central de la página siendo esta la región más grande e importante.

En esta región se mostraran los diferentes datos que se obtienen tras navegar por los distintos menús de la web. Dentro del mismo cuerpo se pueden distinguir diferentes secciones:

Web administradas								
Nombre web	Área social	Área Económica	Voluntariado	SAOS	Listados	Administración	Expedientes	Área Jurídica
Fundamay	Parcialmente	Contratado	Contratado	Contratado	Contratado	Contratado	Contratado	Contratado
FranMotor	Parcialmente	Contratado	Contratado	Contratado	Contratado	Contratado	Contratado	Contratado

Figura 11. Cuerpo de la web con datos.

Se observan tres zonas bien distinguidas en el propio cuerpo de la web:

- Título de los datos: indica el título de la tabla con los datos que se van a mostrar.
- Cabecera de la tabla: títulos de las secciones de la cabecera de la tabla.
- Datos: contiene los datos a mostrar, en el caso de la imagen indica lo que permisos tienen contratados cada una de las webs.

5.1.1.5 PIE DE PÁGINA

El pie de página se encuentra en la parte inferior de la web, debajo del cuerpo de la web. La región delimita el fin del contenido de la web y contiene la información de copyright (año de desarrollo y persona que realizó el desarrollo de la web).



Figura 12. Pie de página.

5.1.1.6 ESTÉTICA

La estética de una web es muy importante para distinguir las diferentes secciones que contiene la propia web y saber en todo momento en que parte de la web nos encontramos. En la elección de la estética nos basamos en la que se tenía en la web de FUNDAMAY ya que era muy amigable y resultaba muy fácil de saber en qué zona de la web nos encontrábamos.

La paleta de colores a utilizar se muestra en la imagen 13.

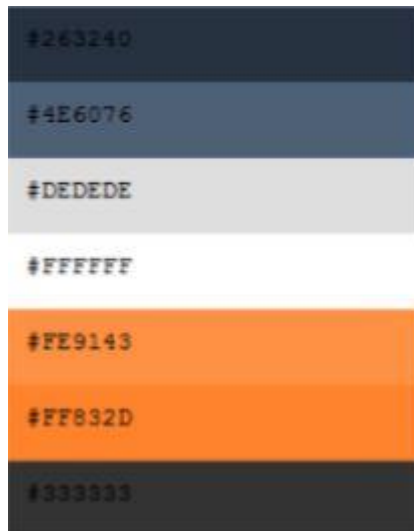


Figura 13. Paleta de colores utilizada en la web de mantenimiento.

Para el desarrollo de la hoja de estilos que proporciona los colores y tipos de escritura se han utilizado hojas de estilo de cascada (CSS) que se encuentran definidas en el archivo style.css situado en la ruta: style/style.css (formato .css).

En esta hojas de estilo podemos encontrar todas las secciones de la web las cuales las estamos proporcionando el estilo mediante estas hojas.

Respecto al tipo de letra utilizada, es recomendable utilizar una que sea fácil de leer y no tenga una serigrafía complicada que provoque confusiones en la lectura, por lo que se ha utilizado la fuente Arial en toda la web.

El fondo de la web siempre será blanco ya que es el fondo por defecto en el que mejor se lee, teniendo las letras del cuerpo siempre colores oscuros dando ese contraste necesario para tener una buena lectura.

Las hojas de estilo, en la sección de ampliación de la web, hablaremos un poco más ya que serán de vital importancia para proporcionar la capacidad de dar diseño a la web de FUNDAMAY por medio de su administrador.

5.1.2 USABILIDAD DE LA NUEVA WEB

La parte más importante del diseño es otorgar una usabilidad sencilla. La Usabilidad es la medida de la calidad de la experiencia que tiene un usuario cuando interactúa con un producto o sistema. Los parámetros que se han tenido en cuenta en el desarrollo son:

-Compatibilidad: debido al elevado número de navegadores existentes, hay que tener en cuenta que no todos los navegadores soportan todas las características de HTML o CSS por lo que se ha tenido en cuenta como algo primordial que tenga todas las funcionalidades en cualquier navegador.

-Simplicidad: la web es lo más sencilla y legible posible siendo muy amena para el usuario. Es fácil de utilizar gracias a su simplicidad. A todas las secciones se pueden llegar accediendo a través del menú principal con sus submenús no hay que hacer nada especial para llegar a un punto concreto.

- Utilización de formularios: con la ayuda de Zebra_form, realizamos los formularios necesarios para el uso de esta web indicando que campos son obligatorios rellenar y cuáles no. Es muy importante su uso y el de su validación ya que agiliza la web.

-Utilización de pops-ups: para confirmar las acciones que se realizan en la web en el caso de añadir usuarios o modificar datos, se podían avisar a los usuarios por medio de ventanas emergentes lo cual es molesto y a veces desesperante por lo que se ha decidido prescindir de ello.

-Rapidez: es muy importante que el tiempo de carga de una web sea lo más rápido posible por ello es aconsejable no cargar dominios externos en la web al igual que utilizar en toda la web siempre los mismos scripts y estilos para no tener que cargar en cada navegación diferentes hojas de estilo u hojas de scripts.

Con todo lo anterior conseguimos tener una usabilidad bastante buena para el usuario pero también utilizamos iconos e hipervínculos para hacer más intuitivo algunas acciones y simplificar la web como son el caso de modificar a los usuarios, eliminarlos o mostrar los datos de un usuario concreto.

Personal de mantenimiento

Apellidos	Nombre	DNI	Email	Tlf. 1	Permisos	
Moñivas Rodríguez	Francisco Javier	66666666F	francisco@alumnos.uva.es	665 00 96 88	Jefe de mantenimiento	 
Muñoz	Javier	11111111N		665 00 96 88	Empleado de mantenimiento	 

Figura 14. Iconos para simplificar la web.

5.1.3 ORGANIZACIÓN DE LOS ARCHIVOS

El código fuente de la web esta distribuidos en diferentes archivos y estos archivos se encuentran situados en diferentes directorios para proporcionar una fácil búsqueda de archivos en caso de necesitar encontrar algún archivo necesario.

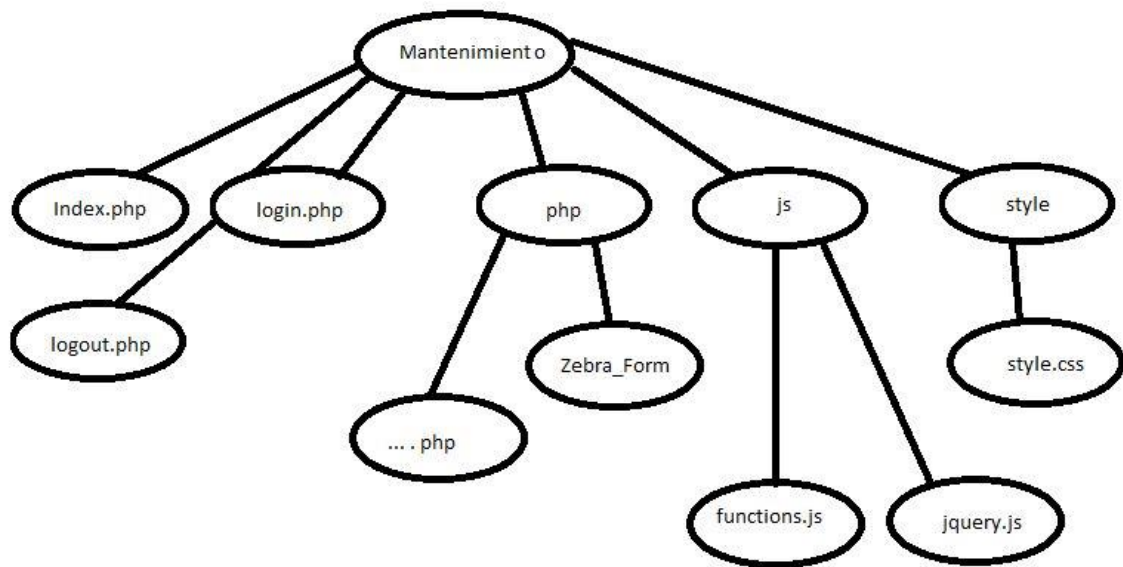


Figura 15. Distribución de los archivos.

Los archivos están distribuidos en diferentes directorios en los que podemos distinguir la raíz que es el directorio Mantenimiento que contiene todos los archivos.

Dentro de Mantenimiento encontramos tres directorios más y archivos de código:

-Index.php: es el archivo más importante de todo el proyecto. Con este archivo obtendremos el acceso a todas las secciones de la web, sin él no podemos acceder a ninguna sección de la web. En él se inicia la sesión PHP, realizando la conexión a la base de datos y comprobamos los permisos al acceder a las secciones.

-login.php: archivo que contiene el código php necesario para comprobar las credenciales del usuario y permite o no acceder a la web.

-logout.php: destruimos la sesión PHP iniciada en el index.php y cierra la sesión del usuario.

En los directorios contenidos dentro de Mantenimiento hay 3 que son los siguientes:

- Directorio js: directorio que contiene todos los archivos con extensión js con su código fuente escrito en JavaScript. Por medio de estos archivos js implementamos diferentes funcionalidades de la web.
 - Functions.js: archivo JavaScript que contiene todas las funciones necesarias que se utilizan para las peticiones AJAX, comprobar formularios, mostrar diálogos y datos...

- Archivo jquery.js: corresponde a los archivos descargados con la última versión del framework jQuery. Es utilizado por functions.js y por la librería Zebra_form.
- Directorio style: este directorio contiene imágenes y el código fuente CSS para proporcionar los estilos a la web.
- Directorio php: en el encontramos todos los archivos con extensión .php y podemos distinguir diferentes tipos de archivos:
 - _header.php: archivo que contiene el código php de la cabecera de la web.
 - _footer.php: archivo que contiene el código php del pie de página de la web.
 - _functions.php: archivo que contiene el código php para crear distintas funciones desarrolladas que complementan a las desarrolladas en JavaScript.
 - man_*.php: conjunto de archivos que contienen todo el código php que hacen referencia a la región de mantenimiento de las webs de tutela jurídica.
 - Usu_*.php: conjunto de archivos que contienen todo el código php que hace referencia a la región de mantenimiento de usuarios de la web de gestión.
 - Zebra_Form: otro directorio que contiene la librería de php Zebra_Form. Dentro de este directorio, hay diversos archivos descargados, necesarios para poder utilizar dicha librería.

Ya se ha visto la forma de distribuir los archivos de la web, hay que destacar que el archivo index.php es el más importante y con el proporcionamos una seguridad a nuestra web no permitiendo acceder a ninguna región si no es a partir del index.php.

5.1.4 NAVEGACIÓN POR LA WEB

Como hemos visto en las secciones anteriores, podemos ver que nuestra web tiene distintas secciones de navegación que todas parten del index.php. En esta sección se va a comentar el diagrama de flujo de la web para navegar por las distintas secciones de las que dispone de la web.

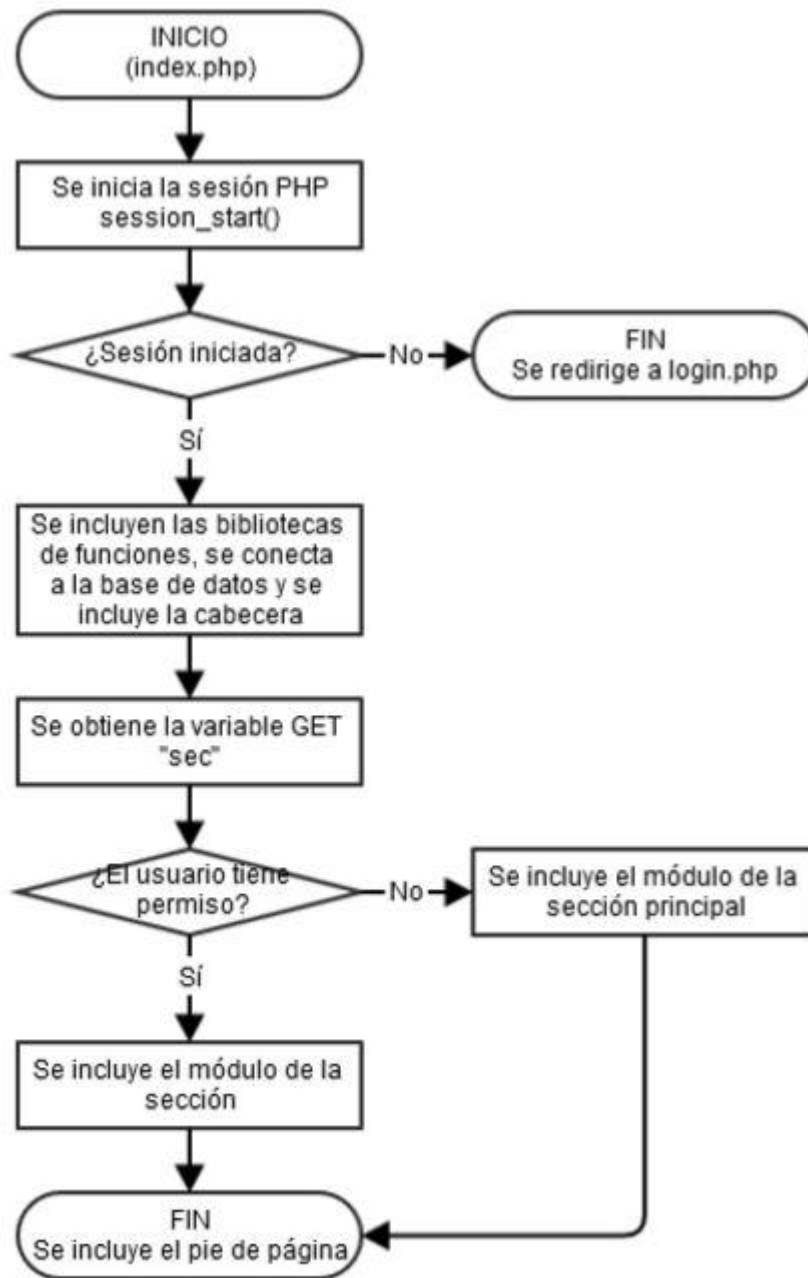


Figura 16. Diagrama de flujo carga de una página de la web.

El diagrama de flujo es el siguiente.

-Se accede al index.php que contiene todos los archivos de la web, desde donde son cargados.

-El primer paso es redirigir al usuario al archivo login.php donde mostrará una pantalla de loguearse al propio usuario y a la vez se inicia la sesión PHP que se mantiene abierta hasta que se acceda a logout.php.

Iniciar sesión

DNI *	<input type="text"/>
Contraseña *	<input type="password"/>
	<input type="button" value="Entrar"/>

Figura 17. Página de login.

-El usuario introduce su usuario y contraseña, si es correcto accede a la página principal con el proceso de incluir todas las bibliotecas, funciones y librerías necesarias en la aplicación. Este proceso por el que se añaden todos estos archivos es invisible al usuario. Si el login es incorrecto nos redirige al logout.php poniendo fin a la sesión php.

-Cuando el usuario está logueado se obtiene la variable GET "set" necesaria para ir accediendo y cargando los distintos archivos de la web.

-Se cargan los permisos del usuario, si no tiene permisos para algún área concreta, esta área no se carga y directamente se carga el pie de página, si por el contrario, el usuario si tiene los permisos se cargan las distintas áreas y por último el pie de página.

Este proceso se realiza siempre que ingrese un usuario. El proceso de navegación por las distintas páginas de la web se hace por medio de variables GET. Dependiendo de a que sección de la web, se irán cargando unas u otras variables GET. Las distintas variables GET para navegar por la web son las siguientes:

-man_*: teniendo * dos posibles valores, web (con el cual vemos los datos principales de las webs que tenemos proporcionando los permisos) y modificar (muestra las webs para posteriormente ir a modificar sus permisos)

-usu_*: teniendo * tres posibles valores, mostrar (se muestran los usuarios de la web), nuevo (añadir nuevo usuario), configurar (de los usuarios que existen, permite modificar los datos de ellos).

Todo el diseño de la web está basado en el uso de index.php para acceder así a todas las secciones y este diseño proporciona una serie de ventajas respecto a otros diseños.

Dentro de las ventajas que proporciona, la primera es la seguridad respecto a otros diseños ya que no se puede acceder a ningún archivo que no esté contenido en el index.php.

También proporciona un ahorro de código a la hora de desarrollar el proyecto ya que al inicio se incluyen librerías comunes a toda la web como lo son la cabecera y el pie de página.

Y la más importante para nosotros, es que este diseño proporciona un control de permisos de visión de las distintas secciones de la web. Esto proporciona una gran seguridad y control de datos ya que usuarios no permitidos en una sección concreta no tendrá forma alguna de acceder a dicha sección.

5.1.5 PERMISOS DE LA WEB

La web cuenta con un sistema de permisos para acceder a las distintas áreas según el tipo de usuario que se encuentre logueado en cada instante para realizar tareas de mantenimiento. Como hemos comentado anteriormente tenemos dos tipos de usuarios en la web de los que vamos a hablar a continuación.

Hay dos tipos de usuarios que son:

- Jefe de mantenimiento: el cual tiene todos los permisos para ver todas las secciones de esta web de mantenimiento.

- Empleado de mantenimiento. Solo tiene los permisos para ver la sección relacionada con las webs administradas.

Para poder ver de forma más intuitiva lo que cada tipo de usuario ve en la web, se detallara en la siguiente tabla indicando las distintas secciones y subsecciones de la web. En la tabla se indica mediante la afirmación sí, cuando un usuario tiene permisos para ver esa área o sección concreta.

Hay que explicar, que se ha realizado esta distinción entre los dos tipos de usuarios ya que se ha solicitado y buscado información en diferentes tipos de empresas que trabajan con sistemas de permisos y nos hemos basado en el utilizado en la empresa Techrules S.A la cual trabaja solo con dos tipos de usuarios a la hora de otorgar un mantenimiento en su producto.

Una vez elegido el sistemas de permisos ya mostramos la tabla que nos mostrara lo que cada usuario podrá ver o no en la web.

	Jefe de Mantenimiento	Empleado de mantenimiento
Mantenimiento de las webs		
Webs administradas	SI	SI
Configurar webs administradas	SI	SI
Usuarios de mantenimiento		
Mostrar usuarios	SI	
Nuevo usuario	SI	
Configurar usuarios	SI	

Figura 18. Tabla de permisos de usuarios.

El jefe de mantenimiento, normalmente solo hay uno o dos usuarios en la propia web con este rango y será el único que podrá acceder a la sección de Usuarios de mantenimiento. En esta sección podemos realizar tareas de mantenimiento de los propios usuarios de esta web que consistirá en mostrarlo, crear nuevos usuarios y configurar sus datos e incluso permisos.

El empleado de mantenimiento, no tendrá el rango necesario para poder acceder a la sección de Usuarios de mantenimiento pero si podrá acceder a la sección de Mantenimiento de webs en la cual se muestran las webs que tendremos para realizar la configuración de sus permisos y realizar tareas de modificación de estos permisos.

5.1.6 FORMULARIOS DE REGISTRO Y CONTROL DE PERMISOS

La web de mantenimiento necesita de muchos formularios para la creación de usuarios o para modificar los permisos de las webs de una forma más sencilla por eso se necesita de alguna herramienta que facilite este proceso y que ahorre tiempo.

Para ello se ha utilizado la librería Zebra_Form de PHP para el manejo de formularios, la cual nos permite definir por medio de PHP los campos del formulario, si son obligatorios rellenarlos, añadir restricciones y además proporciona la herramienta de validación por medio de jQuery tanto a la hora de llenar los distintos campos como cuando se envía para almacenar estos datos en la BBDD.

El framework Zebra Form permite multitud de tipos de controles distintos. En el desarrollo se la plataforma se han utilizado los controles que mejor se adaptan al tipo de contenido, utilizando controles de texto y campos de texto para permitir libertad en las respuestas y el resto de controles para mostrar opciones predefinidas.

En la siguiente imagen se pueden ver los controles utilizados con su apariencia dentro del formulario.



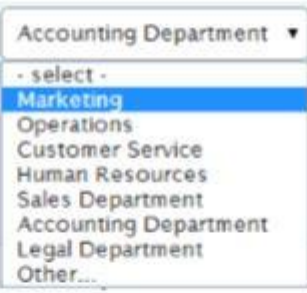

Control	Imagen de muestra
Control de texto	
Campo de texto	
Control de opción	<input checked="" type="radio"/> Sí <input type="radio"/> No
Control de verificación	<input type="checkbox"/> Flipchard and pens <input checked="" type="checkbox"/> Plasma TV screen <input type="checkbox"/> Coffee, tea and mineral water
Lista desplegable	
Control de fecha	

Figura 19. Controles formularios Zebra_Form.

El diagrama de flujo de cómo funciona los formularios por medio de Zebra_Form lo mostramos en la imagen 20.

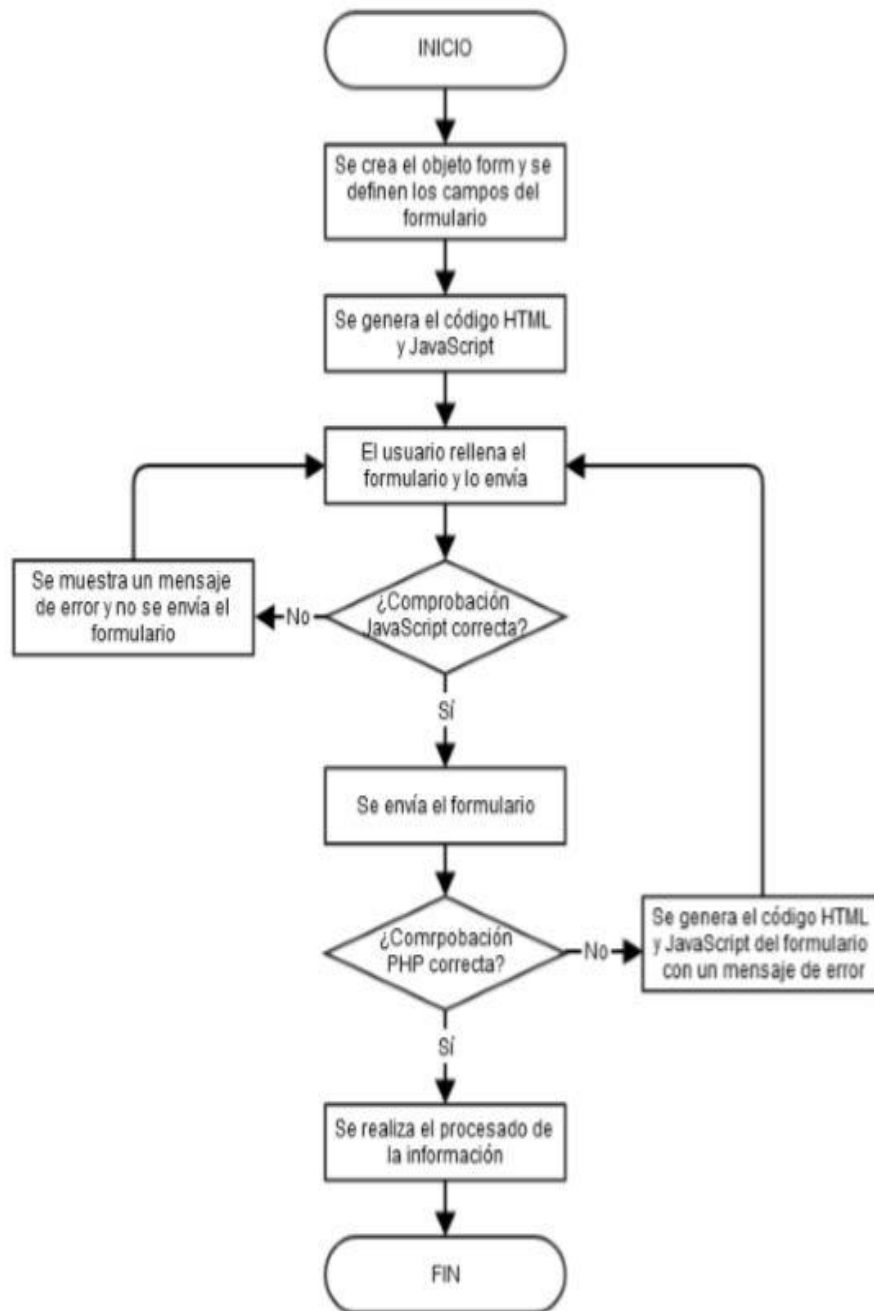


Figura 20. Diagrama de flujo de los formularios.

El diagrama de flujo de cómo funciona los formularios por medio de Zebra_Form lo vamos a explicar detalladamente.

- Inicialmente creamos por medio de php un objeto del tipo form y generamos los campos como se muestra en el fragmento de código siguiente en el que creamos un formulario para crear un nuevo usuario. Se puede observar también que se crean los campos para proporcionar los permisos al usuario siendo este un jefe de mantenimiento o un empleado de

mantenimiento. También se ve la etiqueta creada para añadir el nombre del usuario.

```
// instantiate a Zebra_Form object
$form = new Zebra_Form('form');

// Estado activo por defecto
$form->auto_fill(array(
    'password' => randomPassword()
), true);

// Permisos
$form->add('label', 'label_permisos', 'permisos', 'Permisos:');
$obj = $form->add('select', 'permisos');
$obj->add_options(array(
    '1' => 'Empleado de mantenimiento',
    '10' => 'Jefe de mantenimiento',
));
$obj->set_rule(array(
    'required' => array('error', ';Introduzca el permiso!')
));

// Nombre
$form->add('label', 'label_nombre', 'nombre', 'Nombre:');
$obj = $form->add('text', 'nombre', '', array('autocomplete' => 'off'));
$obj->set_rule(array(
    'required' => array('error', ';Introduzca el nombre!'),
));
```

Figura 21. Fragmento de código. Crear nuevo usuario.

- Una vez creado el formulario, este se crea por medio de JavaScript y HTML. Posteriormente el usuario de la web tiene que introducir los datos necesarios para completar el formulario. Mostraremos uno de los formularios para ver cómo queda su estilo una vez creado por medio de Zebra_Form.

Vamos a explicar de forma más detallada el formulario más importante de la web de mantenimiento que hace referencia a otorgar permisos a las diferentes webs a las que proporcionamos el servicio de permisos de las diferentes áreas.

Por medio de Zebra_Form creamos este formulario el cual dispone de distintas áreas que hacen referencia a todas las secciones de las que se dispone en la web de gestión de tutela jurídica. En la siguiente imagen se muestra un fragmento del formulario, el cual tiene un manejo muy sencillo en el que el usuario solo tiene que marcar la opción deseada (si o no), para proporcionar los permisos a la web de tutela jurídica.

Editar permisos web contratados

Nombre web: *

ÁREA SOCIAL

Contactos: *

- sí
 no

Enseres: *

- sí
 no

Medico: *

- sí
 no

Figura 22. Formulario de permisos.

Nuevo usuario de mantenimiento

Permisos: *	<input type="text" value="- Seleccionar -"/>
Nombre: *	<input type="text"/>
Apellidos: *	<input type="text"/>
DNI: *	<input type="text"/>
Password: *	<input type="text" value="IB2aWDku"/> <small>▸ La contraseña debe tener entre 6 y 15 caracteres.</small>
Provincia:	<input type="text" value="- Seleccionar -"/>
Municipio:	<input type="text"/>
Dirección:	<input type="text"/>

Figura 23. Ejemplo de formulario.

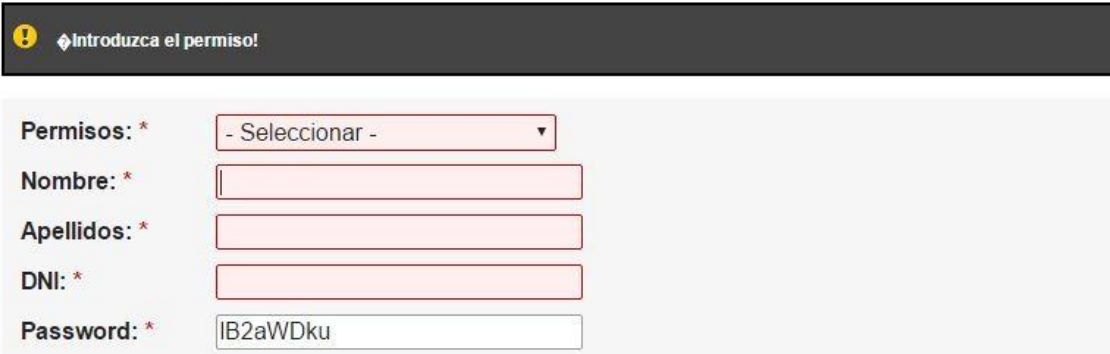
- Cuando se rellenan todos los campos se procede a enviar el formulario, con una comprobación de que los campos hayan sido completados correctamente. Este proceso se realiza por medio de JavaScript, y en caso de no

se correcto alguno de los campos se avisa al usuario por medio de un mensaje e indica que campo no es el correcto.

-En el momento de que se han comprobado que los campos son correctos se envía el formulario y se realiza una segunda comprobación por medio de PHP. Si es correcta esta comprobación se realiza un procesado de la información para mandarla a la base de datos, en caso de que sea incorrecto se vuelve a mostrar el formulario al usuario.

Una vez explicado los pasos principales en la creación de formularios por medio de Zebra_Form hay que destacar que continuamente va mostrando mensajes de ayuda al usuario en caso de no introducir correctamente algún campo o que falta algún campo por cumplimentar como se muestra en la imagen siguiente.

Nuevo usuario de mantenimiento



The screenshot shows a web form titled "Nuevo usuario de mantenimiento". At the top, a dark grey banner contains a yellow warning icon and the text "Introduzca el permiso!". Below this, the form fields are: "Permisos: *" with a dropdown menu showing "- Seleccionar -"; "Nombre: *" with an empty text input; "Apellidos: *" with an empty text input; "DNI: *" with an empty text input; and "Password: *" with a text input containing "IB2aWDku". All input fields have a light red border, indicating they are required or have an error.

Figura 24. Mensajes de ayuda en los formularios.

5.1.7 SEGURIDAD

La seguridad de los datos es lo más importante en cualquier aplicación web por lo que a lo largo de nuestro desarrollo hemos utilizado diferentes técnicas para proporcionar la mayor seguridad de estos datos para no tener robos ni pérdidas de ellos.

Para conseguir desarrollar una plataforma que sea segura se han empleado diversas técnicas tanto a la hora de programar como a la hora de otorgar permisos de usuarios de acceso para que no puedan acceder a ver datos primordiales.

En la técnica se ha añadido una línea de código primordial, para que no podamos acceder a los archivos si no es por medio del index. Está presente en todos los archivos con extensión .php del proyecto.

```

<?php
// No se puede abrir el fichero directamente por seguridad, tiene que ser a través del index
if(!isset($GLOBAL_INC) || $GLOBAL_INC != true) {
    header('Location: index.php');
    exit();
}
?>

```

Figura 25. Fragmento de código para proporcionar seguridad.

En la técnica se ha añadido una línea de código primordial, para que no podamos acceder.

A parte de estas medidas que se han comentado, se ha preparado la aplicación para que se use una conexión segura mediante el uso de HTTPS en el servidor Apache y de un certificado a partir de la biblioteca OpenSSL.

La versión que se ha utilizado de OpenSSL es la 1.0.1g que corrige la vulnerabilidad conocida como Heartbleed, presente en la versión f.

Para asegurar los datos se usa la función `mysqli_real_escape_string`, esta escapa los caracteres especiales de una cadena para usar el dato en una sentencia SQL. Todos los datos que se envían a través de los métodos GET y POST van a pasar por esta función, quedando así protegidas las consultas realizadas a la base de datos.

5.1.8 CASOS DE USO

UML (Lenguaje Unificado de Modelado) es un lenguaje de modelado cuyo vocabulario y sintaxis están ideados para la representación conceptual y física de un sistema. Estos modelos son precisos, completos y pueden ser trasladados directamente a una gran variedad de lenguajes de programación.

Como puede verse en la siguiente tabla existen dos actores en el sistema, que mostraremos todos de los diagramas UML para el acceso a las distintas áreas de la web.



Actores del sistema	Nombre del actor simplificado
	Jefe de mantenimiento
	Persona de mantenimiento

Figura 26. Actores del sistema.

Vamos a ir detallando todos los casos de usos de los dos actores en la web. Comenzamos con el diagrama de acceso al sistema. Todos los usuarios de la web necesitan loguearse para poder acceder al sistema, iniciando la sesión en la plataforma introduciendo su DNI y su contraseña. Dentro de la web, los usuarios van a poder cerrar su sesión y liberar las variables de sesión creadas al loguearse.

El jefe de mantenimiento es la única persona que es capaz de añadir nuevos usuarios y de editarlos en la base de datos. Para poder añadir un nuevo usuario necesita introducir el DNI, nombre, apellidos y la función que va a desempeñar este nuevo usuario en la web.

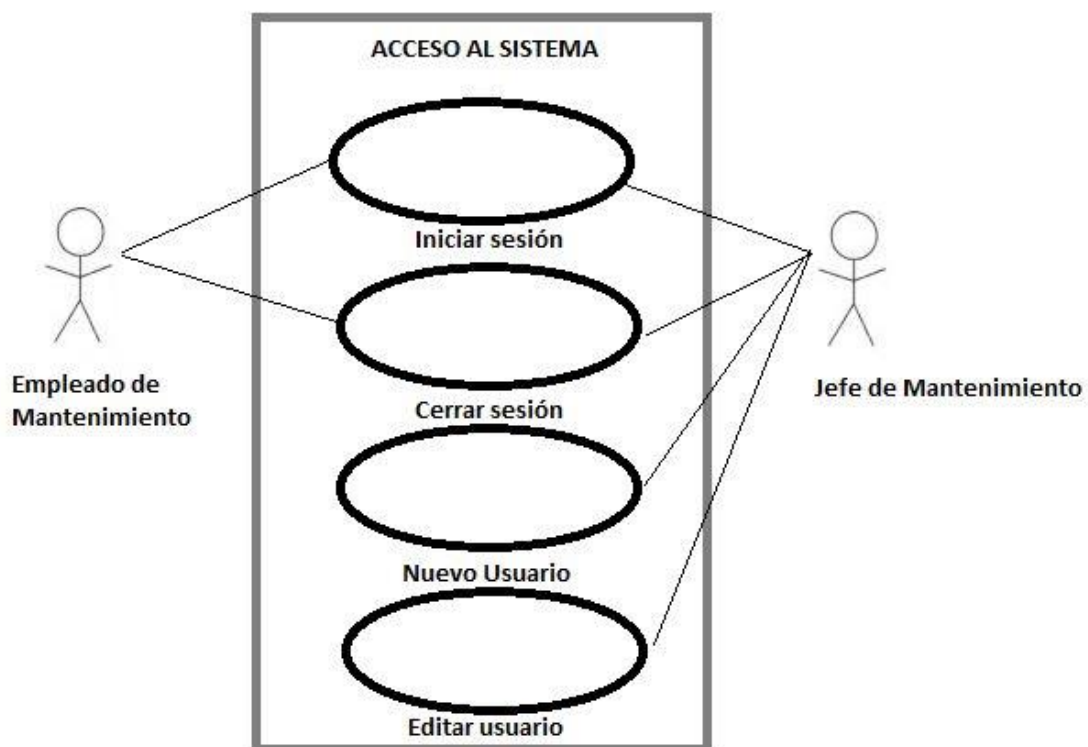


Figura 27. Diagrama de casos de uso de acceso al sistema.

Procedemos a crear los diagramas de casos de uso para las dos secciones principales que tenemos en la web.

Comenzamos con la sección de mantenimiento de las webs donde ambos actores pueden ver todas las secciones disponibles ya que es la función principal de la web se los otorga el privilegio de poder ver a todos los usuarios esta sección con sus respectivas subsecciones.

En la sección de mantenimiento de las webs tenemos dos acciones las cuales una es mostrar los datos con los permisos contratados por cada una de las webs administradas, y la segunda y la más importante en este proyecto, es la configuración de las webs administradas.

Esta acción de configuración de las webs administradas es la sección en la que los actores del sistema tienen la capacidad de editar los permisos de visión de las diferentes áreas contratadas por las fundaciones en la herramienta de gestión de tutelas jurídicas.

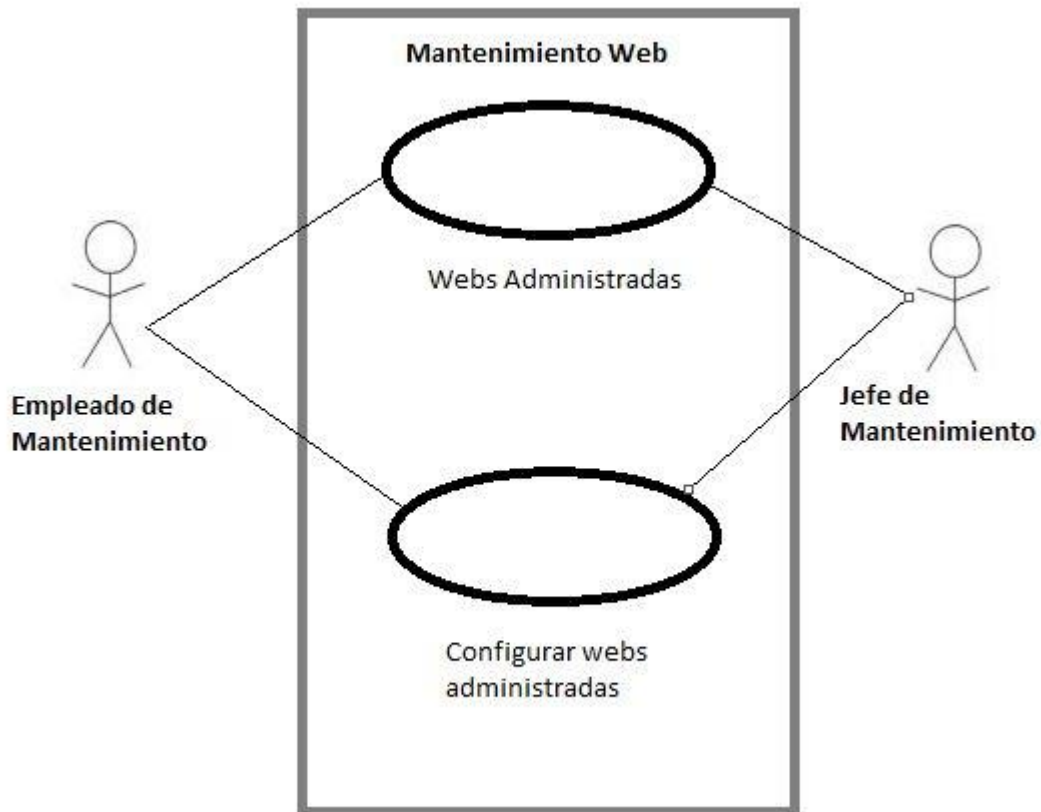


Figura 28. Diagrama de casos de la sección del mantenimiento de las webs

El último de los diagramas de casos de hace referencia a la sección de mantenimiento de usuarios de la web, en esta sección no tenemos acceso con todos los actores disponibles en el sistema sino que solo puede acceder los que posean un rango de jefe de mantenimiento.

En esta sección tenemos tres acciones las cuales solo están disponibles para el jefe de mantenimiento. Las acciones consisten en mostrar, añadir y configurar los diferentes usuarios que estén en la base de datos de nuestra aplicación.

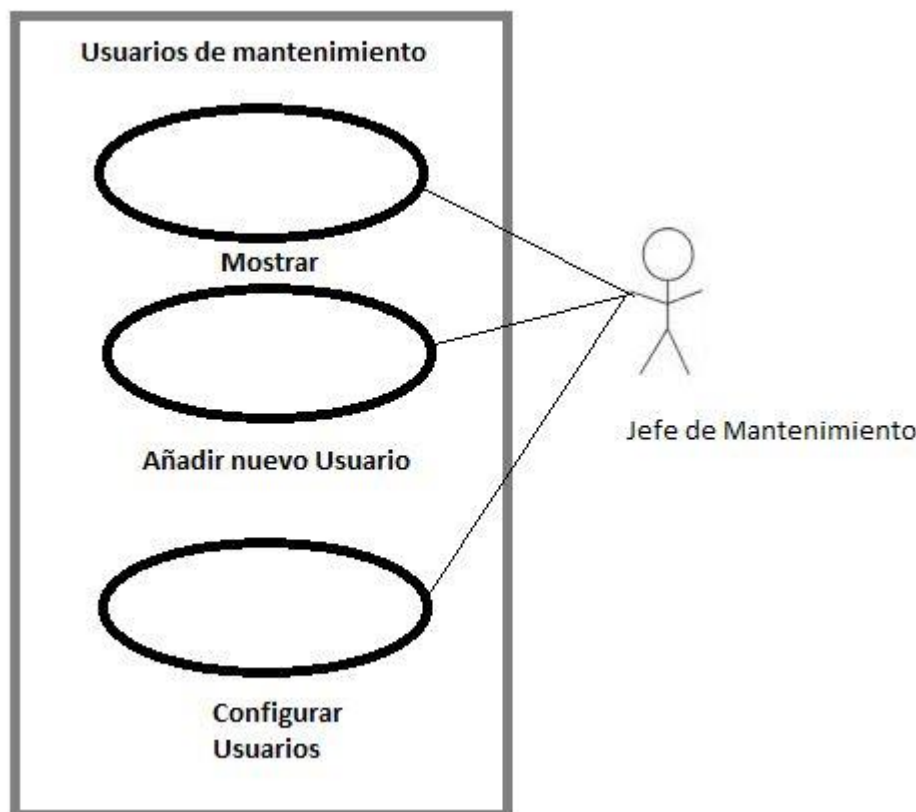


Figura 29. Diagrama de casos de uso de la sección de mantenimiento de los usuarios.

Como observamos, solo tenemos tres casos de uso en nuestra aplicación, lo hemos intentado comprimir al máximo para tener una aplicación lo más sencilla de manejar y que la funcionalidad sea la idónea.

5.1.8 BASE DE DATOS

Lo más importante de todo el proyecto es crear una base de datos que almacene con total seguridad, todos los datos necesarios para proporcionar esa flexibilidad deseada por medio de permisos de las distintas áreas.

Por lo tanto, se ha creado una base de datos llamada mantenimiento, que está formada por dos tablas, una hace referencia a los usuarios que tienen acceso a la web de mantenimiento, y la otra almacena las variables necesarias para otorgar los permisos necesarios a la web de tutela jurídica. No hay relación entre ambas tablas, ya que una contiene la información de los usuarios que pueden acceder al sistema y la otra tiene la información necesaria para que la aplicación de gestión de permisos pueda ver las distintas áreas de las que esta dotada.

Las dos tablas son llamadas:

- Usuarios.
- Websadmisnitradas.

Comenzamos hablando de la tabla de usuarios. Es una tabla de la base de datos que contiene la información de todos los usuarios que se encuentran almacenados en la web.

#	Nombre	Tipo
1	id_usuario 	int(11)
2	dni	varchar(9)
3	passwd	varchar(15)
4	nombre	varchar(100)
5	apellidos	varchar(100)
6	email	varchar(100)
7	telefono1	varchar(9)
8	telefono2	varchar(9)
9	provincia	varchar(1)
10	municipio	varchar(100)
11	direccion	varchar(100)
12	cp	varchar(5)
13	titulacion	varchar(100)
14	profesion	varchar(100)
15	permisos	int(3)

Figura 30. Tabla usuarios de la base de datos.

La tabla está formada por 15 filas, donde cada una almacena información de un usuario. Las distintas filas son:

-id_usuario: variable de tipo int y es la clave de la tabla, identifica a un usuario del resto.

-dni: tipo varchar de longitud 9 para introducir y almacenar el dni con la letra del usuario.

-passwd: tipo varchar, acepta tanto números como letras y almacena la contraseña de cada usuario.

-nombre: varchar, almacenamos el nombre de cada usuario.

-apellidos: varchar, almacena los apellidos del usuario.

-email: varchar, en caso de tener correo electrónico, lo almacenamos en esta variable de la base de datos.

-telefono1 y telefono2: Ambos varchar de longitud 9 para almacenar los dígitos de los números de teléfono.

-provincia: varchar, almacena la provincia del usuario.

-municipio: varchar, almacenamos el municipio donde vive el usuario.

-dirección: varchar, almacenamos la dirección del usuario.

-cp: varchar de longitud 5, almacenamos el código postal del usuario.

-titulación: varchar, sirve para indicar la titulación del usuario.

-profesión: varchar, almacenamos la profesión del usuario.

-permisos: int, variable muy importante para indicar si un usuario es jefe de mantenimiento o empleado de mantenimiento.

Esta tabla tiene toda la información necesaria para registrar a los usuarios y distinguirlos unos entre otros. Hay campos que no son obligatorios completarlos, pero otros son obligatorios.

Los campos obligatorios a rellenar son el id_usuario, el cual se crea al introducir un nuevo usuario en la base de datos y se hace de forma autoincremental. No puede haber dos usuarios con el mismo id_usuario.

Los datos del DNI, nombre, apellidos, y passwd también han de estar llenos en todos los usuarios.

Y el campo permisos es muy importante, es obligatorio que esté lleno ya que servirá para que un usuario vea o no las distintas secciones de la web.

La segunda de las tablas hace referencia a obtener los permisos de la web de gestión de tutelas jurídicas.

La tabla estará formada por 46 columnas que hacen referencia a otorgar permisos o no a cada una de las secciones de las webs de tutela jurídica. Todos los campos son obligatorios llenar ya que son imprescindibles para proporcionar la flexibilidad a la aplicación de gestión de tutelas.

Decidimos realizar una tabla tan grande ya que es aconsejable tener los datos que hacen referencia a una misma funcionalidad en una misma tabla en vez de tener que realizar comunicaciones entre tablas para obtener una misma funcionalidad.

Respecto a que hay muchas filas, cada una corresponde a un permiso concreto de la web de tutela jurídica, pudiéndolo haberlo realizado por medio de una única variable genérica pero otorgaba menor seguridad y mayores complicaciones en la lógica de otorgar valores para que las distintas secciones sean visibles o no.

1	id_web 	int(11)
2	nombre_web	varchar(40)
3	permisos_expedientes	varchar(3)
4	permisos_administracion	varchar(3)
5	permisos_saos	varchar(3)
6	permisos_social_personal	varchar(3)
7	permisos_social_medico	varchar(3)
8	permisos_social_contactos	varchar(3)
9	permisos_social_prestaciones	varchar(3)
10	permisos_social_enseres	varchar(3)
11	permisos_social_urgencias	varchar(3)
12	permisos_social_seguimiento	varchar(3)
13	permisos_economica_cuentas	varchar(3)
14	permisos_economica_ingresos	varchar(3)
15	permisos_economica_seguros	varchar(3)
16	permisos_economica_obligaciones	varchar(3)
17	permisos_juridica_juzgado	varchar(3)
18	permisos_juridica_fechas	varchar(3)

19	permisos_juridica_retribucion	varchar(3)
20	permisos_juridica_judiciales	varchar(3)
21	permisos_voluntariado	varchar(3)
22	permisos_listado_residencia	varchar(3)
23	permisos_listado_juzgados	varchar(3)
24	permisos_listado_informes	varchar(3)
25	datPersonalPaciente	varchar(3)
26	datResidencialPaciente	varchar(3)
27	datTrasladoPaciente	varchar(3)
28	pacienteMedicos	varchar(3)
29	medicamentos	varchar(3)
30	citas	varchar(3)
31	hospitalizaciones	varchar(3)
32	cirugias	varchar(3)
33	dependencia	varchar(3)
34	acompañamientos	varchar(3)
35	publicaResidencial	varchar(3)
36	prestaciones	varchar(3)
37	teleasistencia	varchar(3)
38	catering	varchar(3)
39	otrasPrestaciones	varchar(3)
40	ropa	varchar(3)
41	objetosValor	varchar(3)
42	pedidos	varchar(3)
43	alquileres	varchar(3)
44	pensiones	varchar(3)
45	expedienteJudgado	varchar(3)
46	judgadoDatos	varchar(3)

Figura 31. Tabla websadministradas de la base de datos.

De todas las filas, la más importante es `id_web`, con la cual diferenciamos una web del resto por lo que es un valor único que no hay dos iguales y se autoincrementa al añadir una nueva web.

El resto de filas no pueden ser nulas siendo todas un valor varchar que sirve para distinguir si se tiene permiso o no para ver las diferentes secciones.

Una vez realizada nuestra web para realizar las tareas de gestión de tutelados necesitamos detallar y explicar de forma técnica como se comunica nuestra web con las webs que tendremos que administrar sus permisos.

5.1.10 COMUNICACIÓN ENTRE WEBS

Teniendo una web externa mediante la cual realizaremos las tareas de flexibilización de la aplicación de la que partíamos, hay que tener en cuenta cómo vamos a realizar la conexión entre ambas webs ya que no tenemos una base de datos única para todo sino que cada web tendrá su propia base de datos.

La comunicación entre las bases de datos de la web gestora de permisos y la aplicación gestora de tutelados es en un sentido.

A partir de la web gestora de permisos no podemos acceder a la base de datos de la aplicación de gestión de tutelados. En el sentido contrario, si hay conexión y esta conexión es la siguiente. Un usuario al acceder a la aplicación de gestión de tutelados genera una sesión PHP con la información necesaria de esta web que es enviada a la base de datos de la web gestora de permisos, donde se comprueba la información que le llega si es correcta devolviendo una serie de variables que contienen la información necesaria para hacer visibles las distintas áreas de la aplicación de gestión de tutelados.

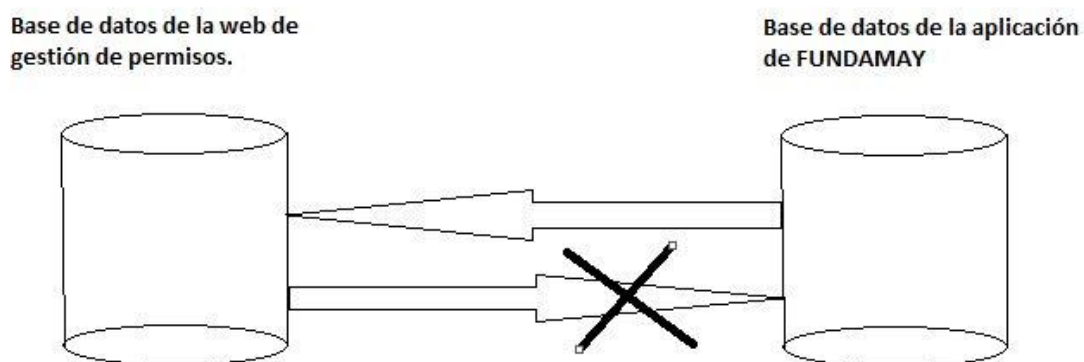


Figura 32. Conexión entre las bases de datos actuales.

El objetivo a largo plazo, es tener muchas bases de datos que conecten con nuestra web de gestión de permisos para que soliciten obtener las variables de permiso que se almacenan.

5.1.11 NUEVA WEB DE GESTIÓN DE TUTELAS

Ya creada la web que se encarga de dar los permisos de la web procedemos a mostrar cómo quedaría la aplicación de gestión de tutelas jurídica una vez flexible ya que tiene distintos niveles de permisos que explicaremos todos. Para poder ver perfectamente que se está realizando correctamente esta flexibilización por medio de permisos, entraremos en la web de tutela jurídica por medio del usuario denominado director que posee todos los permisos para acceder a todas las secciones.

Los permisos de la web quedan distribuidos en tres niveles que hacen referencia:

-Menú principal: haciendo visible o no a las secciones principales de la web que son: área social, económica, jurídica, voluntariado, listados, expedientes, administración y SAOS. Dependiendo de si son visibles o no estas áreas provoca que sus sub-secciones sean visibles o no.

-Submenú: hace visible o no los submenús que se poseen dentro de los menús principales.

-Datos del paciente: los permisos de esta zona nos referimos a los datos que se muestran de cada paciente cuando hay un paciente seleccionado.

A continuación se ve la diferencia entre la misma web con distintos permisos. En las siguientes imágenes veremos cómo quedan las webs con todos los permisos de visión o solo con una serie de permisos.



Es necesario seleccionar un tutelado para acceder a esta sección.

Figura 33. Web con todos los permisos



Es necesario seleccionar un tutelado para acceder a esta sección.

Figura 34. Web sin permisos de primer nivel Voluntariado y Listados.



Figura 35. Web sin todos los permisos de primer nivel (II).

Como podemos observar cuando otorgamos o no permisos al menú principal, los submenús desaparecen aunque el usuario registrado si tenga permisos para verlos. También se observara que las regiones que hacen referencia a los pacientes que se tutelas se ajustan al lado izquierdo sin dejar una zona vacía y en el caso de ser las secciones que son de expedientes, SAOS o administrador se desplaza hacia la derecha para que esos espacios vacíos no queden visibles y que la apariencia de la web sea bonita.

A continuación procedemos a ver otros dos ejemplos de webs que poseen todos los permisos de segundo nivel o de submenús y otras que no lo poseen que ocurre exactamente lo mismo en cuanto a desplazamientos para que no se vean espacios en blanco y el aspecto estético siga siendo amigable.



Figura 36. Web con todos los permisos de segundo nivel del área social.



Figura 37. Web sin los permisos médicos y seguimientos los permisos de segundo nivel del área social.

Por último, hemos de seleccionar a un paciente del que se tengan datos e informes para poder ver el último nivel de permisos. Estos permisos hacen referencia a datos del paciente tutelado seleccionado en ese mismo momento.

Con las imágenes mostraremos un ejemplo en el que se tienen todos los permisos del paciente y en el segundo que falte alguna sección para ver que sigue desplazándose las diferentes regiones para que la web siga siendo bonita.

Nombre	Rubén	Apellidos	Merino Alonso	DNI		Núm. expediente	1235/2010
Estado	Aceptado	Ref. tutelar	Susana Marbán Marbán	Fig. guarda	Curatela	Fecha de entrada	08-04-2014

! Este tutelado tiene mas de un expediente abierto. Utilice el buscador avanzado para encontrar todos los expedientes.

Datos residenciales

[Editar](#)

Vivienda

Capital o provincia Provincia

Municipio Dirección CP

Trasladados de residencia

[Editar](#)

Fecha de traslado 09-04-2014

Residencia anterior Soria - Residencia Azul

Observaciones Prefería vivir en una residencia.

© 2016 GESTUTELAR

Figura 38. Web con todos los permisos de tercer nivel o de datos del paciente.

FUNDAMAY GESTUTELAR

TUTELADO RUBÉN MERINO ALONSO

Usuario: 66666666F | Permisos: Director [Cerrar sesión](#)

Área Social | Área Económica | Área Jurídica | Voluntariado | Listados | Expedientes | Administrador | SAOS

DATOS PERSONALES | DATOS MÉDICOS | CONTACTOS | PRESTACIONES SOCIALES | ENSERES PERSONALES | URGENCIAS | SEGUIMIENTOS

Nombre	Rubén	Apellidos	Merino Alonso	DNI		Núm. expediente	1235/2010
Estado	Aceptado	Ref. tutelar	Susana Marbán Marbán	Fig. guarda	Curatela	Fecha de entrada	08-04-2014

! Este tutelado tiene mas de un expediente abierto. Utilice el buscador avanzado para encontrar todos los expedientes.

Datos residenciales

[Editar](#)

Vivienda

Capital o provincia Provincia

Municipio Dirección CP

© 2016 GESTUTELAR

Figura 39. Web sin los permisos de mostrar datos del traslado de residencias.

Como se ha visto en las distintas imágenes se ha conseguido tener flexible 100% la web de gestión de tutelas por medio de una web externa encargada de

administrar estos permisos. El primer objetivo del proyecto se observa en estas imágenes que se lleva a cabo de forma correcta.

5.1.12 CONCLUSIÓN

Una vez visto el punto 5.1 del documento podemos observar cómo se ha realizado la web externa que se proponía para gestionar los permisos y otorgar esa flexibilidad necesaria a la web.

A lo largo de todo el desarrollo hubo una serie de problemas que se fueron solucionando por medio de búsqueda de información.

El primer problema que apareció fue a la hora de que un usuario se encuentra gestionando los permisos de una web concreta. Si estamos gestionando permisos estamos modificando nuestra base de datos que contiene la información de los permisos de las distintas webs, per un usuario que se encuentre logueado en la aplicación de tutelas, no se sabía cuándo iba a apreciar dichos cambios de permisos.

Para este problema se nos ocurrieron dos soluciones:

- Cuando un usuario se encuentre modificando permisos de la web de tutela jurídica, la propia web de tutela jurídica quedaría desconectada en ese momento.
- Las sesiones que se crean al iniciar sesión en la web de gestión de tutelas hacerlas de un tiempo finito, esto quiere decir que en un tiempo determinado se cerrara la sesión.

Se decide realizar las sesiones de tiempo finito ya que es el método más sencillo y no hay que comunicarse directamente con la web de gestión de tutelas poniendo en peligro los datos de los pacientes tutelados.

El segundo de los problemas, fue la decisión de cómo crear la base de datos y la lógica de permisos para almacenar y hacer visibles las distintas regiones de la aplicación de tutela jurídica.

Comenzamos con la lógica de permisos, en la cual nos decidimos por elegir la más sencilla, utilizar variables con lógica booleana, es decir, un rango corresponde a un TRUE y el rango que se queda fuera corresponde a FALSE. Podíamos haber realizado rangos concretos (1-5 sea visible área social, económica pero no el resto, 6-8 visible solo área económica) lo cual era muy complejo y necesitábamos de mucho tiempo para llegar a unos rangos que funcionaran correctamente.

En cuanto a la distribución de los permisos en una única tabla en la base de datos o dividirla en distintas tablas según secciones se opta por elegir una única tabla, de gran extensión para minimizar el número de tablas de pocos campos.

5.2 AMPLIACIÓN DE LA WEB DE FUNDAMAY

La segunda parte del proyecto consistía en ampliar la web de gestión de tutelas que existía, añadiendo la posibilidad de que los usuarios pudieran modificar la apariencia de la web.

Para esta parte del proyecto necesitamos buscar mucha información para ver como deseábamos realizarlo y llegamos a la decisión de crear hojas de estilo por medio de PHP, en vez de utilizar las hojas de estilo CSS.

Por último, queda añadir la funcionalidad en la sección de administración de la web de FUNDAMAY para que el director sea capaz de modificar la apariencia de la web a su antojo. Hay que destacar que también se ha añadido permisos para que esta sección sea flexible, es decir, que sea visible o no según lo que contrate cada web y que no es visible por todos los usuarios de la web de FUNDAMAY.

5.2.1 HOJAS DE ESTILO CON EXTENSIÓN PHP

Las hojas de estilo con extensión PHP, también llamadas hojas de estilos dinámicas, son hojas de estilos que cambian cuando en el lado del cliente se realiza un cambio en alguna de sus propiedades.

Para crear estas nuevas hojas de estilo lo primero de todo era cambiar la extensión de las hojas de estilo que teníamos en la aplicación que eran .css y ahora pasan a ser .php.

Al cambiar la extensión lo que conseguimos es poder introducir variables PHP dentro de las hojas de estilo con las cuales manejamos colores para modificar la apariencia de las distintas regiones de la web. Así conseguimos modificar la apariencia de la web.

Comenzamos hablando de estas nuevas hojas de estilo con extensión .php en las cuales es necesario añadir una cabecera para que estas hojas de estilo creadas en PHP funcionen correctamente.

```
<?php
    header("Content-type: text/css");
?>
*.html {
```


Figura 37. Cabecera necesaria en las hojas de estilo .php .

Al utilizar esta cabecera y esta nueva extensión, aparecen las conocidas como hojas de estilo dinámicas en las que utilizamos variables PHP para poder modificar los valores en distintas áreas. Hay que destacar que las hojas de estilo dinámicas o con extensión PHP, son compatibles con todos los navegadores a diferencia de todas las funcionalidades de CSS.

Estas variables de PHP en las hojas de estilo se utilizan por medio de la sentencia `<?=$variable?>`.

En la imagen número 38 podemos ver un fragmento de código en el que cargamos de la base de datos la configuración de colores que se tiene inicialmente en la web y como se utiliza estas variables PHP en las distintas regiones de la hoja de estilos.

```
,
div#logo h1 a, div#logo h1 a:visited, div#logo h1 a:hover {
    background: none repeat scroll 0 0 <?=$fondoLogo?>;
    font-size: 22px;
    height: auto;
    line-height: 22px;
    padding: 10px 10px 7px;
    width: auto;
    color: <?=$letrasLogo?>;
    display: block;
    float: left;
    font-weight: normal;
    text-decoration: none;
}
div#buscador_box {
    float:right;
    color: <?=$colorBuscadorLetras?>;
    font-size:14px;
    text-align:right;
    padding-top:8px;
}
div#buscador_box a {
    position: relative;
    top: 4px;
    text-decoration: none;
}
}
```

Figura 41. Fragmento de código de una hoja de estilos con extensión .php.

5.2.2 NUEVA FUNCIONALIDAD EN LA APLICACIÓN DE GESTIÓN DE TUTELAS

La nueva funcionalidad que vamos a proporcionar por medio de las hojas de estilo con extensión PHP será la capacidad de modificar la apariencia de la web de gestión de tutelas jurídicas.

Comenzamos hablando de la distribución de los archivos en distintos directorios que posee la aplicación de tutela jurídica por medio de una imagen para posteriormente situar nuestros nuevos archivos de código donde lo desarrollaremos.

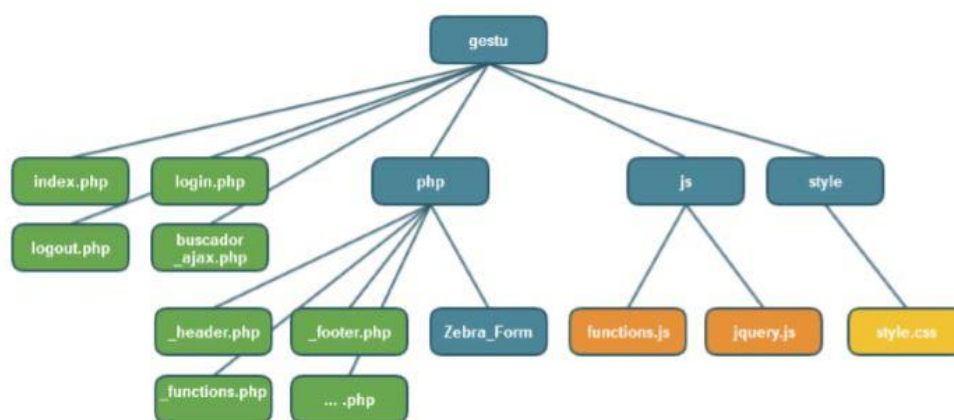


Figura 42. Distribución de los archivos y directorios.

En el directorio raíz se encuentran los siguientes elementos:

-index.php: es el archivo más importante de la plataforma. El acceso a todas las secciones se realiza a través de este archivo. También se encarga de iniciar la sesión PHP, establecer la conexión con la base de datos y comprobar los permisos al intentar acceder a las secciones.

-login.php: se encarga de comprobar las credenciales del usuario y permite o deniega el acceso a la plataforma.

-logout.php: cierra la sesión del usuario y destruye la sesión PHP.

-buscador_ajax.php: realiza una consulta en la base de datos y devuelve una lista de los tutelados que coinciden con los criterios de búsqueda. Este fichero es llamado de forma asíncrona utilizando AJAX.

- Directorio php: este directorio contiene los archivos con el código fuente PHP que implementan las distintas funcionalidades de la plataforma. También contiene los

archivos del framework Zebra Form. Más adelante de detallan los contenidos del mismo.

-Directorio js: este directorio contiene los archivos con el código fuente JavaScript que implementan las distintas funcionalidades de la plataforma.

-El archivo functions.js contiene todas las funciones JavaScript que se utilizan para realizar peticiones con AJAX, para comprobar formularios, para mostrar diálogos de confirmación, etc.

-El archivo jquery.js se corresponde con la última versión del framework jQuery. Es utilizado tanto por functions.js como por el framework Zebra Form.

-Directorio style: este directorio contiene el archivo style.css con el código fuente CSS y las imágenes utilizadas en la plataforma.

Dentro del directorio php se encuentran los siguientes elementos:

- _header.php: archivo que contiene la cabecera de la plataforma.
- _footer.php: archivo que contiene el pie de página de la plataforma.
- _functions.php: contiene todas las funciones desarrolladas que complementan o añaden funcionalidades de la plataforma.
- adm_*.php: contiene la sección de administración.
- eco_*.php: contiene la sección de área económica.
- exp_*.php: contiene la sección de expedientes.
- jur_*.php: contiene la sección de área jurídica.
- saos_*.php: contiene la sección SAOS.
- soc_*.php: contiene la sección de área social.
- vol_*.php: contiene la sección de voluntariado.
- Zebra_Form: contiene los ficheros del framework Zebra Form, empleado para el desarrollo de los formularios.

Una vez vista la distribución de los archivos, decidimos añadir los nuevos para desarrollar la nueva funcionalidad dentro del directorio php y tendrán la extensión adm_configuracion_*.php.

Las nueva hoja de estilo, estarán en el directorio style y se llamara style.php.

5.2.3 DISEÑO DE LA WEB

El primer paso consiste en añadir la sección para que podamos modificar la apariencia de la web. Para ello tenemos que decidir si todos los usuarios pueden modificar la apariencia o no y se decide utilizar únicamente esta funcionalidad si se ha

registrado el usuario con los permisos de director. Por lo tanto se situara dentro de la sección de administración el apartado de configuración de web.



Figura 43. Nuevo apartado de configuración web.

Una vez añadido el menú para que aparezca la opción para realizar la configuración de la apariencia de la web, procedemos a crear esta vista para que al usuario le permita, de forma sencilla, modificar la apariencia de la propia web.

Para esto utilizaremos una web en la que añadimos botones que al pulsarlos aparezca una paleta de colores para indicar el color deseado para las distintas secciones.

Estas paletas de colores que aparecen se realizan por medio de HTML con una funcionalidad que posee los botones y es con un `input="color"`. Pero este método no es compatible con todos los navegadores como se pudo ver en el estudio previo de tecnologías por lo que decidimos realizar dos métodos para cambiar la apariencia según con el navegador que estemos lanzando nuestra web.

Por lo tanto, para acceder a un método u otro de modificar el diseño, se necesita utilizar algún método para distinguir el navegador que se está utilizando y se realiza por medio de JavaScript y PHP para asegurarnos que podremos modificar la apariencia de la web en cualquier navegador existente.

5.2.3.1 DISEÑO POR MEDIO DE INPUT="COLOR"

Es compatible solo con una serie de navegadores entre los cuales se encuentran google Chrome, Mozilla Firefox y Opera. La pestaña que permite modificar la apariencia de la web se muestra en la imagen 44.

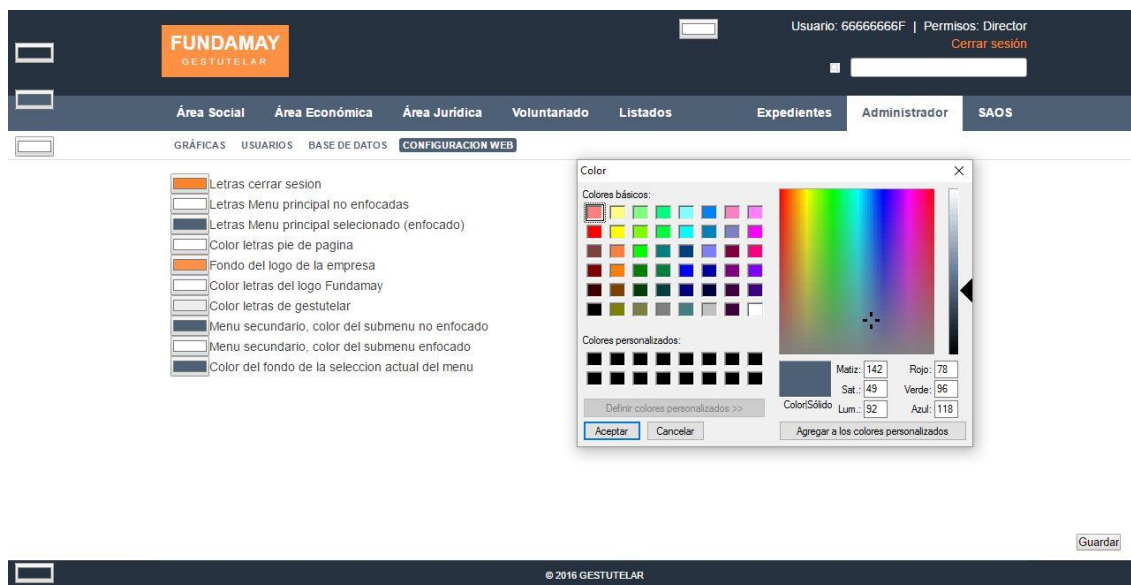


Figura 44. Configuración apariencia de la web desde Chrome.

Vemos en la imagen que se añaden distintos botones que al pulsarlos aparece una paleta de colores en la que seleccionamos el color deseado. En la zona central de la web, aparecen más botones detallando a qué secciones hace referencia para modificar su apariencia.

Una vez que se haya concluido la configuración del diseño, se necesita pulsar el botón de guardar para almacenar la configuración del diseño de la web. Estos datos se mandan a la base de datos a una nueva tabla que se haya creado, que posteriormente hablemos de ella, para almacenarlo y que se carguen la próxima vez que accedamos a la web.

5.2.3.2 ESTILOS YA CREADOS CON ANTERIORIDAD

En navegadores que no soportan el método explicado en el apartado anterior necesitamos crear estilos con anterioridad y que el usuario seleccione uno u otro para modificar la apariencia de la web.

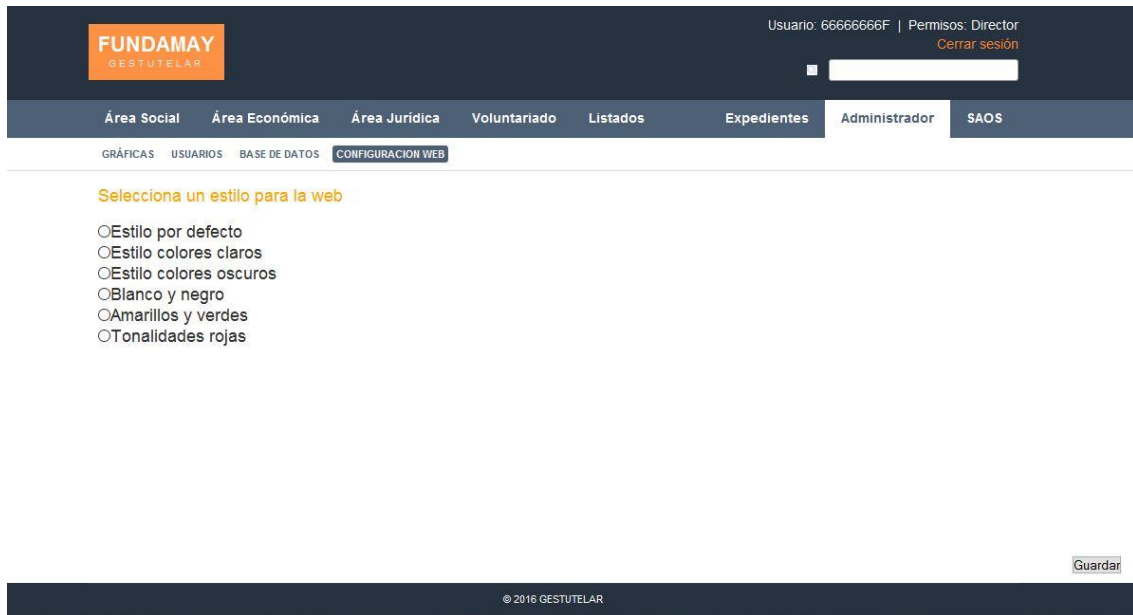


Figura 45. Configuración apariencia de la web desde Safari navegador.

Inicialmente se han creado 6 estilos de colores. A continuación mostraremos estos seis estilos según como quedaría la web.

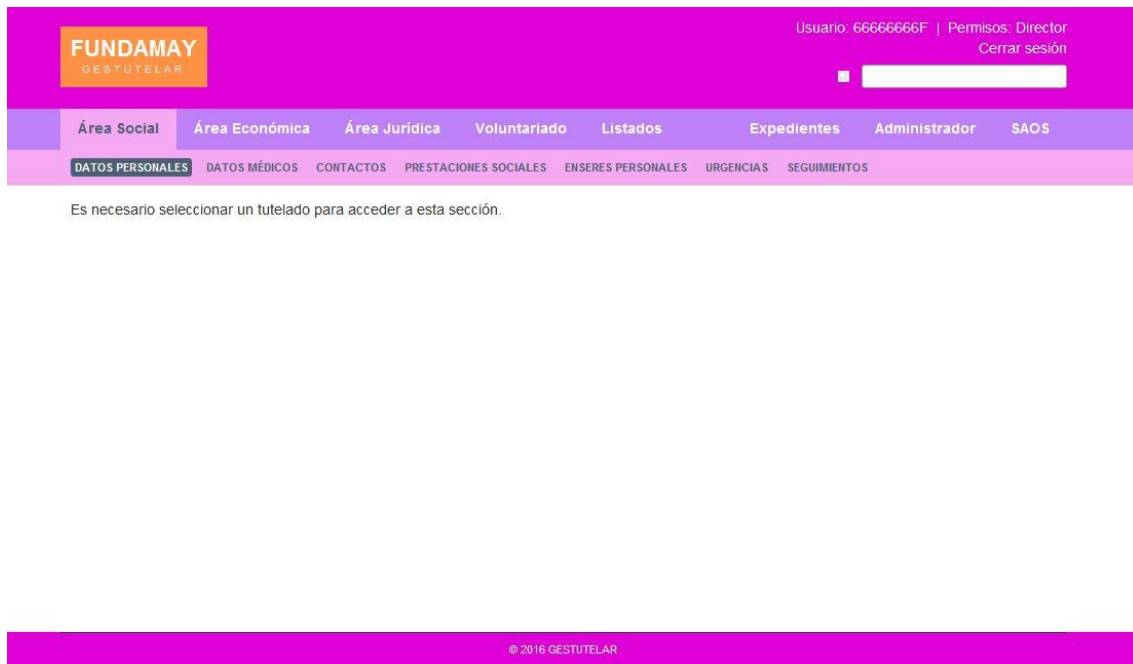


Figura 46. Estilo de colores claros.



Figura 47. Estilo de colores oscuros.



Figura 48. Estilo de colores blancos y negros.

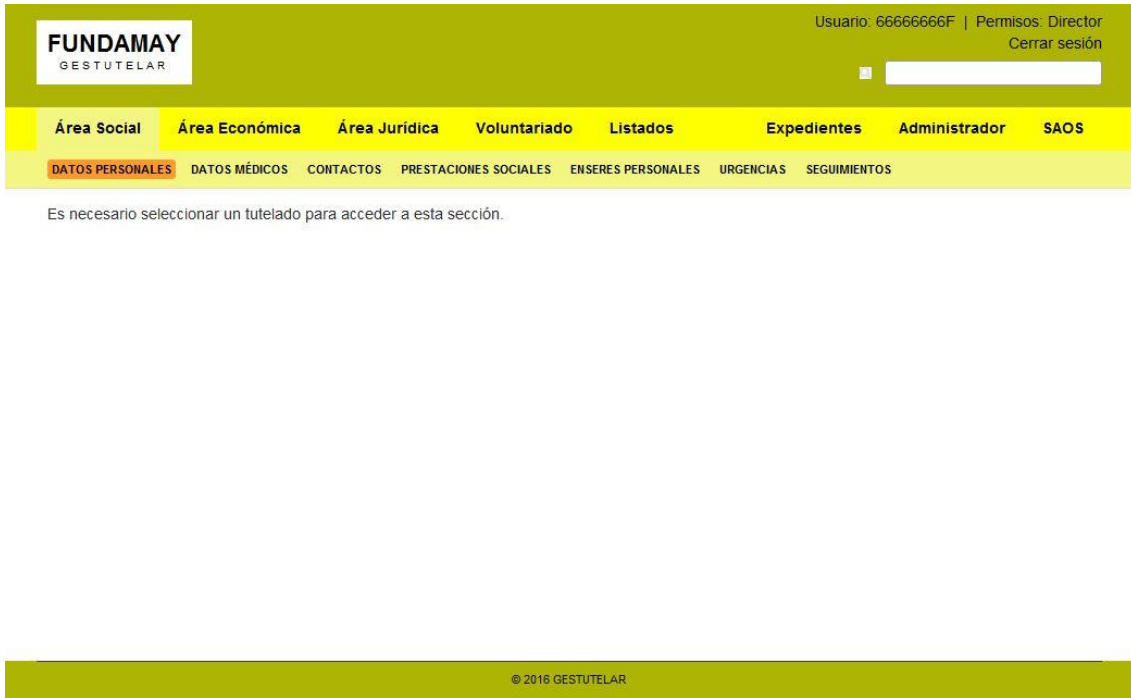


Figura 48. Estilo de colores amarillos y verdes.

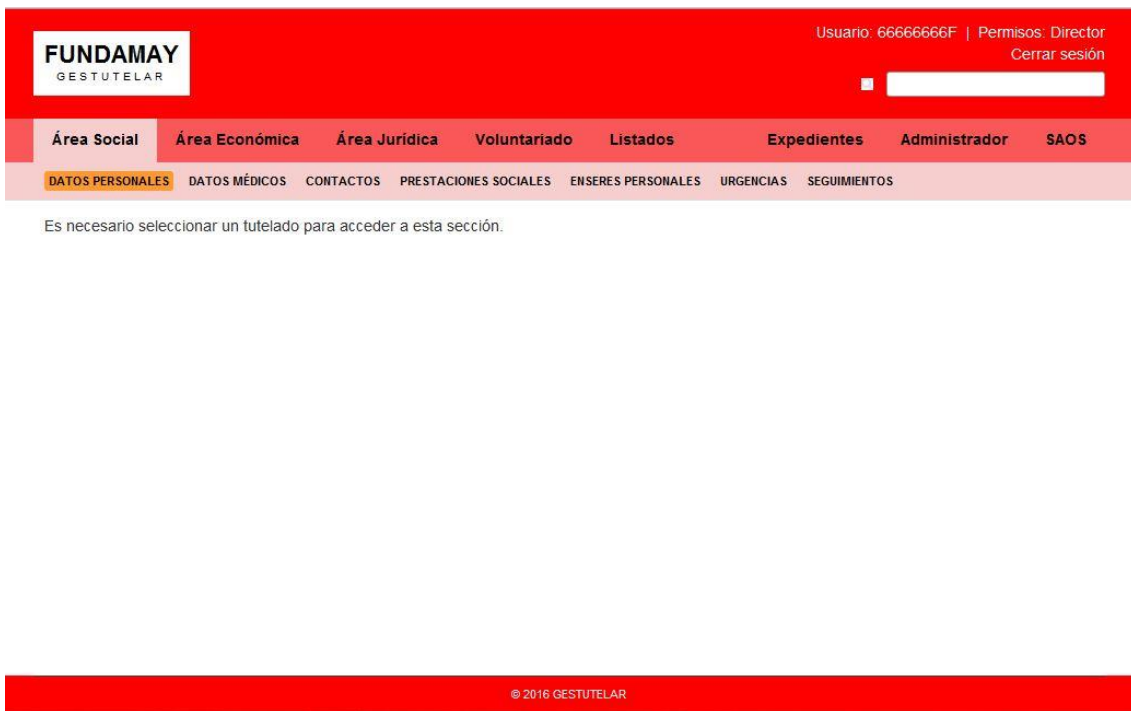


Figura 49. Estilo de colores rojos.

5.2.4 CASOS DE USO DE LA NUEVA FUNCIONALIDAD

Para describir la funcionalidad utilizamos de nuevo los diagramas de casos de USO.

Dado que existen nueve tipos de usuarios distintos en la plataforma y muchos de ellos son comparables en términos de permisos, se han simplificado los diagramas de casos de uso de forma que su comprensión sea más sencilla. En la imagen 47 se pueden ver los usuarios reales existentes en el sistema al lado izquierdo y el usuario virtual que los representa en los diagramas.
















Actores del sistema	Actores simplificados
  Voluntario Auxiliar	 Voluntario
   Referente social Referente económico Referente jurídico	   Referente social Referente económico Referente jurídico
   Coordinador social Coordinador económico Coordinador jurídico	 Coordinador
 Director	 Director

Figura 50. Actores del sistema.

En la sección de administrador, donde situamos la nueva funcionalidad, ni los referentes ni el voluntario va a tener acceso a esa sección. El coordinador va a poder generar las gráficas y el director va a poder acceder a la sección completa de administrador, por lo tanto a la nueva región que permite la nueva funcionalidad de la web solo será accesible si se tienen permisos de director.

5.2.5 BASE DE DATOS

Con la nueva funcionalidad creada en la aplicación de gestión de tutelas, es necesario crear una tabla nueva en la base de datos ya que se necesita guardar las distintas variables necesarias para cargar los estilos de las hojas de estilos PHP.

Para ello en la base de datos existente de la aplicación de gestión de tutela jurídica, donde la base de datos se llama gestu, se añade una nueva tabla en la que se almacena estas variables.

A continuación procedemos a mostrar la estructura de la tabla que utilizaremos para almacenar las variables de control de estilos de la web.


1	id_web 	int(11)
2	colorHeader	varchar(7) latin1_swedish_ci
3	colorNav	varchar(7) latin1_swedish_ci
4	colorLetrasNavNopulsadas	varchar(7) latin1_swedish_ci
5	colorLetrasNavPulsadas	varchar(7) latin1_swedish_ci
6	colorSubNav	varchar(7) latin1_swedish_ci
7	colorFooter	varchar(7) latin1_swedish_ci
8	colorFooterLetras	varchar(7) latin1_swedish_ci
9	colorBuscadorLetras	varchar(7) latin1_swedish_ci
10	colorCajaBuscador	varchar(7) latin1_swedish_ci
11	colorWeb	varchar(7) latin1_swedish_ci
12	colorHipervinculos	varchar(7) latin1_swedish_ci
13	fondoLogo	varchar(7) latin1_swedish_ci
14	letrasLogo	varchar(7) latin1_swedish_ci
15	colorGestutelar	varchar(7) latin1_swedish_ci
16	colorLetrasSubNavNoPulsadas	varchar(7) latin1_swedish_ci
17	colorLetrasSubNavPulsadas	varchar(7) latin1_swedish_ci
18	colorCuadroSubNavPulsadas	varchar(7) latin1_swedish_ci

Figura 51. Tabla de estilos web.

La nueva tabla está formada por 18 columnas que corresponde cada una a una región de la web para modificar su estilo.

La `primary_key` o clave principal de la tabla corresponde con el identificador web, único de la web que se envía para comprobar que esta tabla corresponde a la web.

El resto de filas de la tabla:

-colorHeader: fila que almacena un código hexadecimal que otorga el color de la cabecera de la web de tutela jurídica.

-colorNav: almacena un código hexadecimal en una variable varchar. Corresponde al color del navegador principal (menú principal de la web) de la web.

-colorLetrasNavNoPulsadas y colorLetrasNavPulsadas: estas dos filas almacenan al igual que las anteriores, un código hexadecimal que controla el color de las letras del navegador principal de la web (menú principal).

-colorSubnav: fila para almacenar un código hexadecimal para el control del color de la región del submenú de la web.

-colorFooter y colorFooterLetras: ambas filas sirven para almacenar los códigos hexadecimales que otorgan el color del pie de página y de las letras que corresponden al copyright y año de desarrollo de la web.

-colorCajaBuscador: fila para almacenar el color de la caja de buscador, aunque para poder cambiar el color es necesario de una lógica compleja en la web ya que deseamos que sea visible siempre y se añaden una serie de restricciones a la hora de añadir un color determinado.

-colorWeb: fila que hace referencia al color del cuerpo de la web.

-colorHipervinculos: almacenamos el color que otorgara el estilo a todos los hipervínculos de la web.

-fondoLogo y letrasLogo: filas para almacenar el estilo que corresponde al fondo y las letras de la web. En nuestro caso es el logo cuyo nombre es FUNDAMAY, podemos modificar los colores tanto de su fondo como de sus letras.

-colorGestutelar: fila en la que se almacena el estilo correspondiente a las letras que aparecen en el fondo del logo y que aparece gestutelar como texto.

-colorLetrasSubNavNoPulsadas, colorLetrasSubNavPulsadas, colorCuadroSubNavPulsadas: estas tres filas corresponden al submenú de la web. Modifican la apariencia de las letras del submenú indicando con distintos colores en que submenú nos encontramos y en cuáles no.

Una vez detalladas cada una de las filas es recomendable visitar la sección de manual de usuario para detallar el funcionamiento de la edición del estilo de la web donde se detallara de forma más exacta como podemos modificarla por medio de esta tabla.

5.2.6 CONCLUSIONES DE LA AMPLIACIÓN

Al realizar esta ampliación conseguimos que el propio director de la aplicación pueda modificar el estilo de la web para hacerla más amena a la vista o que utilice el código de colores deseado por la empresa. Con este método podemos otorgar una nueva funcionalidad que actualmente la mayoría de las webs posee y que es de gran ayuda ya que cada persona desea tener una apariencia diferente a la hora de ver la web.

Hay que destacar, que solo se le permite al director de la fundación realizar este cambio, tal vez, en una futura versión se pueda permitir al resto de los usuarios que no tengan el cargo de director la posibilidad de modificar el estilo de la web.

CAPÍTULO 6

Manual de usuario

En el próximo capítulo, el número seis, va destinado a crear una pequeña guía de usuario para las personas que se van a encargar de manejar la nueva funcionalidad que hemos creado en la web de FUNDAMAY, la modificación del estilo de la web.

Esta guía de usuario también contiene detallada la funcionalidad y el manejo de la nueva web que se ha creado para administrar los permisos de la aplicación de tutela jurídica.

6.1 INTRODUCCIÓN

Comenzamos explicando la estructura de la web existente de gestión de tutelas jurídicas ya que está formada por diferentes menús de navegación y posteriormente nos centraremos en la nueva funcionalidad que hemos desarrollado.

6.1.1 SECCIONES DE LA PLATAFORMA

La plataforma de la que partíamos estaba constituida por distintas secciones que seguían el esquema de trabajo de la fundación FUNDAMAY. Cada una de estas secciones corresponde a una sección de tutela. Las distintas secciones son las siguientes:

-Área social: contiene los datos personales y médicos, los contactos, las prestaciones sociales, los enseres personales, las urgencias y los seguimientos realizados a un tutelado.

-Área económica: contiene las cuentas bancarias, los ingresos, los seguros y las obligaciones generadas para un tutelado.

-Área jurídica: en esta sección se encuentran los datos del juzgado asignado a un tutelado, las fechas relevantes (de entrada, de aceptación en Fundamay y de baja, entra otras), la retribución y las autorizaciones judiciales de un tutelado.

-Voluntariado: permite añadir un nuevo informe de voluntario sobre un tutelado y, en función de los permisos del usuario, ver los informes de voluntario generados.

-Expedientes: contiene el buscador básico y avanzado y permite generar un nuevo expediente.

-SAOS: permite añadir un nuevo informe de atención de tipo SAOS y ver los informes previamente generados.

-Administrador: contiene la lista de residencias, de juzgados y de usuarios y permite generar listados de cualquier dato de la plataforma en formato de tabla de Excel así como realizar copias de seguridad de la base de datos.

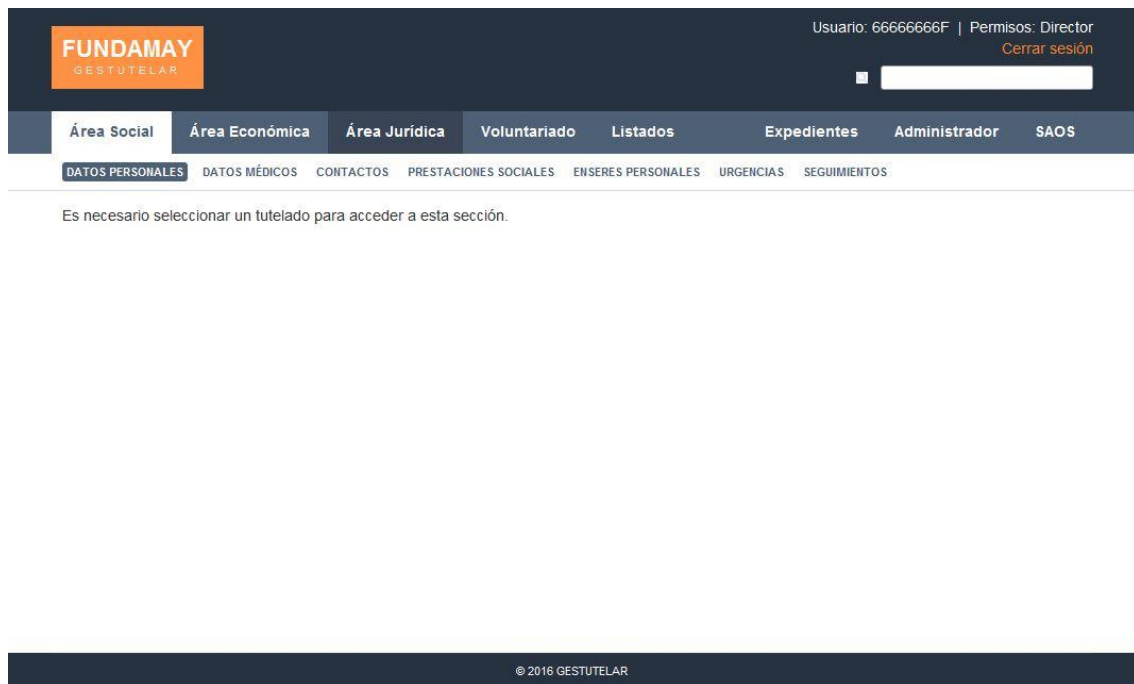


Figura 52. Web de tutela inicialmente.

Nos centraremos en la sección de administración que es donde se sitúa la nueva funcionalidad que hemos desarrollado para la web.

6.2 NUEVA FUNCIONALIDAD EN ADMINISTRACIÓN

En esta sección de administración se puede encontrar el menú característico del director del sistema donde solo él puede acceder y donde situamos nuestra nueva funcionalidad, la configuración del estilo de la web.

En la sección de administración se tienen cuatro submenús que son únicos para el director.



Figura 53. Menú del director.

La nueva funcionalidad añadida a la web es el submenú: configuración web, el cual explicaremos a continuación como manejarlo de forma sencilla.

6.2.1 CONFIGURACIÓN WEB

Submenú característico del director ya que solo el posee los permisos necesarios para acceder a este menú.

Para acceder a este submenú hay que loguearse inicialmente en la web gestu, con el usuario 66666666F con el password 123456, así accedemos con un usuario con los permisos de director.

Una vez logueado accedemos a una vista similar a la que se ve en la figura 49. Hemos de pulsar en Administrador para abrir su submenú y aparecen cuatro submenús donde el importante el llamado configuración web.

De nuevo se pincha en configuración web para acceder a la vista desarrollada. La apariencia que nos aparecerá cuando nos encontremos dentro de este submenú variara dependiendo en el navegador que nos encontremos lanzando la aplicación teniendo dos vistas completamente distintas y con unas funcionalidades diferentes.

6.2.1.1 NAVEGADORES QUE SOPORTAN INPUT="COLOR"

Cuando accedemos a la aplicación de tutelas jurídicas a través de un navegador que soporta el atributo INPUT="color", la pantalla que se mostrará será la siguiente que se muestra en la siguiente imagen.

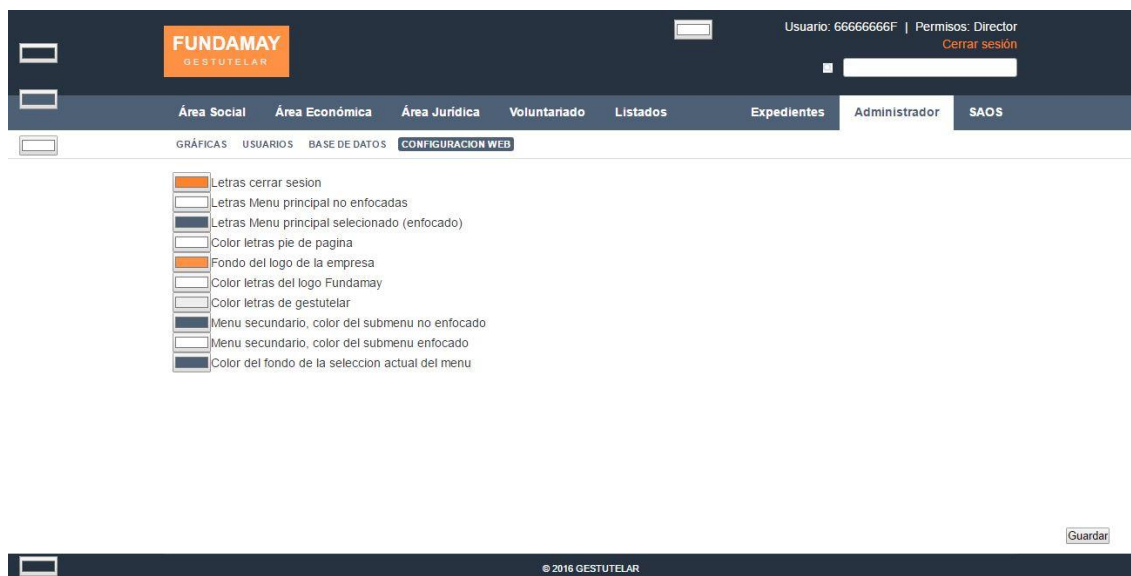


Figura 54. Submenú configurar web en navegador google Chrome.

Se observa que aparecen 15 botones de colores que corresponden a las 15 variables que se pueden modificar en las hojas de estilo PHP para modificar la apariencia de la web.

Podemos distinguir que hay 5 etiquetas del tipo color situadas sin ningún texto explicativo pero están situadas justamente en la zona que van a editar.

Vamos a explicar la función de cada uno de los botones paso a paso indicando con ejemplos la parte de la web que modifican. Se irán enumerando para explicar los 15 botones que hay. Primero explicaremos que todos los botones, al ser pulsados se despliega una paleta de colores que tiene todos los colores existentes. Con esta paleta podemos buscar el color deseado o asignarlo por código hexadecimal.

Una vez elegido el color deseado para un botón concreto solo hace falta pulsar a aceptar en el pop-up emergente de la paleta de colores para asignar ese color al botón correspondiente. Si deseamos cancelar la operación podemos dar al botón cancelar o directamente cerrar la ventana pulsando en la x.

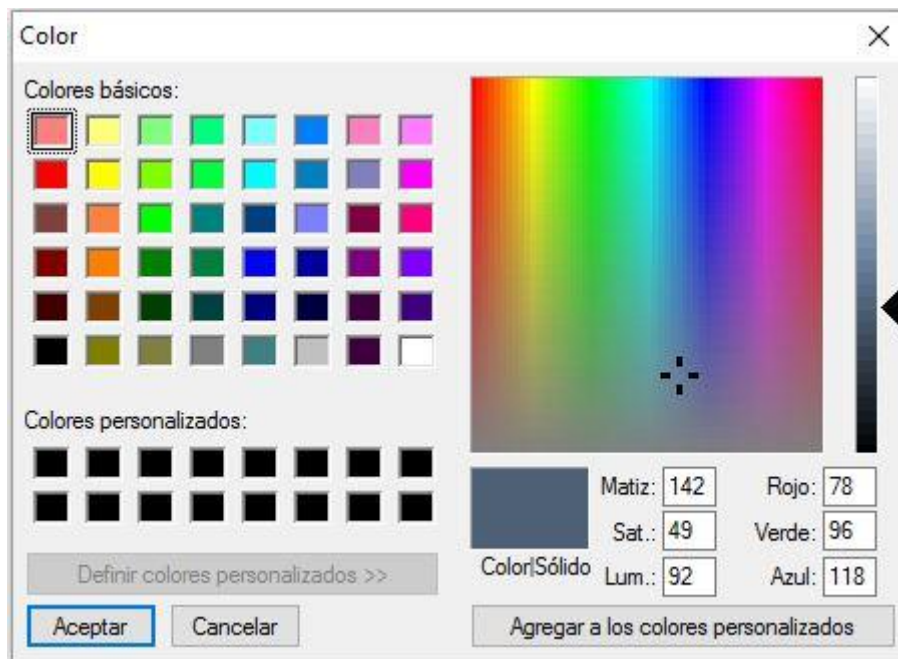


Figura 55. Paleta de colores.

Comenzamos explicando la posición de cada botón y la región que modifica.

1) Este botón se encuentra situado en la parte superior izquierda de la web que se encarga de modificar el color de la cabecera.



Figura 56. Botón de la cabecera.

2) El segundo botón se encuentra justamente debajo del que modifica la cabecera de la web. Este botón se encarga de modificar la apariencia del menú principal.

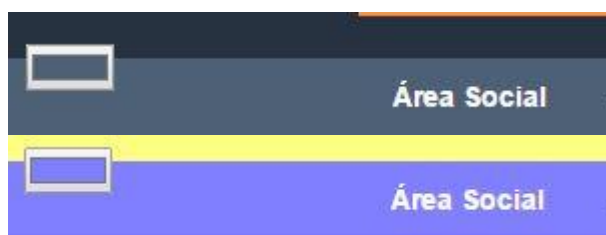


Figura 57. Botón del menú principal.

3) El tercer botón se encuentra justo debajo del botón anterior y se encarga de modificar la apariencia del submenú de la aplicación.



Figura 58. Botón del submenú.

4) este botón se encuentra situado en la parte inferior izquierda de la web. Modifica el color del pie de página.



Figura 59. Botón del pie de página.

5) Situado en la parte superior derecha de la aplicación. Modifica botón se encuentra situado en la parte inferior izquierda de la web. Modifica el color de las letras que indica el rango del usuario que está registrado y su DNI.



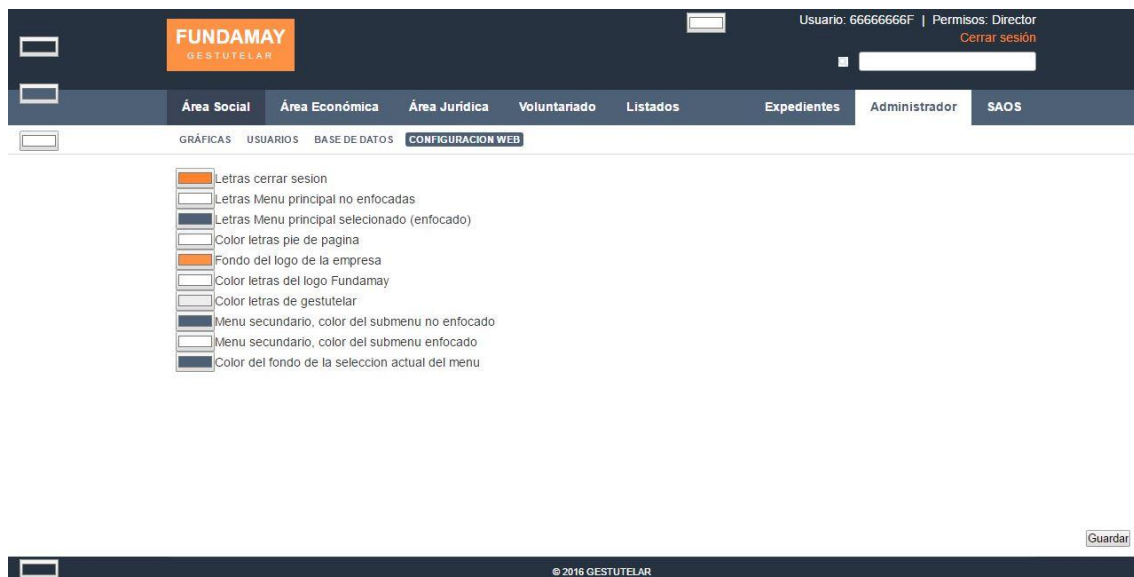
Figura 60. Botón posición superior derecha.

El resto de botones se encuentran situados en la parte central de la web y cada uno tiene un texto indicando lo que modifican en la web por medio de un texto explicativo fácil de entender por los usuarios.

- El primero de los botones de la parte central, modifica el color de las letras cuyo texto es: cerrar sesión.
- El siguiente botón, modifica el color de las letras que se sitúan en el menú principal indicando las diferentes secciones de la web (área social, área económica, SAOS...).

- El tercer botón modifica el color de las letras del menú principal en el que nos encontramos situados.
- El cuarto botón modifica las letras que se sitúan en el pie de página que hacen referencia al copyright y el desarrollador de la web.
- Los siguientes tres botones sirven para modificar el logo de la empresa pudiendo modificar tanto el fondo como las letras que indican FUNDAMAY o gestutelar.
- A continuación de estos tres botones, se encuentran dos botones para modificar el cuadro del submenú que aparece indicando en la sección que nos encontramos. Estas letras del submenú podemos modificar el color de la zona en que nos encontramos situados y en la que no.
- El último de los botones, modifica el color del cuadro que indica en la sección concreta que nos encontramos.

Mostraremos una imagen en la que veremos la apariencia original de la web y una con la modificación de todos los botones para poder ver de forma visual a que área corresponde cada botón. Como se puede observar, los botones contienen el color de la región que modifican para aclararnos que zona van a editar.



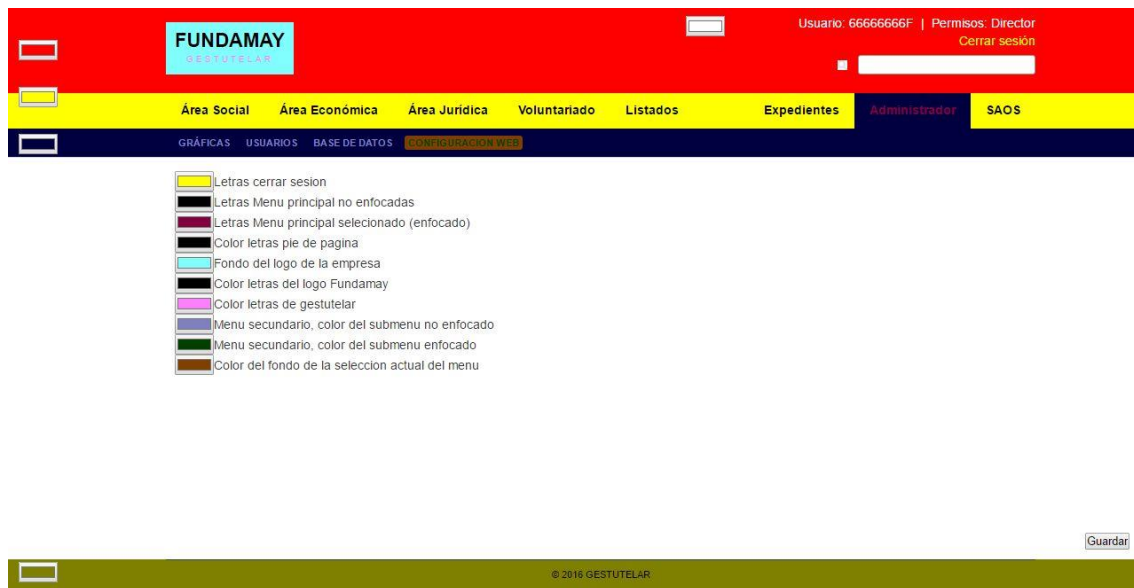


Figura 56. Comparación de distintos estilos.

Una vez que hemos modificado la apariencia de la web, para que los cambios surtan efecto es necesario pulsar el botón aceptar situado en la parte inferior derecha y modificamos por completo la apariencia de la web.

6.2.1.2 NAVEGADORES QUE NO SOPORTAN INPUT="COLOR"

Dentro de los navegadores que no soportan este atributo de HTML podemos encontrar safari o internet explorer en sus versiones más antiguas.

Para poder tener esta nueva funcionalidad de la web en estos navegadores que no lo soportan, hemos creado una apariencia un poco diferente a la anterior, en la cual se han creado un conjunto de estilos por defecto los cuales el director de la aplicación puede seleccionar y aplicar para la apariencia de la web.

Actualmente se han creado 6 estilos que pueden ser seleccionados y aplicados para modificar la apariencia, los cuales podrían ser ampliados en caso de que se necesitaran crear nuevos estilo.

Para aplicar estos nuevos estilos, el director ha de acceder al menú del administrador y a su submenú de configuración de la web. Una vez dentro le aparecerá una pantalla con unos radio-button cada uno con el texto que indica el tipo de estilo que puede aplicar a la web.

Selecciona un estilo para la web

- Estilo por defecto
- Estilo colores claros
- Estilo colores oscuros
- Blanco y negro
- Amarillos y verdes
- Tonalidades rojas

Figura 62. Opciones de estilos.

El usuario selecciona uno de estos estilos y una vez quiera guardarlo ha de pulsar el botón con el texto de guardar situado en la parte inferior derecha.

Los 6 estilos creados para estos navegadores los mostramos en las siguientes imágenes para ver cómo quedaría la web. El orden de los estilos que vemos en la imagen es el siguiente:

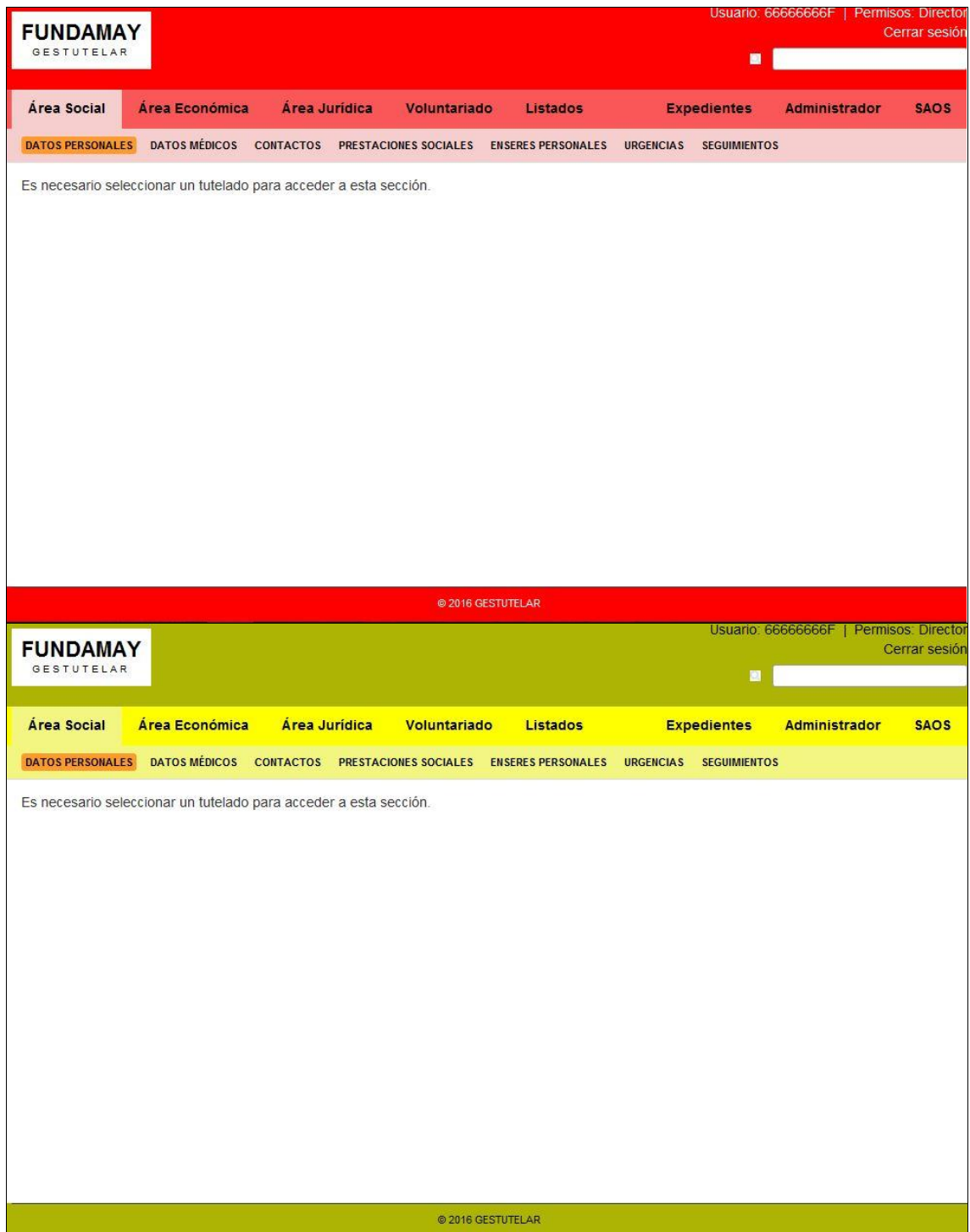
- Estilo por defecto.
- Blancos y negros.
- Colores claros.
- Colores oscuros.
- Tonalidades rojas.
- Amarillos y verdes.

Es necesario seleccionar un tutelado para acceder a esta sección.

Es necesario seleccionar un tutelado para acceder a esta sección.

Es necesario seleccionar un tutelado para acceder a esta sección.

Es necesario seleccionar un tutelado para acceder a esta sección.



En el caso que se desee añadir nuevos estilos, el administrador no lo puede hacer por sí mismo. Para añadir nuevos estilos es necesario contactar con el desarrollador el cual es el encargado de crearlos (se ofrece tareas de soporte).

6.3 WEB DE GESTIÓN DE PERMISOS

En este apartado explicaremos de forma detallada cómo manejarnos por la nueva web de gestión de permisos.

Hay que destacar que lo primer paso para acceder a la web es loguearse en ella. Para ello necesitamos introducir nuestro usuario y contraseña, que se comprueban en la base de datos, y si los datos son correctos accedemos a la página principal de la web.



Mantenimiento

Iniciar sesión

DNI *

Contraseña *

Entrar

Figura 64. Pantalla de login.

Para poder acceder al sistema y realizar las pruebas convenientes, se han registrado dos usuarios en la base de datos cada uno con un permiso distinto.

- DNI: 11111111N con permisos de empleado de mantenimiento.
- DNI: 66666666F con permisos de jefe de mantenimiento.

Ambos usuarios tienen la misma contraseña, 123456.

Una vez logueados en la web se accede a la página principal con las distintas secciones con sus respectivas subsecciones. El usuario con permisos de mantenimiento no puede modificar los datos de un usuario que se encuentra registrado en la base de datos de la web. Las secciones de las que está compuesta nuestra web de gestión de tutelas son las siguientes:

-Mantenimiento web: contiene los datos de las distintas webs a las que se está ofreciendo el servicio de proporcionar los permisos de visión de las distintas áreas de la herramienta de gestión de tutelas.

-Usuarios de mantenimiento: contiene los datos de los distintos usuarios que se encuentran registrado para acceder a esta web y hay un apartado en el que solo se podrá acceder en el caso de estar logueado como jefe de mantenimiento, este apartado permite modificar los datos de un usuario registrado.

6.3.1 CABECERA DE LA WEB

La cabecera de la web es común a todas las páginas de la plataforma, y consta de distintas secciones:

- Menú principal que contiene las secciones principales.
- Submenú o menú secundario que contiene las subsecciones.
- Información del usuario.

La parte superior de la plataforma se muestra en la siguiente figura.

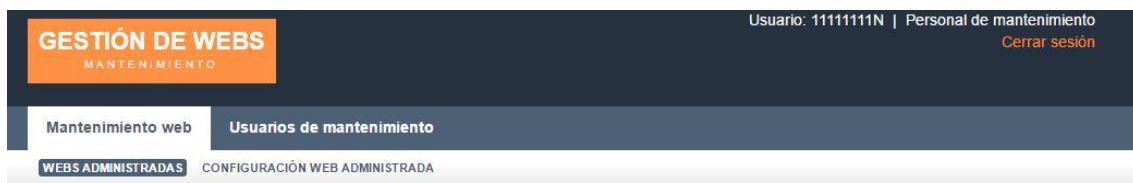


Figura 65. Cabecera de la plataforma.

6.3.2 MENÚ PRINCIPAL Y SUBSECCIONES

Desde el menú de secciones se puede acceder a cada una de las secciones de la plataforma con solo hacer clic sobre la deseada. La sección en la que se encuentra el usuario queda resaltada de color gris claro.



Figura 66. Menú principal.

Desde el menú de subsecciones se puede acceder a las diferentes subsecciones de las que consta una sección con solo hacer clic en la deseada. La subsección en la que se encuentra el usuario queda resaltada de color azul, como muestra la Figura 61.



Figura 67. Menú de subsección.

6.3.3 INFORMACIÓN DEL USUARIO

En esta parte de la cabecera se muestra el DNI del usuario que ha accedido a la plataforma, su nivel de permisos, la opción de cerrar sesión para salir de la plataforma.

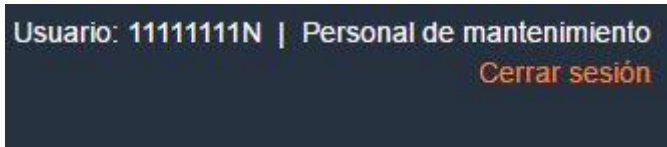


Figura 68. Información del usuario logueado.

6.3.4 SECCIÓN MANTENIMIENTO WEB

En esta sección se puede encontrar toda la información sobre la web a las que estamos proporcionando el servicio de permisos en su web de tutela jurídica. Tenemos dos subsecciones:

-Webs administradas: subsección en la que se muestra el nombre de la web y las distintas áreas de tutela jurídica con los permisos que tienen contratados.

Nombre web	Área social	Área Económica	Voluntariado	SAOS	Listados	Administracion	Expedientes	Área Jurídica
Fundamay	Contratado	Contratado	Contratado	Contratado	Contratado	Contratado	Contratado	Contratado

Figura 69. Subsección de webs Administradas.

-Configuración de web administrada: subsección en la que se muestran las webs que administramos los permisos de visión de las distintas áreas de tutela, con un icono que permite redirigir a una página para administrar los permisos.

Nombre web	Área social	Área Económica	Voluntariado	SAOS	Listados	Administracion	Expedientes	Área Jurídica
Fundamay	Contratado	Contratado	Contratado	Contratado	Contratado	Contratado	Contratado	Contratado 

Figura 70. Subsección configuración web.

Como observamos, se muestra una tabla de datos con la información de una web a la que prestamos el servicio ofrecido por nuestra aplicación. En la tabla se muestra el nombre de la web y las distintas áreas de la herramienta de gestión de tutelas jurídicas con la información de si tienen contratada el área o no. Es posible que se muestren tres valores respecto a cada área:

-Contratado: se tiene contratada por completo toda la sección con sus respectivas subsecciones

-Contratado Parcial: se tiene contratada esa sección pero falta alguna subsección.

- No contratado: indica que esa sección no es contratada por la fundación por lo que no será visible.

En la subsección de webs administradas solo mostramos información acerca de los permisos que contrata cada web. Al pulsar sobre el nombre de una de estas webs accedemos a una página donde se detalla de forma más explícita en una tabla todos los permisos de todas las áreas y subsecciones de la web.

WEBS ADMINISTRADAS		CONFIGURACIÓN WEB ADMINISTRADA				
Nombre Web	Fundamay					
Permisos Área Social						
Datos Personales	Datos Médicos	Contactos	Prestaciones Sociales	Enseres personales	Urgencias	Seguimientos
Contratado	Contratado	Contratado	Contratado	Contratado	Contratado	Contratado
Permisos Área Económica						
Cuentas Bancarias	Ingresos	Seguros	Obligaciones			
Contratado	Contratado	Contratado	Contratado			
Permisos Área Jurídica						
Datos del Juzgado	Fechas	Retribución	Autorizaciones Judiciales			
Contratado	Contratado	Contratado	Contratado			
Permisos Área Voluntariado						
Ver y Crear Informes						
Contratado						
Listados						
Listado de Residencias	Listado de Juzgados	Listado de Informes				
Contratado	Contratado	Contratado				
SAOS						

Figura 71. Información detallada de los permisos de una web.

Respecto a la subsección de configuración web, se muestran las mismas webs que en la otra subsección pero con un icono situado en el lado derecho el cual al pulsarle podemos acceder a otra página donde se muestra un formulario con los datos necesarios para modificar los permisos de visión de las diferentes secciones que posee la aplicación de tutela.

Editar permisos web contratados

[Volver](#)

Nombre web: *

ÁREA SOCIAL

Contactos: *

 si
 no

Enseres: *

 si
 no

Medico: *

 si
 no**Figura 72.** Formulario de edición de permisos.

Al acceder a esta página de edición de permisos se carga un formulario con todas las secciones que se pueden modificar su visión en la web que hemos seleccionado. Si se desea modificar algún permiso de visión, basta con cambiar la situación del radio-button a la opción contraria y pulsar el botón “guardar” que se encuentra al final del formulario.

En caso de no desear cambiar ningún permiso podemos acceder a la sección anterior pulsando el botón “volver”.

En esta sección de permisos hay que detenerse a explicar que se poseen permisos relacionados con mostrar datos concretos del paciente que corresponden a los denominados permisos de tercer nivel. Estos permisos son visibles cuando en la aplicación de tutela se tiene seleccionado una persona tutelada y se accede a un área concreta de esta persona en la cual aparecen una serie de datos. En la siguiente imagen podemos ver ejemplo de cómo podemos modificar los permisos de las distintas áreas y los permisos denominados de tercer nivel.

Nombre	Luis Antonio	Apellidos	García Marqués	DNI		Núm. expediente	1236/2010 (2 exp.)
Estado	En valoración	Ref. tutelar		Fig. guarda	Curatela	Fecha de entrada	04-03-2014

Datos residenciales

							
Vivienda							
Capital o provincia	Capital	Provincia	Valladolid				
Municipio	Valladolid	Dirección	San Millan 14		CP	47012	

Traslados de residencia

							
Fecha de traslado	03-04-2015						
Residencia anterior	-1 -						
Observaciones							

Nombre	Luis Antonio	Apellidos	García Marqués	DNI		Núm. expediente	1236/2010 (2 exp.)
Estado	En valoración	Ref. tutelar		Fig. guarda	Curatela	Fecha de entrada	04-03-2014

Datos residenciales

							
Vivienda							
Capital o provincia	Capital	Provincia	Valladolid				
Municipio	Valladolid	Dirección	San Millan 14		CP	47012	

Traslados de residencia

							
Fecha de traslado	03-04-2015						
Residencia anterior	-1 -						
Observaciones							

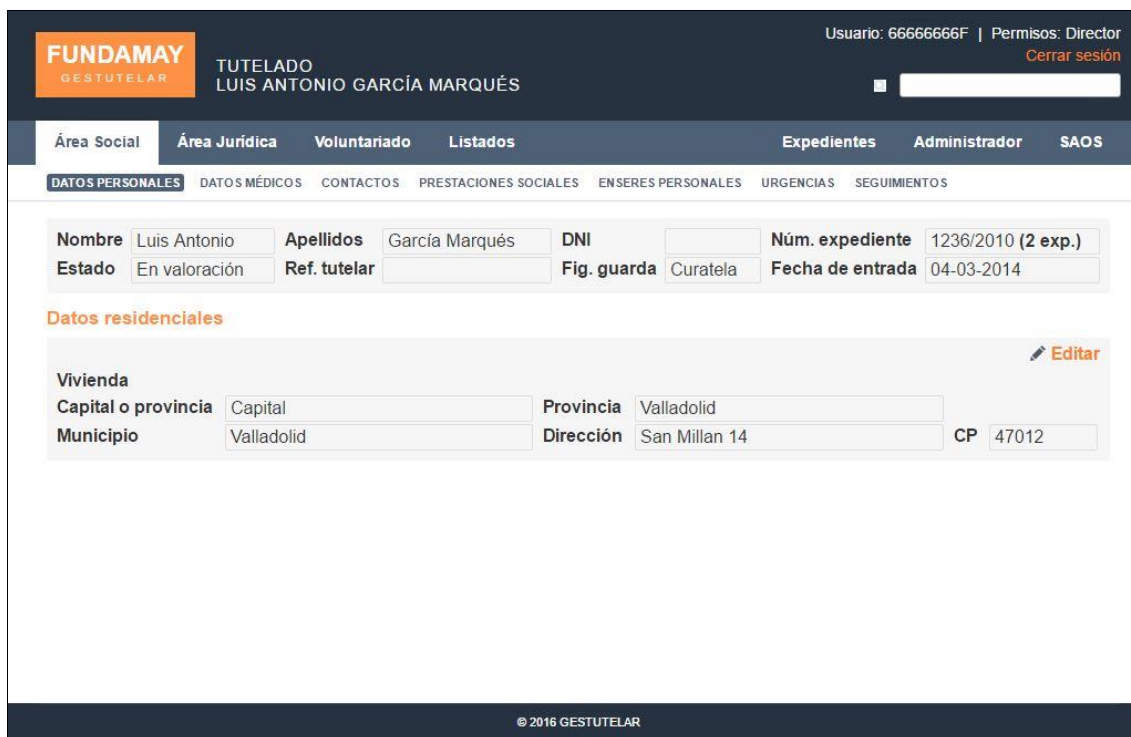


Figura 73. Misma web con distintos permisos.

En las imágenes anteriores observamos la misma web pero con distintos permisos de visión de las distintas áreas, para observarlo mejor se ha seleccionado un tutelado por defecto para poder ver los permisos activados correspondientes a la visualización de ciertos datos de un tutelado.

En la primera imagen se puede observar que la web tiene el permiso para poder ver todos los datos tanto del menú principal con todas las subsecciones, y podemos ver todos los datos.

En la segunda imagen se tiene permiso de todas las áreas excepto de la sección área económica. Respecto a los datos del paciente se pueden ver lo mismo que en la primera imagen.

En la última de las imágenes se ha quitado los permisos relativos al área económica y los datos que indican los traslados del paciente seleccionado.

En la siguiente tabla indicaremos todos los permisos que se utilizan para ofrecer la flexibilidad a la herramienta de gestión de tutelas jurídicas otorgando la visión o no de las diferentes áreas.

- Área social: podemos tener permisos para las siguientes subsecciones:
 - Contactos.
 - Enseres.
 - Médico.

- Personal.
- Prestaciones.
- Seguimientos.
- Urgencias

Si tenemos activado uno o más de estos permisos la sección principal del área social será visible, en caso de que ninguna de no poseer los permisos de visión de estas subsecciones, el área social no será visible en la web de gestión de tutelas.

-Área económica: podemos tener permisos para las siguientes subsecciones:

- Cuentas.
- Ingresos.
- Obligaciones.
- Seguros.

Si tenemos activado uno o más de estos permisos la sección principal del área económica será visible, en caso de que ninguna de no poseer los permisos de visión de estas subsecciones, el área económica no será visible en la web de gestión de tutelas.

-Área jurídica: podemos tener permisos para las siguientes subsecciones:

- Fechas.
- Autorizaciones.
- Juzgados.
- Retribución.

Ocurre exactamente lo mismo que en el resto de áreas, para que sea visible en el menú principal esta sección ha de estar activado un o más permisos de esta subsección.

-Voluntariado: solo podemos hacer visible por completo esta sección ya que las subsecciones tienen dependencia entre ellas y no es lógico mostrarlas por separado.

-Área Listado: podemos tener permisos para las siguientes subsecciones:

- Informes.
- Juzgado.
- Residencias.

-SAOS: solo podemos hacer visible por completo esta sección ya que las subsecciones tienen dependencia entre ellas y no es lógico mostrarlas por separado.

-Área expedientes: solo podemos hacer visible por completo esta sección ya que las subsecciones tienen dependencia entre ellas y no es lógico mostrarlas por separado.

-Área administración: en esta área, los permisos de visión que otorgamos o no tienen que ver con la ampliación de la web. Al tener los permisos de esta área activados podemos acceder a la sección que permite modificar la apariencia de la web.

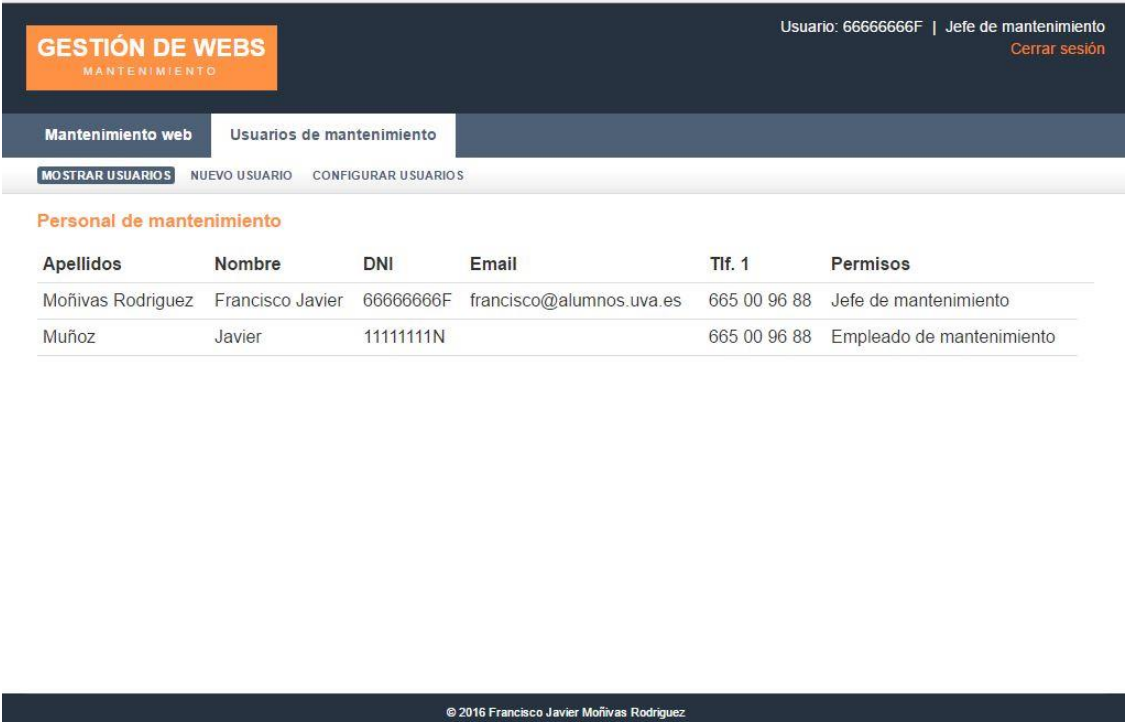
-Permisos de tercer nivel, permisos de visión de datos de un paciente. Esta sección contiene todos los datos que podemos ver de un paciente cuando esta seleccionado en la aplicación. Podemos determinar qué datos queremos o no ver de los tutelados ya que hay ciertos datos que no se desean ver.

6.3.5 SECCIÓN USUARIOS DE MANTENIMIENTO

La segunda sección principal de la web de mantenimiento, está relacionada con el mantenimiento de los propios usuarios de esta web.

En esta web hay que distinguir que dependiendo del rango, permisos, que tiene el usuario registrado en la web tendrá acceso a unas subsecciones u a otras.

En el caso de estar registrado como jefe de mantenimiento, la web de mantenimiento en su sección de mantenimiento de usuarios tiene 3 subsecciones.



The screenshot shows the 'Gestión de Webs' maintenance interface. At the top right, it displays 'Usuario: 66666666F | Jefe de mantenimiento' and a 'Cerrar sesión' link. The main navigation bar includes 'Mantenimiento web' and 'Usuarios de mantenimiento'. Below this, there are buttons for 'MOSTRAR USUARIOS', 'NUEVO USUARIO', and 'CONFIGURAR USUARIOS'. The 'Personal de mantenimiento' section contains a table with the following data:

Apellidos	Nombre	DNI	Email	Tif. 1	Permisos
Moñivas Rodríguez	Francisco Javier	66666666F	francisco@alumnos.uva.es	665 00 96 88	Jefe de mantenimiento
Muñoz	Javier	11111111N		665 00 96 88	Empleado de mantenimiento

At the bottom of the page, there is a copyright notice: '© 2016 Francisco Javier Moñivas Rodríguez'.

Figura 74. Sección mantenimiento de usuarios web. Vista del jefe de mantenimiento.

Las tres secciones a las que tiene acceso el jefe de mantenimiento son:

-Mostrar usuarios: tabla que muestra todos los usuarios registrados en la web con el cargo que desempeña dentro de la misma web.

-Nuevo usuario: aparece un formulario el cual al completarlo y validarlo por medio de las distintas técnicas de validación que se comentaron en la descripción técnica del proyecto, se añade un nuevo usuario a la base de datos de la web de mantenimiento.

El formulario será similar al de la imagen 69.

Nuevo usuario de mantenimiento

Permisos: *	<input type="text" value="- Seleccionar -"/>
Nombre: *	<input type="text"/>
Apellidos: *	<input type="text"/>
DNI: *	<input type="text"/>
Password: *	<input type="text" value="B2aWDkuH"/> <small>▸ La contraseña debe tener entre 6 y 15 caracteres.</small>
Provincia:	<input type="text" value="- Seleccionar -"/>
Municipio:	<input type="text"/>
Dirección:	<input type="text"/>
CP:	<input type="text"/>
Telefono 1: *	<input type="text"/>
Telefono 2:	<input type="text"/>
Email:	<input type="text"/>
Titulación:	<input type="text"/>

Figura 75. Formulario de nuevo usuario.

El formulario es fácil de completar y tiene ayudas en caso de que algún campo sea completado de forma incorrecta. Al completar el formulario, para almacenar el nuevo usuario es necesario pulsar el botón “guardar” situado en la parte inferior del formulario para mandar los datos a la base de datos y almacenar el nuevo usuario de la web.

-Configurar usuario: aparece un listado con los distintos usuarios que hay en la base de datos de la web y dos iconos situados a la derecha. Esta sección solo es visible si accedemos a la web con los permisos de jefe de mantenimiento, en el caso de ser

empleado de mantenimiento no tenemos la capacidad de modificar los datos de ninguno de los usuarios de esta web.

Personal de mantenimiento

Apellidos	Nombre	DNI	Email	Tif. 1	Permisos	
Moñivas Rodriguez	Francisco Javier	66666666F	francisco@alumnos.uva.es	665 00 96 88	Jefe de mantenimiento	 
Muñoz	Javier	11111111N		665 00 96 88	Empleado de mantenimiento	 

Figura 76. Configurar usuarios.

Respecto a los dos iconos, que son intuitivos, al pulsarlo tenemos dos funcionalidades distintas.

Al hacer click en el aspa de uno de los usuarios se elimina ese usuario de la web mientras que si hacemos click en el lapicero accedemos a un formulario que viene completado con los datos de ese paciente y podemos modificar sus datos y/o permisos.

CAPÍTULO 7

Presupuesto económico

Para lograr los objetivos iniciales del proyecto se han necesitado de diferentes herramientas tanto para el desarrollo del software como la utilidad de un hardware potente para almacenar la base de datos.

- Un ordenador portátil para distintas fases: tanto en el desarrollo de la nueva web de gestión de permisos como de la ampliación de la web existente de gestión de tutelas, como para la creación del documento final explicativo y la presentación y defensa del proyecto.

Se ha utilizado un ordenador Toshiba Intel(R) Core(TM) i7, 2.10Ghz y 8 Gb de memoria RAM con un sistema operativo de Windows 10 de 64 bits.

-Un ordenador de sobremesa para la etapa de desarrollo, Dell Intel(R) Core(TM) i5 de 3.20 Ghz y 8 Gb de RAM con un sistema operativo Windows 10 de 64 bits.

-PHP 5.6.16 con su extensión mysqli para realizar la conexión con MySQL.

-MySQL 5.7.9 como sistema gestor de base de datos.

-Servidor web Apache/2.4.17.

-Tecnologías estándar web como son CSS3, HTML5, JavaScript con su librería jQuery.

-Editores de texto para la fase de desarrollo, en nuestro caso el notepad++ y el Dreamweaver.

-Bouml, editor de casos de uso para la fase de documentación y estructuración inicial del proyecto.

Todas las herramientas utilizadas para el desarrollo del proyecto han sido por medio de herramientas gratuitas sin necesidad de comprar licencias, ahí nos ahorramos mucho dinero a la hora de desarrollar el software.

El tiempo de desarrollo de la aplicación teniendo en cuenta la fase de documentación, las reuniones con la tutora del proyecto, fase de desarrollo, fase de pruebas y elaboración del documento final ha sido de 210 horas.

Con los salarios actuales de un programador web freelance en la actualidad (año 2017) con menos de un año de experiencia en el desarrollo de aplicaciones web (aproximadamente 18€ la hora), experiencia que poseía al inicio del desarrollo, el coste total del proyecto incluyendo todas las fases de la misma sería de unos 3.780€.

Se puede ver que es muy bajo el coste de desarrollo de la aplicación, pero el grueso del coste sería cuando se necesite realizar tareas de soporte otorgando los permisos a las diferentes fundaciones que deseen poseer la aplicación de tutela a las cuales debemos ofrecerlas el nuevo servicio. Esta tarea de soporte será fundamental en la idea de nuestra aplicación.

El resto de costes que nos puede suponer, será la compra de un servidor web físico para almacenar los datos de los permisos de visión de las diferentes fundaciones que contraten este servicio. En el mercado actual el precio ronda los 200€ un servidor de 1Tb de memoria, lo cual es suficiente para el comienzo.

En definitiva, el proyecto no es extremadamente caro elaborarlo, lo que realmente proporcionará gastos será el momento en que se necesite contratar personas para realizar las tareas de mantenimiento de las fundaciones a las que ofrecemos este servicio.

CAPÍTULO 8

Conclusiones y líneas futuras

En el último capítulo del documento se pasa a detallar las conclusiones del proyecto y la visión de futuro que se tiene con la realización de este proyecto.

Además detallaremos en este apartado una serie de problemas que surgieron a la hora de elaborar el proyecto y como fueron solventados.

8.1 CONCLUSIONES

El proyecto surge como ampliación y mejora de una aplicación web existente. El primer paso que tuvimos que dar, fue el estudio de la web que existía y el conjunto de funcionalidades que ofrecía, observando en que podíamos mejorarla y conseguir la flexibilidad que nos pedían para este proyecto.

Para poder tomar la decisión de cómo mejorar la aplicación que existía, nos centramos en estudiar y buscar información acerca de las diferentes fundaciones de gestión de tutela de pacientes que existen en la actualidad y llegamos a la conclusión de que cada día aparecen más fundaciones que se encargan de estas labores, pero no todas estas fundaciones se encargan de tutelar todos los áreas disponibles que había en la aplicación. Por lo tanto, se llega a la conclusión de que teníamos que otorgar una serie de permisos a la aplicación de tutela que existía para hacer visible o no las diferentes secciones dependiendo de si la fundación la necesitaba o no.

Una vez llegado a la conclusión anterior, nos centramos en estudiar las posibilidades que teníamos a la hora de ofrecer este nuevo servicio que permitiera ver las distintas áreas que poseía la aplicación en relación a los permisos que se la otorgara. Se nos ocurrieron tres ideas de las que nos decantamos por crear una web externa, por la cual otorgaríamos los permisos necesarios a la web de gestión de tutelas.

Cuando vimos esta situación, observamos que la herramienta que estaba desarrollada para la gestión de tutelas, era muy potente y que con la flexibilidad que podíamos otorgarla, por medio de esta nueva web, podíamos crear una herramienta genérica útil para todas esas fundaciones que se encargan de gestión de tutelas de personas pero que se centran en un solo área o en un conjunto de áreas y no en todas las disponibles.

Una vez terminados los primeros pasos de estudio y decisión de como flexibilizar la web, nos centramos en el análisis de todas las alternativas disponibles para el desarrollo de la aplicación tanto en el lado del cliente como en el lado del servidor así como los distintos sistemas de gestión de bases de datos existentes.

Tras analizar estas tecnologías llegamos al primer punto en el cual necesitábamos tomar una decisión para elegir que tecnologías de las analizadas utilizaríamos a la hora de elaborar nuestro proyecto. Por el lado del servidor, utilizamos el lenguaje de programación PHP con el sistema gestor de base de datos MySQL y su lenguaje SQL. En el lado del cliente, se utiliza los lenguajes estándar de programación web que son HTML, CSS y JavaScript con su librería jQuery.

Como en el proyecto necesitábamos crear formularios se buscó técnicas que simplificaran la creación y validación de formularios encontrando el framework Zebra_Form para facilitar este desarrollo.

La segunda parte del proyecto, se centró en la manera de poder ampliar la aplicación de gestión de tutelas llegando a la conclusión de que podíamos añadir la posibilidad de que se pudiera modificar el estilo de esta web.

Se buscó información para poder realizar esta modificación de estilos de la web y nos decantamos por utilizar las hojas de estilo dinámicas con lenguaje PHP. Además, decidimos buscar una manera sencilla y fácil de implementar por el lado del cliente para poder seleccionar las distintas secciones a modificar su apariencia llegando a encontrar un atributo muy potente de HTML (`input="color"`), con la desventaja de que no es compatible con todos los navegadores por lo que decidimos diferenciar la nueva pantalla de configurar el estilo de la web dependiendo en qué navegador se esté ejecutando la aplicación.

En el caso de que el navegador soportara este atributo, para poder modificar la apariencia se realiza de forma sencilla con el atributo mencionado el cual al ser pulsado aparece una paleta de colores para seleccionar el color deseado para cada una de las secciones que deseamos modificar su estilo. Si el navegador no soportaba este método, decidimos crear una serie de estilos por defecto donde el usuario de la aplicación podía seleccionar uno u otro de los estilos.

Con todos los apartados anteriores analizados y con la elección de las distintas tecnologías de desarrollo, comenzamos elaborando la nueva web encargada de gestionar los permisos de la aplicación de gestión de tutelas.

El primer paso fue desarrollar la vista de la aplicación, nos basamos en un estilo amigable y sencillo de manejar por parte de los usuarios encargados de modificar los permisos de visión de la web.

Después de desarrollar el diseño de la web, comenzamos a crear la base de datos necesaria para almacenar los permisos que otorgamos a la aplicación que existía

y a los usuarios que se encargarían de acceder a esta nueva web para realizar tareas de mantenimiento de permisos.

Una vez creada la web, la base de datos y la funcionalidad de las distintas secciones, se llevó a cabo la primera fase de pruebas que consistió en añadir una primera web a la que otorgaríamos los permisos de visión de las diferentes secciones de tutela que posee la herramienta con la que comenzamos el proyecto. Estas pruebas consistían en comprobar el correcto funcionamiento de la web, otorgando permisos y comprobando que si estaban activados se vieran las áreas respectivas al permiso.

La segunda fase de pruebas de la nueva web, nos centramos en hacer fácil de manejar. Para ello nos centramos en crear formularios para el manejo de selección de los permisos, con sus respectivas pruebas enseñando la web a unos compañeros para que comprobaran si era sencilla de manejar o muy compleja.

Una vez que se finaliza por completo el desarrollo de esta web, nos centramos en desarrollar la ampliación de la aplicación de gestión de tutelas añadiendo la funcionalidad por la cual podían los usuarios modificar el estilo de la web.

Para esta fase de desarrollo se necesita crear una nueva tabla en la base de datos de la aplicación de gestión de tutelas, que almacenara distintas variables con sus respectivos valores para modificar los colores de las distintas áreas de la web.

La última parte del proyecto, se realiza un manual de usuario sencillo y muy detallado con el mayor número de imágenes posibles para que los usuarios finales tanto de la web de mantenimiento de permisos como los propios usuarios de la aplicación de gestión de tutelas puedan hacer uso de ella en caso de no conocer el correcto funcionamiento del desarrollo.

Por lo tanto, como hemos visto en todo el desarrollo y estudio previo, con la creación de esta nueva web para gestionar permisos de visión de las áreas de la aplicación de gestión de tutelas, conseguimos flexibilizar la aplicación de gestión de tutelas para poder utilizarla como una web genérica para otras fundaciones que deseen contratar los servicios necesarios para utilizarla y así tener una herramienta que facilite el trabajo a muchas personas de distintas fundaciones encargadas de la tutela de pacientes.

Hay que destacar, que la aplicación de la que partimos es de propiedad de la Universidad de Valladolid que se desarrolló para el uso de una fundación concreta, FUNDAMAY. Nosotros nos basamos en la flexibilización de la web de que esta aplicación podría ser prestada por la Universidad de Valladolid a otras fundaciones que realicen tareas similares a FUNDAMAY o que vendan el servicio que proporciona la aplicación de gestión de tutelas. Si conseguimos que la aplicación pueda ser utilizada por el mayor número posible de fundaciones, la web creada para el mantenimiento de gestión de permisos tenga una mayor utilidad que si solo se ofrece este servicio a una única fundación.

A título personal, la realización de este proyecto ha sido una gran experiencia que me ha ayudado a conocer nuevas tecnologías de desarrollo y las distintas fases de desarrollo que hay en un proceso de desarrollo de software. Además, conseguí aprender a desenvolverme ante la diversidad de problemas que iban surgiendo en la realización del proyecto.

Dentro de los problemas que han surgido en el proyecto, el primero de todo fue la toma de decisiones, las cuales eran complicadas ya que condicionaba por completo la realización del proyecto. Otro de los problemas que surge es la búsqueda de toda la información necesaria para elaborar todo el desarrollo ya que según se iban desarrollando las distintas funcionalidades surgen distintos problemas a los que hay que buscar su solución.

En definitiva, el proyecto ha sido una gran experiencia que me ha ayudado a desenvolverme en las labores del día a día de un ingeniero, enriqueciendo mi sabiduría y aprendiendo cada día cosas nuevas.

8.2 LÍNEAS FUTURAS DEL PROYECTO

Con el desarrollo del proyecto surgen posibilidades de continuar modificándolo y ampliándolo.

Una de las posibilidades surge en relación al estilo. Nosotros solo hemos modificado los colores de la web, se podría desarrollar en futuras versiones, la posibilidad de modificar la colocación de las distintas áreas de la web según desee el usuario.

También vemos que toda la web está en español, pudiendo generalizarla por medio de uso de una variable de sesión que almacene el idioma en que se desee ver la aplicación y con tokens (variables o keys que almacenan palabras que tengan traducciones en diferentes idiomas), teniendo una aplicación de gestión de tutelas en todos los idiomas existentes.

Respecto a la web de gestión de permisos que hemos realizado para generalizar la herramienta de tutelas que existe, se podían añadir nuevas funcionalidades. Dentro de estas nuevas funcionalidades podíamos añadir la posibilidad de interactuar con la web de gestión de tutelas para poder añadir nuevos usuarios o gestionar los usuarios de esa aplicación.

Al igual que ocurre con la aplicación de gestión de tutelas, en esta de gestión de permisos sería posible desarrollarla en más idiomas.

Otra proposición, sería desarrollar esta nueva web en otras tecnologías como son IOS o Android para poder tener acceso desde diferentes dispositivos móviles y no depender de un ordenador.

Por último, se podrían añadir nuevos estilos creados a la aplicación de gestión de tutelas y la funcionalidad de modificar la apariencia de la web no solo dejarla en manos del director o administrador de dicha aplicación, sino que cada usuario de la aplicación pudiera tener su propio estilo en la aplicación, como ocurre en Gmail.

Bibliografía

[1] Proyección de la Población de España a Corto Plazo 2013-2023. (2013, 22 de noviembre). Instituto Nacional de Estadística

[2] FUNDAMAY. (2017). Descripción de la fundación. Recuperado de <http://www.fundamay.org/> (mayo 2017)

[3] Niño, Jesús. (2011). *Gestores de contenidos*. ISBN:8490030987.

[4] Caballero, Carlos. (2015). *Programación con lenguajes de guion en páginas web*. ISBN – 8428396876.

[5] Fernández, C. (2008). *Fundamentos de programación*. ISBN:8448107004.

[6] Gauchat, J. (3 de julio de 2013). *El Gran Libro De HTML5, CSS3 Y Javascript*. ISBN10: 8426719953. ISBN13: 978-8426719959.

[7] Durango, A. (16 de febrero de 2014). *Diseño Web con CSS*. ISBN-10: 1495967824. ISBN-13: 978-1495967825

[8] Resig, J. and Bibeault, B. (December 2012). *Secrets of the JavaScript Ninja*. ISBN:9781933988696.

[9] Sawyer, D . (2012). *JavaScript y jQuery*. ISBN10:844153151X. ISBN13:9788441531512.

[10] Babin, L. (2007). *Introducción a Ajax con PHP*. ISBN 8441522006, 9788441522008.

[11] Mitchell, S., Walther, S. and Seven, F. (2005). *ASP.NET Tips, Tutorials, and Code, Volumen 1*. ISBN 0-672-3243-2

[12] Harms, D. (2001). *JSP, Servlets, and MySQL*. ISBN 0764547879, 9780764547874.

[13] Berman, J. (2007). *the perl programming language*. ISBN 978-0-7637-5758-8.

[14] Cobo, A. (2005). *PHP y MySQL: Tecnología para el desarrollo de aplicaciones web*. ISBN 8479787066, 9788479787066.

[15] Connolly, T., Begg C., Pearson y Wesley, A. (2005). *Sistemas de bases de datos: un enfoque práctica para diseño, implementación y gestión*. ISBN: 8478290753, 9788478290758.

[16] Nevado, M. (2009). *Introducción a Las Bases de Datos Relacionales*. ISBN 8499836178, 9788499836171.

[17] Marqués, M. (2001). *SQL y desarrollo de aplicaciones en Oracle 8*. ISBN 8480213183, 9788480213189.

[18] Kofler, M. (2001). *MySQL*. ISBN 1893115577, 9781893115576.

[19] Gabillaud, J. (2009). *SQL Server 2008 - SQL, Transact SQL: Diseño y creación de una base de datos*. ISBN 2746049112, 9782746049116.

[20] Merino, E and Martino, L. (1995). *Sistemas de bases de datos orientadas a objetos: conceptos y arquitecturas*. ISBN: 0201653567, 9780201653564.

[21] Zebra Form. (21 de abril). Zebra_Form, a jQuery augmented PHP library for creating and validating HTML forms. Recuperado de <http://stefangabos.ro/phplibraries/zebra-form/> (Marzo 2017)

[22] Jackson, J. (2016). *HTML5 Quick Markup Reference*. ISBN 1430265361, 9781430265368.

